

LNAI 4099

Qiang Yang  
Geoff Webb (Eds.)

# PRICAI 2006: Trends in Artificial Intelligence

9th Pacific Rim International Conference  
on Artificial Intelligence  
Guilin, China, August 2006, Proceedings



Springer

Lecture Notes in Artificial Intelligence 4099

Edited by J. G. Carbonell and J. Siekmann

Subseries of Lecture Notes in Computer Science

Qiang Yang Geoff Webb (Eds.)

# PRICAI 2006: Trends in Artificial Intelligence

9th Pacific Rim International Conference  
on Artificial Intelligence  
Guilin, China, August 7-11, 2006  
Proceedings

**Series Editors**

Jaime G. Carbonell, Carnegie Mellon University, Pittsburgh, PA, USA  
Jörg Siekmann, University of Saarland, Saarbrücken, Germany

**Volume Editors**

Qiang Yang  
Hong Kong University of Science and Technology  
Department of Computer Science and Engineering  
Clearwater Bay, Kowloon, Hong Kong, China  
E-mail: qyang@cse.ust.hk

Geoff Webb  
Monash University  
School of Information Technology  
P.O. Box 75, Victoria 3800, Australia  
E-mail: Geoff.Webb@infotech.monash.edu

Library of Congress Control Number: 2006929802

CR Subject Classification (1998): I.2, F.1

LNCS Sublibrary: SL 7 – Artificial Intelligence

ISSN 0302-9743  
ISBN-10 3-540-36667-9 Springer Berlin Heidelberg New York  
ISBN-13 978-3-540-36667-6 Springer Berlin Heidelberg New York

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer. Violations are liable to prosecution under the German Copyright Law.

Springer is a part of Springer Science+Business Media

[springer.com](http://springer.com)

© Springer-Verlag Berlin Heidelberg 2006  
Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India  
Printed on acid-free paper SPIN: 11801603 06/3142 5 4 3 2 1 0

# Preface

The Pacific Rim International Conference on Artificial Intelligence (PRICAI) is one of the preeminent international conferences on artificial intelligence (AI). PRICAI 2006 (<http://www.csse.monash.edu.au/pricai06/Header.htm>) was the ninth in this series of biennial international conferences highlighting the most significant contributions to the field of AI. The conference was held during August 7-11, 2006, in the beautiful city of Guilin in Guangxi Province, China.

As in previous years, this year's technical program saw very high standards in both the submission and paper review process, resulting in an exciting program that reflects the great variety and depth of modern AI research. This year's contributions covered all traditional areas of AI, including machine learning and data mining, knowledge representation and planning, probabilistic reasoning, constraint satisfaction, computer vision and automated agents, as well as various exciting and innovative applications of AI to many different areas. There was particular emphasis in the areas of machine learning and data mining, intelligent agents, evolutionary computing and intelligent image and video analysis.

The technical papers in this volume were selected from a record of 596 submissions after a rigorous review process. Each submission was reviewed by at least three members of the PRICAI Program Committee, including at least two reviewers and one Vice Program Chair. Decisions were reached following discussions among the reviewers of each paper, and finalized in a highly selective process that balanced many aspects of a paper, including the significance of the contribution and originality, technical quality and clarity of contributions, and relevance to the conference objectives.

The technical paper review process was very selective. Out of the 596 submissions, we accepted 81(13.6%) papers for oral presentation and 87 papers (14.6%) for presentation as posters at the conference. This corresponds to an overall acceptance rate of 28.8% among all submissions. In addition, we were honored to have keynote speeches by notable leaders in the field: Pedro Domingos, Ah Chung Tsoi, Wei-Xiong Zhang, Ning Zhong and Zhi-Hua Zhou and an invited talk by a further distinguished academic, Kuang-chih Huang. In addition to the main conference, PRICAI 2006 also featured an exciting tutorial program and workshop program, as well as several co-located international conferences.

PRICAI 2006 relied on the generous help of many people. We extend our appreciation to the Vice PC Chairs: David Albrecht, Hung Bui, William Cheung, John Debenham, Achim Hoffmann, Huan Liu, Wee Keong Ng, Hui Xiong, Mingsheng Ying, Shichao Zhang and Zhi-Hua Zhou, as well as the hard work of 333 members of the Program Committee and reviewers. We thank in particular the professional help of Rong Pan and Michelle Kinsman, who provided an enormous amount of assistance with the conference reviewing system and website, and thank the generous help of the previous PRICAI Chair Chengqi Zhang

and the support of Local Arrangement Chairs Shichao Zhang and Taoshen Li. We thank the strong support of the Conference General Chairs Ruqian Lu and Hideyuki Nakashima. We also thank the PRICAI Steering Committee for giving us this chance to co-chair the PRICAI 2006 conference, and Springer for its continuing support in publishing the proceedings.

August 2006

Qiang Yang and Geoff Webb

# **Organization**

## **Conference Co-chairs:**

Ruqian Lu (Chinese Academy of Sciences) China  
Hideyuki Nakashima (Future University - Hakodate) Japan

## **Program Committee Co-chairs:**

Qiang Yang (Hong Kong University of Science and Technology) Hong Kong  
Geoff Webb (Monash University) Australia

## **Organizing Chair:**

Shichao Zhang (Guangxi Normal University) China  
Taoshen Li (Guangxi Normal University) China

## **Workshops Chair:**

Riichiro Mizoguchi (Osaka University) Japan  
Rong Pan (Hong Kong University of Science and Technology) Hong Kong

## **Tutorials Chair:**

Charles Ling (University of Western Ontario) Canada

## **Industrial Chair:**

Wei-Ying Ma (Microsoft Research) China

## **Sponsorship Co-chairs:**

Zhongzhi Shi (Chinese Academy of Sciences) China  
Chengqi Zhang (University of Technology, Sydney) Australia

## **Publicity Co-chairs:**

Jian Pei (Simon Fraser University) Canada  
Xudong Luo (University of Southampton) UK

## VIII Organization

### Program Committee Vice Chairs:

David Albrecht (Monash University) Australia  
Hung Bui (SRI International) USA  
William Cheung (Hong Kong Baptist University) Hong Kong  
John Debenham (University of Technology, Sydney) Australia  
Achim Hoffmann (University of New South Wales) Australia  
Huan Liu (Arizona State University) USA  
Wee Keong Ng (Nanyang Technological University) Singapore  
Hui Xiong (Rutgers University) USA  
Mingsheng Ying (Tsinghua University) China  
Shichao Zhang (University of Technology, Sydney) Australia  
Zhi-Hua Zhou (Nanjing University) China

### Program Committee:

David Albrecht	Christian Freksa
Aijun An	Yan Fu
A. Anbulagan	Sharon XiaoYing Gao
Hiroki Arimura	Yang Gao
Laxmidhar Behera	Shyam Gupta
Hung Bui	Udo Hahn
Longbing Cao	James Harland
Tru Cao	Achim Hoffmann
Nicholas Cercone	Jiman Hong
Rong Chen	Michael Horsch
Yin Chen	Wynne Hsu
Zheng Chen	Xiangji Huang
Jian-Hung Chen	Joshua Huang
Songcan Chen	Shell Ying Huang
David Cheung	Zhiyong Huang
William Cheung	Mitsuru Ishizuka
Yiu-ming Cheung	Sanjay Jain
Sung-Bae Cho	Margaret Jefferies
Andy Chun	Rong Jin
Paul Compton	Geun Sik Jo
Jirapun Daengdej	Ken Kaneiwa
Honghua Dai	Hiroyuki Kawano
Dao-Qing Dai	Ray Kemp
Pallab Dasgupta	Shamim Khan
Manoranjan Dash	Deepak Khemani
John Debenham	Boonserm Kijsirikul
James Delgrande	Eun Yi Kim
Zhi-Hong Deng	Yasuhiko Kitamura
Norman Foo	Alistair Knott

Ramamohanarao Kotagiri	Rudy Setiono
Peep Kngas	Yidong Shen
Kazuhiro Kuwabara	ZhongZhi Shi
James Kwok	Daming Shi
Wai Lam	Akira Shimazu
Longin Jan Latecki	Carles Sierra
Wee Sun Lee	Arul Siromoney
Tze Yun Leong	Raymund Sison
Xue Li	Paul Snow
Hang Li	Von-Wun Soo
Chun-Hung Li	Kaile Su
Jinyan Li	Ruixiang Sun
Gerard Ligozat	Wing Kin Sung
Ee-Peng Lim	Hideaki Takeda
Zuoquan Lin	Ah-Hwee Tan
Hong Liu	Chew Lim Tan
Tie-Yan Liu	Qing Tao
Jiming Liu	Takao Terano
Bing Liu	John Thornton
Huan Liu	Kai Ming Ting
Dickson Lukose	Shusaku Tsumoto
Xudong Luo	Miroslav Velev
Michael Maher	Toby Walsh
Yuji Matsumoto	Huaiqing Wang
Chris Messon	Jun Wang
Chunyan Miao	Lipo Wang
KyongHo Min	Takashi Washio
Antonija Mitrovic	Ian Watson
Shivashankar Nair	Geoff Webb
Geok See Ng	Ji-Rong Wen
Wee Keong Ng	Graham Williams
Zaiqing Nie	Wayne Wobcke
Masayuki Numao	Limsoon Wong
Takashi Okada	Zhaohui Wu
Lin Padgham	Xindong Wu
Rong Pan	Hui Xiong
Jeng-Shyang Pan	Baowen Xu
Hyeyoung Park	Seiji Yamada
Fred Popowich	Jun Yan
Arun K. Pujari	Qiang Yang
Hiok Chai Quek	Ying Yang
Anca Luminita Ralescu	Hyun Seung Yang
Jochen Renz	Yiyu Yao
Claude Sammut	Min Yao
Ken Satoh	Roland H. C. Yap

Dit-Yan Yeung	Benyu Zhang
Jian Yin	Junping Zhang
Mingsheng Ying	Mingyi Zhang
Xinghuo Yu	Xuegong Zhang
Jeffrey Xu Yu	Byoung-Tak Zhang
Lei Yu	Jian Zhang
Jian Yu	Shichao Zhang
Pong Chi Yuen	Jun Zhang
Huajun Zeng	Ning Zhong
Hongbin Zha	Aoying Zhou
Chengqi Zhang	Shuigeng Zhou
Zili Zhang	Zhi-Hua Zhou

## Additional Reviewers:

Mina Akaishi	Martina Dankova
Bill Andreopoulos	Luc De Raedt
Hiroshi Aoyama	Marina De Vos
Naresh Babu	Aldric Degorre
Nilufar Baghaei	Mike Dixon
Stuart Bain	Jeremy Dokter
Shankar Balachandran	Didier Dubois
Ravindran Balaraman	Frank Dylla
Thomas Barkowsky	Weiguo Fan
J. P. Bekmann	Joel Fenwick
Sven Bertel	Liliana Mara Carrillo Flrez
Michael Blumenstein	Lutz Frommberger
Abdenour Bouzouane	Ken-ichi Fukui
Tiberio Caetano	Naoki Fukuta
Lawrence Cavedon	Chun Che Fung
Hong Chang	Gabriel Fung
Ratthachat Chatpatanasiri	Dorian Gaertner
Qingliang Chen	Bin Gao
Jilin Chen	Alban Grastien
Yuanhao Chen	Xue Gui-Rong
Jie Chen	Makoto Haraguchi
Ding-Yi Chen	Xuefeng He
Kenil Cheng	Keijo Heljanko
Pak-Ming Cheung	Jan Hladik
Vic Ciesielski	Chenyong Hu
Andrew Connor	Jinbo Huang
Diana Cukierman	Aaron Hunter
Guang Dai	Sun Jian-Tao
Ugo Dal Lago	Mike Jones

Sindhu Joseph  
Rohit Joshi  
Norihiro Kamide  
Gour Karmakar  
Yoshikiyo Kato  
Elizabeth Kemp  
Philip Kilby  
Kazuki Kobayashi  
Takanori Komatsu  
Yasuo Kudo  
Sreenivasa Kumar  
Satoshi Kurihara  
Roberto Legaspi  
Hua Li  
Xiaodong Li  
Guoliang Li  
Chavalit Likitvivatanavong  
Chenxi Lin  
Han Lin  
Ning Liu  
Jimmy Liu  
Yang Liu  
Xiangyu Luo  
Wei Luo  
Stephen MacDonell  
Stephen Marsland  
Akio Maruyama  
Shouichi Matsui  
Le Ngoc Minh  
Nguyen Le Minh  
Masaharu Mizumoto  
Mikihiko Mori  
Koji Morikawa  
Koichi Moriyama  
Masao Mukaidono  
Hiroshi Murata  
Tsuyoshi Murata  
N. S. Narayanaswamy  
Nide Naoyuki  
Yoshimasa Ohmoto  
Masayuki Okabe  
Yoshiaki Okubo  
Takashi Onoda  
Mehmet A. Orgun  
Mehrdad Oveisí  
Piero Pagliani  
Jeffrey Junfeng Pan  
Tanasanee Phienthrakul  
Adrin Perreau de Pinninck  
Kim Leng Poh  
Wayne Pullan  
Prasertsak Pungprasertying  
Josep Puyol-Gruart  
Ho Bao Quoc  
M. Masudur Rahman  
Shri Rai  
Arthur Ramer  
Delip Rao  
Ramesh Rayudu  
Jochen Renz  
Kai-Florian Richter  
Juan A. Rodriguez-Aguilar  
Maxim Roy  
Jordi Sabater-Mir  
Ashish Sabharwal  
Falko Schmid  
Holger Schultheis  
Inessa Seifert  
Steven Shapiro  
Andy Song  
Fausto Spoto  
Anantaporn Srisawat  
Sufatrio  
Xichen Sun  
Yasufumi Takama  
Shiro Takata  
Martti Tammi  
Thora Tenbrink  
Quan Thanh Tho  
Mirek Truszcynski  
Ivor Tsang  
Dinh Duc Anh Vu  
Jan Oliver Wallgrün  
Meng wang  
Gang Wang  
Minhong Wang  
Yang Wendy Wang  
Amali Weerasinghe  
Miao Wen  
Michael Winter

## XII Organization

Kok Wai Wong  
Swee Seong Wong  
Bozena Wozna  
Wensi Xi  
Shuicheng Yan

Sheng Zhang  
Kai Zhang  
Mengjie Zhang  
Xin Zheng

# Table of Contents

## Keynote Speech

Learning, Logic, and Probability: A Unified View <i>Pedro Domingos</i> .....	1
Impending Web Intelligence (WI) and Brain Informatics (BI) Research <i>Ning Zhong</i> .....	2
Learning with Unlabeled Data and Its Application to Image Retrieval <i>Zhi-Hua Zhou</i> .....	5

## Regular Papers

### Intelligent Agents

Learning as Abductive Deliberations <i>Budhitama Subagdja, Iyad Rahwan, Liz Sonenberg</i> .....	11
Using a Constructive Interactive Activation and Competition Neural Network to Construct a Situated Agent's Experience <i>Wei Peng, John S. Gero</i> .....	21
Rule-Based Agents in Temporalised Defeasible Logic <i>Guido Governatori, Vineet Padmanabhan, Antonino Rotolo</i> .....	31
Compact Preference Representation for Boolean Games <i>Elise Bonzon, Marie-Christine Lagasquie-Schiex, Jérôme Lang</i> .....	41
Agent-Based Flexible Videoconference System with Automatic QoS Parameter Tuning <i>Sungdo Lee, Sanggil Kang, Dongsoo Han</i> .....	51
Kalman Filter Based Dead Reckoning Algorithm for Minimizing Network Traffic Between Mobile Game Users in Wireless GRID <i>Seong-Whan Kim, Ki-Hong Ko</i> .....	61
Affective Web Service Design <i>Insu Song, Guido Governatori</i> .....	71

An Empirical Study of Data Smoothing Methods for Memory-Based and Hybrid Collaborative Filtering <i>Dingyi Han, Gui-Rong Xue, Yong Yu</i> .....	81
Eliminate Redundancy in Parallel Search: A Multi-agent Coordination Approach <i>Jiewen Luo, Zhongzhi Shi</i> .....	91
Intelligent Market Based Learner Modeling <i>Maryam Ashoori, Chun Yan Miao, Angela Eck Soong Goh, Wang Qiong</i> .....	101
User Preference Through Bayesian Categorization for Recommendation <i>Kyung-Yong Jung</i> .....	112
<b>Automated Reasoning</b>	
A Stochastic Non-CNF SAT Solver <i>Rafiq Muhammad, Peter J. Stuckey</i> .....	120
Reasoning About Hybrid Probabilistic Knowledge Bases <i>Kedian Mu, Zuoquan Lin, Zhi Jin, Ruqian Lu</i> .....	130
Update Rules for Parameter Estimation in Continuous Time Bayesian Network <i>Dongyu Shi, Jinyuan You</i> .....	140
On Constructing Fibred Tableaux for BDI Logics <i>Vineet Padmanabhan, Guido Governatori</i> .....	150
The Representation of Multiplication Operation on Fuzzy Numbers and Application to Solving Fuzzy Multiple Criteria Decision Making Problems <i>Chien-Chang Chou</i> .....	161
Finding a Natural-Looking Path by Using Generalized Visibility Graphs <i>Kyeonah Yu</i> .....	170
Comparison Between Two Languages Used to Express Planning Goals: <i>CTL</i> and <i>EAGLE</i> <i>Wei Huang, Zhonghua Wen, Yunfei Jiang, Aixiang Chen</i> .....	180
Trajectory Modification Using Elastic Force for Collision Avoidance of a Mobile Manipulator <i>Nak Yong Ko, Reid G. Simmons, Dong Jin Seo</i> .....	190

A Hybrid Architecture Combining Reactive Plan Execution and Reactive Learning <i>Samin Karim, Liz Sonenberg, Ah-Hwee Tan</i>	200
A Knowledge-Based Modeling System for Time-Critical Dynamic Decision-Making <i>Yanping Xiang, Kim-Leng Poh</i>	212
<b>Machine Learning and Data Mining</b>	
Mining Frequent Itemsets for Protein Kinase Regulation <i>Qingfeng Chen, Yi-Ping Phoebe Chen, Chengqi Zhang, Liangang Li</i>	222
Constructing Bayesian Networks from Association Analysis <i>Ohm Sornil, Sunatashee Poonvutthikul</i>	231
Bayesian Approaches to Ranking Sequential Patterns Interestingness <i>Kuralmani Vellaisamy, Jinyan Li</i>	241
Mining Multi-dimensional Frequent Patterns Without Data Cube Construction <i>Chuan Li, Changjie Tang, Zhonghua Yu, Yintian Liu, Tianqing Zhang, Qihong Liu, Mingfang Zhu, Yongguang Jiang</i>	251
A New Approach to Symbolic Classification Rule Extraction Based on SVM <i>Dexian Zhang, Tiejun Yang, Ziqiang Wang, Yanfeng Fan</i>	261
Feature Selection for Bagging of Support Vector Machines <i>Guo-Zheng Li, Tian-Yu Liu</i>	271
Neural Classification of Lung Sounds Using Wavelet Packet Coefficients Energy <i>Yi Liu, Caiming Zhang, Yuhua Peng</i>	278
Wireless Communication Quality Monitoring with Artificial Neural Networks <i>Dauren F. Akhmetov, Minoru Kotaki</i>	288
Prediction of MPEG Video Source Traffic Using BiLinear Recurrent Neural Networks <i>Dong-Chul Park, Chung Nguyen Tran, Young-Soo Song, Yunsik Lee</i>	298

Dynamic Neural Network-Based Fault Diagnosis for Attitude Control Subsystem of a Satellite <i>Z.Q. Li, L. Ma, K. Khorasani</i>	308
Gauss Chaotic Neural Networks <i>Yao-qun Xu, Ming Sun, Ji-hong Shen</i>	319
Short-Term Load Forecasting Using Multiscale BiLinear Recurrent Neural Network <i>Dong-Chul Park, Chung Nguyen Tran, Yunsik Lee</i>	329
A Comparison of Selected Training Algorithms for Recurrent Neural Networks <i>Suwat Pattamavorakun, Suwarin Pattamavorakun</i>	339
Neural Network Recognition of Scanning Electron Microscope Image for Plasma Diagnosis <i>Byungwhan Kim, Wooram Ko, Seung Soo Han</i>	350
A New Multi-constrained QoS Routing Algorithm in Mobile Ad Hoc Networks <i>Hu Bin, Liu Hui</i>	358
Sparse Kernel Ridge Regression Using Backward Deletion <i>Ling Wang, Liefeng Bo, Licheng Jiao</i>	365
Using Locally Weighted Learning to Improve SMOreg for Regression <i>Chaoqun Li, Liangxiao Jiang</i>	375
Palmprint Recognition Using Wavelet and Support Vector Machines <i>Xinhong Zhou, Yuhua Peng, Ming Yang</i>	385
Context Awareness System Modeling and Classifier Combination <i>Mi Young Nam, Suman Sedai, Phill Kyu Rhee</i>	394
Non-negative Matrix Factorization on Kernels <i>Daoqiang Zhang, Zhi-Hua Zhou, Songcan Chen</i>	404
Modelling Citation Networks for Improving Scientific Paper Classification Performance <i>Mengjie Zhang, Xiaoying Gao, Minh Duc Cao, Yuejin Ma</i>	413
Analysis on Classification Performance of Rough Set Based Reducts <i>Qinghua Hu, Xiaodong Li, Daren Yu</i>	423

Parameter Optimization of Kernel-Based One-Class Classifier on Imbalance Text Learning <i>Ling Zhuang, Honghua Dai</i> .....	434
Clustering-Based Nonlinear Dimensionality Reduction on Manifold <i>Guihua Wen, Lijun Jiang, Jun Wen, Nigel R. Shadbolt</i> .....	444
Sparse Kernel PCA by Kernel K-Means and Preimage Reconstruction Algorithms <i>Sanparith Marukatat</i> .....	454
Clustering-Based Relevance Feedback for Web Pages <i>Seung Yeol Yoo, Achim Hoffmann</i> .....	464
Building Clusters of Related Words: An Unsupervised Approach <i>Deepak P, Delip Rao, Deepak Khemani</i> .....	474
<b>Natural Language Processing and Speech Recognition</b>	
Recognition of Simultaneous Speech by Estimating Reliability of Separated Signals for Robot Audition <i>Shun'ichi Yamamoto, Ryu Takeda, Kazuhiro Nakadai, Mikio Nakano, Hiroshi Tsujino, Jean-Marc Valin, Kazunori Komatani, Tetsuya Ogata, Hiroshi G. Okuno</i> .....	484
Chinese Abbreviation-Definition Identification: A SVM Approach Using Context Information <i>Xu Sun, Houfeng Wang, Yu Zhang</i> .....	495
Clause Boundary Recognition Using Support Vector Machines <i>Hyun-Ju Lee, Seong-Bae Park, Sang-Jo Lee, Se-Young Park</i> .....	505
Large Quantity of Text Classification Based on the Improved Feature-Line Method <i>XianFei Zhang, BiCheng Li, WenBin Mu, Yin Liu</i> .....	515
Automatic Multi-level Summarizations Generation Based on Basic Semantic Unit for Sports Video <i>Jianyun Chen, Xinyu Zhao, Miyi Duan, Tingting Wu, Songyang Lao</i> .....	524
Query-Topic Focused Web Pages Summarization <i>Seung Yeol Yoo, Achim Hoffmann</i> .....	533

## Computer Vision

Invariant Color Model-Based Shadow Removal in Traffic Image and a New Metric for Evaluating the Performance of Shadow Removal Methods <i>Young Sung Soh, Hwanju Lee, Yakun Wang</i>	544
Uncontrolled Face Recognition by Individual Stable Neural Network <i>Xin Geng, Zhi-Hua Zhou, Honghua Dai</i>	553
Fuzzy Velocity-Based Temporal Dependency for SVM-Driven Realistic Facial Animation <i>Pith Xie, Yiqiang Chen, Junfa Liu, Dongrong Xiao</i>	563
Re-ordering Methods in Adaptive Rank-Based Re-indexing Scheme <i>Kang Soo You, Jae Ho Choi, Hoon Sung Kwak</i>	573
Use of Nested K-Means for Robust Head Location in Visual Surveillance System <i>Hyun Jea Joo, Bong Won Jang, Suman Sedai, Phill Kyu Rhee</i>	583
Appearance Based Multiple Agent Tracking Under Complex Occlusions <i>Prithwijit Guha, Amitabha Mukerjee, K.S. Venkatesh</i>	593

## Perception and Animation

Variable Duration Motion Texture for Human Motion Modeling <i>Tianyu Huang, Fengxia Li, Shouyi Zhan, Jianyuan Min</i>	603
A Novel Motion Blending Approach Based on Fuzzy Clustering <i>Xiangbin Zhu</i>	613
Efficient Optimization of Inpainting Scheme and Line Scratch Detection for Old Film Restoration <i>Seong-Whan Kim, Ki-Hong Ko</i>	623
Partial Encryption of Digital Contents Using Face Detection Algorithm <i>Kwangjin Hong, Keechul Jung</i>	632
Relevance Feedback Using Adaptive Clustering for Region Based Image Similarity Retrieval <i>Deok-Hwan Kim, Seok-Lyong Lee</i>	641

## Evolutionary Computing

Learning and Evolution Affected by Spatial Structure <i>Masahiro Ono, Mitsuru Ishizuka</i> . . . . .	651
Immune Clonal Selection Evolutionary Strategy for Constrained Optimization <i>Wenping Ma, Licheng Jiao, Maoguo Gong, Ronghua Shang</i> . . . . .	661
An Intelligent System for Supporting Personal Creativity Based on Genetic Algorithm <i>Heng-Li Yang, Cheng-Hwa Lee</i> . . . . .	671
Generating Creative Ideas Through Patents <i>Guilhua Wen, Lijun Jiang, Jun Wen, Nigel R. Shadbolt</i> . . . . .	681
An Improved Multiobjective Evolutionary Algorithm Based on Dominating Tree <i>Chuan Shi, Qingyong Li, Zhiyong Zhang, Zhongzhi Shi</i> . . . . .	691
Fuzzy Genetic System for Modelling Investment Portfolio <i>Rahib H. Abiyev, Mustafa Menekay</i> . . . . .	701
Fuzzy Genetic Algorithms for Pairs Mining <i>Longbing Cao, Dan Luo, Chengqi Zhang</i> . . . . .	711
A Novel Feature Selection Approach by Hybrid Genetic Algorithm <i>Jinjie Huang, Ning Lv, Wenlong Li</i> . . . . .	721
Evolutionary Ensemble Based Pattern Recognition by Data Context Definition <i>Mi Young Nam, In Ja Jeon, Phill Kyu Rhee</i> . . . . .	730
Quantum-Behaved Particle Swarm Optimization with a Hybrid Probability Distribution <i>Jun Sun, Wenbo Xu, Wei Fang</i> . . . . .	737
A Selection Scheme for Excluding Defective Rules of Evolutionary Fuzzy Path Planning <i>Jong-Hwan Park, Jong-Hwan Kim, Byung-Ha Ahn, Moon-Gu Jeon</i> . . . . .	747
An Improved Genetic-Based Particle Swarm Optimization for No-Idle Permutation Flow Shops with Fuzzy Processing Time <i>Niu Qun, Xingsheng Gu</i> . . . . .	757

## Industrial Applications

Determinants of E-CRM in Influencing Customer Satisfaction <i>Yan Liu, Chang-Feng Zhou, Ying-Wu Chen</i> .....	767
Penalty Guided PSO for Reliability Design Problems <i>Ta-Cheng Chen</i> .....	777
Developing Methodologies of Knowledge Discovery and Data Mining to Investigate Metropolitan Land Use Evolution <i>Yongliang Shi, Jin Liu, Rusong Wang, Min Chen</i> .....	787
Vibration Control of Suspension System Based on a Hybrid Intelligent Control Algorithm <i>Ke Zhang, Shiming Wang</i> .....	797

## Short Papers Part

### Intelligent Agents

An Intelligent Conversational Agent as the Web Virtual Representative Using Semantic Bayesian Networks <i>Kyoung-Min Kim, Jin-Hyuk Hong, Sung-Bae Cho</i> .....	807
Three-Tier Multi-agent Approach for Solving Traveling Salesman Problem <i>Shi-Liang Yan, Ke-Feng Zhou</i> .....	813
Adaptive Agent Selection in Large-Scale Multi-Agent Systems <i>Toshiharu Sugawara, Kensuke Fukuda, Toshio Hirotsu, Shin-ya Sato, Satoshi Kurihara</i> .....	818
A Mobile Agent Approach to Support Parallel Evolutionary Computation <i>Wei-Po Lee</i> .....	823
The Design of Fuzzy Controller by Means of Genetic Algorithms and NFN-Based Estimation Technique <i>Sung-Kwun Oh, Jeoung-Nae Choi, Seong-Whan Jang</i> .....	829
GA-Based Polynomial Neural Networks Architecture and Its Application to Multi-variable Software Process <i>Sung-Kwun Oh, Witold Pedrycz, Wan-Su Kim, Hyun-Ki Kim</i> .....	834

Topical and Temporal Visualization Using Wavelets <i>T. Mala, T.V. Geetha, Sathish Kumar</i> .....	839
LP-TPOP: Integrating Planning and Scheduling Through Constraint Programming <i>Yuechang Liu, Yunfei Jiang</i> .....	844
Integrating Insurance Services, Trust and Risk Mechanisms into Multi-agent Systems <i>Yuk-Hei Lam, Zili Zhang, Kok-Leong Ong</i> .....	849
Cat Swarm Optimization <i>Shu-Chuan Chu, Pei-wei Tsai, Jeng-Shyang Pan</i> .....	854
Heuristic Information Based Improved Fuzzy Discrete PSO Method for Solving TSP <i>Bin Shen, Min Yao, Wensheng Yi</i> .....	859

## Automated Reasoning

A Network Event Correlation Algorithm Based on Fault Filtration <i>Qiuhua Zheng, Yuntao Qian, Min Yao</i> .....	864
CPR Localization Using the RFID Tag-Floor <i>Jung-Wook Choi, Dong-Ik Oh, Seung-Woo Kim</i> .....	870
Development of a Biologically-Inspired Mesoscale Robot <i>Abdul A. Yumaryanto, Jaebum An, Sangyoon Lee</i> .....	875
Timed Petri-Net(TPN) Based Scheduling Holon and Its Solution with a Hybrid PSO-GA Based Evolutionary Algorithm(HPGA) <i>Fuqing Zhao, Yahong Yang, Qiuyu Zhang, Huawei Yi</i> .....	880
Recognition Rate Prediction for Dysarthric Speech Disorder Via Speech Consistency Score <i>Prakasith Kayasith, Thanaruk Theeramunkong, Nuttakorn Thubthong</i> .....	885
An Emotion-Driven Musical Piece Generator for a Constructive Adaptive User Interface <i>Roberto Legaspi, Yuya Hashimoto, Masayuki Numao</i> .....	890
An Adaptive Inventory Control Model for a Supply Chain with Nonstationary Customer Demands <i>Jun-Geol Baek, Chang Ouk Kim, Ick-Hyun Kwon</i> .....	895

Context-Aware Product Bundling Architecture in Ubiquitous Computing Environments <i>Hyun Jung Lee, Mye M. Sohn</i> . . . . .	901
A Relaxation of a Semiring Constraint Satisfaction Problem Using Combined Semirings <i>Louise Leenen, Thomas Meyer, Peter Harvey, Aditya Ghose</i> . . . . .	907
Causal Difference Detection Using Bayesian Networks <i>Tomoko Murakami, Ryohei Orihara</i> . . . . .	912
Tabu Search for Generalized Minimum Spanning Tree Problem <i>Zhenyu Wang, Chan Hou Che, Andrew Lim</i> . . . . .	918
<b>Evolutionary Computing</b>	
Investigation of Brood Size in GP with Brood Recombination Crossover for Object Recognition <i>Mengjie Zhang, Xiaoying Gao, Weijun Lou, Dongping Qian</i> . . . . .	923
An Immune Algorithm for the Optimal Maintenance of New Consecutive-Component Systems <i>Y.-C. Hsieh, P.-S. You</i> . . . . .	929
Immune Genetic Algorithm and Its Application in Optimal Design of Intelligent AC Contactors <i>Li-an Chen, Peiming Zhang</i> . . . . .	934
The Parametric Design Based on Organizational Evolutionary Algorithm <i>Chunhong Cao, Bin Zhang, Limin Wang, Wenhui Li</i> . . . . .	940
Buying and Selling with Insurance in Open Multi-agent Marketplace <i>Yuk-Hei Lam, Zili Zhang, Kok-Leong Ong</i> . . . . .	945
<b>Game</b>	
Ensemble Evolution of Checkers Players with Knowledge of Opening, Middle and Endgame <i>Kyung-Joong Kim, Sung-Bae Cho</i> . . . . .	950
Dynamic Game Level Design Using Gaussian Mixture Model <i>Sangkyung Lee, Keechul Jung</i> . . . . .	955

## Machine Learning and Data Mining

Application Architecture of Data Mining in Telecom Customer Relationship Management Based on Swarm Intelligence <i>Peng Jin, Yunlong Zhu, Sufen Li, Kunyuan Hu</i>	960
Mining Image Sequence Similarity Patterns in Brain Images <i>Haiwei Pan, Xiaoqin Xie, Wei Zhang, Jianzhong Li</i>	965
Weightily Averaged One-Dependence Estimators <i>Liangxiao Jiang, Harry Zhang</i>	970
SV-kNNC: An Algorithm for Improving the Efficiency of k-Nearest Neighbor <i>Anantaporn Srisawat, Tanasanee Phienthrakul, Boonserm Kijsirikul</i>	975
A Novel Support Vector Machine Metamodel for Business Risk Identification <i>Kin Keung Lai, Lean Yu, Wei Huang, Shouyang Wang</i>	980
Performing Locally Linear Embedding with Adaptable Neighborhood Size on Manifold <i>Guihua Wen, Lijun Jiang, Jun Wen, Nigel R. Shadbolt</i>	985
Stroke Number and Order Free Handwriting Recognition for Nepali <i>K.C. Santosh, Cholwich Nattee</i>	990
Diagnosis Model of Radio Frequency Impedance Matching in Plasma Equipment by Using Neural Network and Wavelets <i>Byunguhan Kim, Jae Young Park, Dong Hwan Kim, Seung Soo Han</i>	995
Program Plagiarism Detection Using Parse Tree Kernels <i>Jeong-Woo Son, Seong-Bae Park, Se-Young Park</i>	1000
Determine the Optimal Parameter for Information Bottleneck Method <i>Gang Li, Dong Liu, Yangdong Ye, Jia Rong</i>	1005
Optimized Parameters for Missing Data Imputation <i>Shichao Zhang, Yongsong Qin, Xiaofeng Zhu, Jilian Zhang, Chengqi Zhang</i>	1010
Expediting Model Selection for Support Vector Machines Based on an Advanced Data Reduction Algorithm <i>Yu-Yen Ou, Guan-Hau Chen, Yen-Jen Oyang</i>	1017

Study of the SMO Algorithm Applied in Power System Load Forecasting <i>Jingmin Wang, Kanzhang Wu</i> . . . . .	1022
Filtering Objectionable Image Based on Image Content <i>Zhiwei Jiang, Min Yao, Wensheng Yi</i> . . . . .	1027
MRA Kernel Matching Pursuit Machine <i>Qing Li, Licheng Jiao, Shuyuan Yang</i> . . . . .	1032
Multiclass Microarray Data Classification Using GA/ANN Method <i>Tsun-Chen Lin, Ru-Sheng Liu, Ya-Ting Chao, Shu-Yuan Chen</i> . . . . .	1037
Texture Classification Using Finite Ridgelet Transform and Support Vector Machines <i>Yunxia Liu, Yuhua Peng, Xinhong Zhou</i> . . . . .	1042
Reduction of the Multivariate Input Dimension Using Principal Component Analysis <i>Jianhui Xi, Min Han</i> . . . . .	1047
Designing Prolog Semantics for a Class of Observables <i>Lingzhong Zhao, Tianlong Gu, Junyan Qian, Guoyong Cai</i> . . . . .	1052
A Fingerprint Capture System and the Corresponding Image Quality Evaluation Algorithm Based on FPS200 <i>Hong Huang, Jianwei Li, Wei He</i> . . . . .	1058
Multi-agent Motion Tracking Using the Particle Filter in ISpace with DINs <i>TaeSeok Jin, ChangHoon Park, Soo-hong Park</i> . . . . .	1063
Combining Multiple Sets of Rules for Improving Classification Via Measuring Their Closenesses <i>Yixin Bi, Shengli Wu, Xuming Huang, Gongde Guo</i> . . . . .	1068
<b>Industrial Applications</b>	
Multiple SVMs Enabled Sales Forecasting Support System <i>Yukun Bao, Zhitao Liu, Rui Zhang, Wei Huang</i> . . . . .	1073
The Application of B-Spline Neurofuzzy Networks for Condition Monitoring of Metal Cutting Tool <i>Pan Fu, A.D. Hope</i> . . . . .	1078

Simplified Fuzzy-PID Controller of Data Link Antenna System  
for Moving Vehicles

- Jong-kwon Kim, Soo-hong Park,  
TaeSeok Jin* ..... 1083

Neuron Based Nonlinear PID Control

- Ning Wang, Jinmei Yu* ..... 1089

## Information Retrieval

An Image Retrieval System Based on Colors and Shapes of Objects

- Kuo-Lung Hong, Yung-Fu Chen, Yung-Kuan Chan,  
Chung-Chuan Cheng* ..... 1094

A Hybrid Mood Classification Approach for Blog Text

- Yuchul Jung, Hogun Park, Sung Hyon Myaeng* ..... 1099

Modeling and Classification of Audio Signals Using Gradient-Based

Fuzzy C-Means Algorithm with a Mercer Kernel

- Dong-Chul Park, Chung Nguyen Tran, Byung-Jae Min,  
Sancho Park* ..... 1104

A Quick Rank Based on Web Structure

- Hongbo Liu, Jiaxin Wang, Zehong Yang, Yixu Song* ..... 1109

A Biologically-inspired Computational Model for Perceiving the TROIs  
from Texture Images

- Wooboom Lee, Wookhyun Kim* ..... 1114

A Computer-Assisted Environment on Referential Understanding  
to Enhance Academic Reading Comprehension

- Wing-Kwong Wong, Jian-Hau Lee, Yu-Fen Yang, Hui-Chin Yeh,  
Chin-Pu Chiao, Sheng-Cheng Hsu* ..... 1119

An Object-Oriented Framework for Data Quality Management  
of Enterprise Data Warehouse

- Li Wang, Lei Li* ..... 1125

## Natural Language Processing

Extending HPSG Towards HDS as a Fragment of pCLL

- Erqing Xu* ..... 1130

Chinese Multi-document Summarization Using Adaptive Clustering and Global Search Strategy <i>Dexi Liu, Yanxiang He, Donghong Ji, Hua Yang, Zhao Wu .....</i>	1135
Genetic Algorithm Based Multi-document Summarization <i>Dexi Liu, Yanxiang He, Donghong Ji, Hua Yang .....</i>	1140
MaxMatcher: Biological Concept Extraction Using Approximate Dictionary Lookup <i>Xiaohua Zhou, Xiaodan Zhang, Xiaohua Hu .....</i>	1145
Bootstrapping Word Sense Disambiguation Using Dynamic Web Knowledge <i>Yuanyong Wang, Achim Hoffmann .....</i>	1150
Automatic Construction of Object Oriented Design Models [UML Diagrams] from Natural Language Requirements Specification <i>G.S. Anandha Mala, G.V. Uma .....</i>	1155
A Multi-word Term Extraction System <i>Jisong Chen, Chung-Hsing Yeh, Rowena Chau .....</i>	1160
<b>Neural Networks</b>	
A Multiscale Self-growing Probabilistic Decision-Based Neural Network for Segmentation of SAR Imagery <i>Xian-Bin Wen, Hua Zhang, Zheng Tian .....</i>	1166
Face Detection Using an Adaptive Skin-Color Filter and FMM Neural Networks <i>Ho-Joon Kim, Tae-Wan Ryu, Juho Lee, Hyun-Seung Yang .....</i>	1171
GA Optimized Wavelet Neural Networks <i>Jinhua Xu .....</i>	1176
The Optimal Solution of TSP Using the New Mixture Initialization and Sequential Transformation Method in Genetic Algorithm <i>Rae-Goo Kang, Chai-Yeoung Jung .....</i>	1181
Steering Law Design for Single Gimbal Control Moment Gyroscopes Based on RBF Neural Networks <i>Zhong Wu, Wusheng Chou, Kongming Wei .....</i>	1186

Automatic Design of Hierarchical RBF Networks for System Identification <i>Yuehui Chen, Bo Yang, Jin Zhou</i> .....	1191
Dynamically Subsumed-OVA SVMs for Fingerprint Classification <i>Jin-Hyuk Hong, Sung-Bae Cho</i> .....	1196
Design on Supervised / Unsupervised Learning Reconfigurable Digital Neural Network Structure <i>In Gab Yu, Yong Min Lee, Seong Won Yeo, Chong Ho Lee</i> .....	1201
Car Plate Localization Using Pulse Coupled Neural Network in Complicated Environment <i>Ming Guo, Lei Wang, Xin Yuan</i> .....	1206
A Split-Step PSO Algorithm in Predicting Construction Litigation Outcome <i>Kwok-wing Chau</i> .....	1211
<b>Computer Vision</b>	
An Efficient Unsupervised MRF Image Clustering Method <i>Yimin Hou, Lei Guo, Xiangmin Lun</i> .....	1216
Robust Gaze Estimation for Human Computer Interaction <i>Kang Ryoung Park</i> .....	1222
A New Iris Control Mechanism for Traffic Monitoring System <i>Young Sung Soh, Youngtak Kwon, Yakun Wang</i> .....	1227
Invariant Object Recognition Using Circular Pairwise Convolutional Networks <i>Choon Hui Teo, Yong Haur Tay</i> .....	1232
Face Detection Using Binary Template Matching and SVM <i>Qiong Wang, Wankou Yang, Huan Wang, Jingyu Yang, Yujie Zheng</i> .....	1237
Gain Field Correction Fast Fuzzy c-Means Algorithm for Segmenting Magnetic Resonance Images <i>Jingjing Song, Qingjie Zhao, Yuanquan Wang, Jie Tian</i> .....	1242
LVQ Based Distributed Video Coding with LDPC in Pixel Domain <i>Anhong Wang, Yao Zhao, Hao Wang</i> .....	1248

XXVIII Table of Contents

Object Matching Using Generalized Hough Transform and Chamfer Matching <i>Tai-Hoon Cho</i> . . . . .	1253
<b>Author Index</b> . . . . .	1259

# Learning, Logic, and Probability: A Unified View

Pedro Domingos

Department of Computer Science and Engineering  
University of Washington  
Seattle, WA 98195, U.S.A.  
`pedrod@cs.washington.edu`  
<http://www.cs.washington.edu/homes/pedrod>

AI systems must be able to learn, reason logically, and handle uncertainty. While much research has focused on each of these goals individually, only recently have we begun to attempt to achieve all three at once. In this talk, I describe Markov logic, a representation that combines first-order logic and probabilistic graphical models, and algorithms for learning and inference in it. Syntactically, Markov logic is first-order logic augmented with a weight for each formula. Semantically, a set of Markov logic formulas represents a probability distribution over possible worlds, in the form of a Markov network with one feature per grounding of a formula in the set, with the corresponding weight. Formulas are learned from relational databases using inductive logic programming techniques. Weights can be learned either generatively (using pseudo-likelihood optimization) or discriminatively (using a voted perceptron algorithm). Inference is performed by a weighted satisfiability solver or by Markov chain Monte Carlo, operating on the minimal subset of the ground network required for answering the query. Experiments in link prediction, entity resolution and other problems illustrate the promise of this approach.

This work, joint with Stanley Kok, Hoifung Poon, Matthew Richardson, and Parag Singla, is described in further detail in Domingos et al. [1]. An open-source implementation of Markov logic and the algorithms described in this talk is available in the Alchemy package [2].

## References

1. Domingos, P., Kok, S., Poon, H., Richardson, M., & Singla, P.: Unifying logical and statistical AI. In *Proceedings of the Twenty-First National Conference on Artificial Intelligence*, AAAI Press, Boston, MA, U.S.A. (2006). <http://www.cs.washington.edu/homes/pedrod/papers/aaai06c.pdf>.
2. Kok, S., Singla, P., Richardson, M., and Domingos, P.: The Alchemy system for statistical relational AI. Technical report, Department of Computer Science and Engineering, University of Washington, Seattle, WA, U.S.A. (2005). <http://www.cs.washington.edu/ai/alchemy>.

# Impending Web Intelligence (WI) and Brain Informatics (BI) Research

Ning Zhong

Department of Information Engineering  
Maebashi Institute of Technology, Japan &  
The International WIC Institute  
Beijing University of Technology, China  
[zhong@maebashi-it.ac.jp](mailto:zhong@maebashi-it.ac.jp)

In this talk, we give a new perspective of Web Intelligence (WI) research from the viewpoint of Brain Informatics (BI), a new interdisciplinary field that studies the mechanisms of human information processing from both the macro and micro viewpoint by combining experimental cognitive neuroscience with advanced information technology. As two related emerging fields of research, WI and BI mutually support each other. When WI meets BI, it is possible to have a unified and holistic framework for the study of machine intelligence, human intelligence, and social intelligence. We argue that new instruments like fMRI and information technology will revolutionize both Web intelligence and brain sciences. This revolution will be bi-directional: new understanding of human intelligence through brain sciences will yield a new generation of Web intelligence research and development, and Web intelligence portal techniques will provide a powerful new platform for brain sciences. The synergy between these two fields will advance our understanding knowledge, intelligence, and creativity. As a result, Web intelligence will become a central topic that will change the nature of information technology, in general, and artificial intelligence, in particular, towards human-level Web intelligence.

## References

1. M. Cannataro and D. Talia, “The Knowledge Grid”, *Communications of the ACM*, 46 (2003) 89-93.
2. I. Foster and C. Kesselman (eds.) *The Grid: Blueprint for a New Computing Infrastructure*, Morgan Kaufmann (1999).
3. M.S. Gazzaniga, R. Ivry, and G.R. Mangun, *Cognitive Neuroscience: The Biology of the Mind*, W.W. Norton (2002) 2nd Edition.
4. T.C. Handy, *Event-Related Potentials, A Methods Handbook*, The MIT Press (2004).
5. J. Hu and N. Zhong, “Organizing Multiple Data Sources for Developing Intelligent e-Business Portals”, *Data Mining and Knowledge Discovery*, Vol. 12, Nos. 2-3, Springer (2006) 127-150.
6. S.H. Koslow and S. Subramaniam (eds.) *Databasing the Brain: From Data to Knowledge*, Wiley (2005)

7. J.E. Laird and M. van Lent, "Human-Level AI's Killer Application Interactive Computer Games", *AI Magazine* (Summer 2001) 15-25.
8. Y. Li and N. Zhong, "Mining Ontology for Automatically Acquiring Web User Information Needs", *IEEE Transactions on Knowledge and Data Engineering*, Vol. 18, No. 4 (2006) 554-568.
9. J. Liu, N. Zhong, Y.Y. Yao, and Z.W. Ras, "The Wisdom Web: New Challenges for Web Intelligence (WI)", *Journal of Intelligent Information Systems*, 20(1) Kluwer (2003) 5-9.
10. J. Liu, "Web Intelligence (WI): What Makes Wisdom Web?", *Proc. Eighteenth International Joint Conference on Artificial Intelligence (IJCAI'03)* (2003) 1596-1601.
11. J. McCarthy, "Roads to Human Level AI?", Keynote Talk at Beijing University of Technology, Beijing, China (September 2004).
12. V. Megalooikonomou and E.H. Herskovits, "Mining Structure-Function Associations in a Brain Image Database", K.J. Cios (ed.) *Medical Data Mining and Knowledge Discovery*, Physica-Verlag (2001) 153-179.
13. T.M. Mitchell, R. Hutchinson, R.S. Niculescu, F.Pereira, X. Wang, M. Just, and S. Newman, "Learning to Decode Cognitive States from Brain Images", *Machine Learning*, Vol. 57, Issue 1-2 (2004) 145-175.
14. A. Newell and H.A. Simon, *Human Problem Solving*, Prentice-Hall (1972).
15. M. Ohshima, N. Zhong, Y.Y. Yao, and C. Liu, "Relational Peculiarity Oriented Mining", *Data Mining and Knowledge Discovery*, Springer (in press)
16. B.R. Rosen, R.L. Buckner, and A.M. Dale, "Event-related functional MRI: Past, Present, and Future", *Proceedings of National Academy of Sciences, USA*, Vol. 95, Issue 3 (1998) 773-780.
17. F.T. Sommer and A. Wichert (eds.) *Exploratory Analysis and Data Modeling in Functional Neuroimaging*, The MIT Press (2003)
18. R.J. Sternberg, J. Lautrey, and T.I. Lubart, *Models of Intelligence*, American Psychological Association (2003).
19. Y. Su, L. Zheng, N. Zhong, C. Liu, and J. Liu, "Distributed Reasoning Based on Problem Solver Markup Language (PSML): A Demonstration through Extended OWL", *Proc. 2005 IEEE International Conference on e-Technology, e-Commerce and e-Service (EEE'05)*, IEEE Press (2005) 208-213.
20. Y. Su, J. Liu, N. Zhong, L. Zheng, and C. Liu, "A Method of Distributed Problem Solving on the Web", *Proc. 2005 IEEE/WIC/ACM International Conference on Web Intelligence (WI'05)*, IEEE Press (2005) 42-45.
21. Y.Y. Yao, N. Zhong, J. Liu, and S. Ohsuga, "Web Intelligence (WI): Research Challenges and Trends in the New Information Age", N. Zhong, Y.Y. Yao, J. Liu, S. Ohsuga (eds.) *Web Intelligence: Research and Development*, LNAI 2198, Springer (2001) 1-17.
22. N. Zhong, J. Liu, Y.Y. Yao, and S. Ohsuga, "Web Intelligence (WI)", *Proc. 24th IEEE Computer Society International Computer Software and Applications Conference (COMPSAC 2000)*, IEEE Press (2000) 469-470.
23. N. Zhong, C. Liu, and S. Ohsuga, "Dynamically Organizing KDD Process", *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 15, No. 3, World Scientific (2001) 451-473.
24. N. Zhong, J. Liu, and Y.Y. Yao, "In Search of the Wisdom Web", *IEEE Computer*, 35(11) (2002) 27-31.
25. N. Zhong, "Representation and Construction of Ontologies for Web Intelligence", *International Journal of Foundations of Computer Science*, World Scientific, Vol. 13, No. 4 (2002) 555-570.

26. N. Zhong, J. Liu, and Y.Y. Yao (eds.) *Web Intelligence*, Springer, 2003.
27. N. Zhong, Y.Y. Yao, and M. Ohshima, “Peculiarity Oriented Multi-Database Mining”, *IEEE Transaction on Knowledge and Data Engineering*, Vol. 15, No. 4 (2003) 952-960.
28. N. Zhong, “Developing Intelligent Portals by Using WI Technologies”, J.P. Li et al. (eds.) *Wavelet Analysis and Its Applications, and Active Media Technology*, Vol. 2, World Scientific (2004) 555-567.
29. N. Zhong, J.L. Wu, A. Nakamaru, M. Ohshima, and H. Mizuhara, “Peculiarity Oriented fMRI Brain Data Analysis for Studying Human Multi-Perception Mechanism”, *Cognitive Systems Research*, 5(3), Elsevier (2004) 241-256.
30. N. Zhong, and J. Liu (eds.) *Intelligent Technologies for Information Analysis*, Springer, 2004.
31. N. Zhong, J. Hu, S. Motomura, J.L. Wu, and C. Liu, “Building a Data Mining Grid for Multiple Human Brain Data Analysis”, *Computational Intelligence*, 21(2), Blackwell Publishing (2005) 177-196.
32. N. Zhong, J. Liu, and Y.Y. Yao, “Envisioning Intelligent Information Technologies (iT) from the Stand-Point of Web Intelligence (WI)”, *Communications of the ACM* (in press).

# Learning with Unlabeled Data and Its Application to Image Retrieval

Zhi-Hua Zhou

National Laboratory for Novel Software Technology  
Nanjing University, Nanjing 210093, China  
[zhouzh@nju.edu.cn](mailto:zhouzh@nju.edu.cn)

**Abstract.** In many practical machine learning or data mining applications, unlabeled training examples are readily available but labeled ones are fairly expensive to obtain because labeling the examples require human effort. So, learning with unlabeled data has attracted much attention during the past few years. This paper shows that how such techniques can be helpful in a difficult task, content-based image retrieval, for improving the retrieval performance by exploiting images existing in the database.

## 1 Learning with Unlabeled Data

In the traditional setting of supervised learning, a large amount of training examples should be available for building a model with good generalization ability. It is noteworthy that these training examples should be *labeled*, that is, the ground-truth labels of them are known to the learner. Unfortunately, in many practical machine learning or data mining applications such as web page classification, although a large number of unlabeled training examples can be easily collected, only a limited number of labeled training examples are available since obtaining the labels require human effort. So, exploiting unlabeled data to help supervised learning has become a hot topic during the past few years.

Currently there are three main paradigms for learning with unlabeled data, i.e., *semi-supervised learning*, *transductive learning* and *active learning*.

Semi-supervised learning deals with methods for automatically exploiting unlabeled data in addition to labeled data to improve learning performance. That is, the exploitation of unlabeled data does not need human intervene. Here the key is to use the unlabeled data to help estimate the data distribution. For example, a lot of approaches consider the contribution of the unlabeled examples by using a generative model for the classifier and employing EM to model the label estimation or parameter estimation process [7,9,12]. Note that previous research on semi-supervised learning mainly focus on classification, while semi-supervised regression only has been studied recently [18]. A recent comprehensive review on semi-supervised learning can be found in [21].

Transductive learning is a cousin of semi-supervised learning, which also tries to exploit unlabeled data automatically. The main difference lies in the different

assumptions they hold: transductive learning assumes that the goal is to optimize the generalization ability on only a given test data set, and the unlabeled examples are exactly the test examples [5,14]; semi-supervised learning does not assume a known test set, and the unlabeled examples are not needed to be test examples.

Active learning deals with methods that assume the learner has some control over the input space. In exploiting unlabeled data, it requires that there is an *oracle*, such as a human expert, can be queried for labels of specific instances, with the goal of minimizing the number of queries needed. Here the key is to select the unlabeled example on which the labeling will convey the most helpful information for the learner. There are two major schemes, i.e. *uncertainty sampling* and *committee-based sampling*. Approaches of the former train a single learner and then query the unlabeled examples on which the learner is least confident [6]; while approaches of the latter generate a committee of multiple learners and select the unlabeled examples on which the committee members disagree the most [1,11].

## 2 A Machine Learning View of CBIR

With the rapid increase in the volume of digital image collections, content-based image retrieval (CBIR) has attracted a lot of research interests [13]. The user can pose an example image, i.e. user query, and ask the system to bring out relevant images from the database. A main difficulty here is the gap between high-level semantics and low-level image features, due to the rich content but subjective semantics of an image. *Relevance feedback* has been shown as a powerful tool for bridging this gap [10,15]. In relevance feedback, the user has the option of labeling a few images according to whether they are relevant to the target or not. The labeled images are then given to the CBIR system as complementary queries so that more images relevant to the user query can be retrieved from the database.

In fact, the retrieval engine of a CBIR system can be regarded as a machine learning process, which attempts to train a learner to classify the images in the database as two classes, i.e. positive (relevant) or negative (irrelevant). However, this learning task has something different from traditional supervised learning tasks, which makes it interesting and challenging.

First, few users will be so patient to provide a lot of example images in the retrieval process. Therefore, even with relevance feedback, the number of labeled training examples are still very small. Second, few users will be so patient to take part in a time-consuming interaction process. Therefore, the learning process should meet the real-time requirement. Third, instead of returning a crisp binary classification, the learner is expected to produce a rank of the images. The higher the rank, the more relevant the image. Fourth, in typical supervised learning the concept classes are known in advance, but in CBIR, since an image can be relevant to one query but irrelevant to another, the concept classes are dynamic, cannot be given a priori. The last but not least important, typical

machine learning algorithms regard the positive and negative examples interchangeably and assume that both sets are distributed approximately equally. In CBIR although it is reasonable to assume that all the positive examples belong to the same target class, it is usually not valid to make the same assumption for the negative ones because different negative examples may belong to different irrelevant classes and the small number of negative examples can hardly be representative for all the irrelevant classes.

### 3 Why Exploiting Images in Database?

Section 2 mentioned that in CBIR, even with relevance feedback, the number of example images provided by the user is still very limited. However, there are abundant images existing in the database. Can those images be helpful? Of course.

It is well-known that a main difficulty of CBIR is the gap between high-level semantics and low-level image features, due to the rich content but subjective semantics of an image. This problem can hardly be solved by simply using stronger visual features, but can be released to some degree by using more example images. Usually, the target concept being queried by the user becomes more clear when the user gives more example images. In fact, the relevance feedback mechanism works simply because more example images are given by the user during the feedback process.

Thus, considering the example images as labeled training examples and the images in the database as unlabeled training examples, the CBIR problem resembles what has motivated the research on learning with unlabeled examples. That is, there are a limited number of labeled training examples which are not sufficient for training a strong learner, but there are abundant unlabeled training examples which can be exploited. So, it is evident that techniques of learning with unlabeled data can be used to help improve the retrieval performance.

Note that when the CBIR process is executed on a given database, the task can be mapped to a transductive learning problem since the generalization ability on the given database is concerned; when the CBIR process is executed on an open image source, such as the web, the task can be mapped to a semi-supervised learning problem. On the other hand, since relevance feedback involves human interaction, active learning can be helpful. Thus, CBIR provides a good arena for techniques of learning with unlabeled data.

### 4 Some Results

We have designed some *co-training* style techniques for exploiting unlabeled data in CBIR [16,17].

Co-training was proposed by Blum and Mitchell [2], which has then been studied and extended by many researchers and thus become a popular scheme in learning with unlabeled data. In its original version, co-training trains two classifiers separately on two *sufficient and redundant views*, i.e. two attribute

sets each of which is sufficient for learning and conditionally independent of the other given the class label, and uses the predictions of each classifier on unlabeled examples to augment the training set of the other. Later, variants which do not require sufficient and redundant views have been presented [3,19], and so does an active learning variant [8].

In order to avoid a complicated learning process such that the real-time requirement of CBIR can be met, we [16,17] employ a very simple model to realize two learners which use Minkowski distances with different orders to measure the image similarities. Each learner will give every unlabeled image a rank which is a value between  $-1$  and  $+1$ , where positive/negative means the learner judges the concerned image to be relevant/irrelevant, and the bigger the absolute value of the rank, the stronger the confidence of the learner on its judgement. Then, each learner will choose some unlabeled images to label for the other learner according to the rank information. After that, both the learners are re-trained with the enlarged labeled training sets and each of them will produce a new rank for the unlabeled images. The new ranks generated by the learners can be easily combined, which results in the final rank for every unlabeled image. Then, unlabeled images with top ranks are returned as the retrieval results which are displayed according to descending order of the real value of their ranks. Besides, unlabeled images with bottom absolute ranks (i.e. near 0) are put into a pool, which is then used for the user to give feedback. By using such an active learning scheme, the images labeled by the user in the relevance feedback process can have bigger chance to be the ones that are most helpful in improving the retrieval performance. It has been shown that introducing both semi-supervised learning and active learning into CBIR are beneficial [16,17].

The above approach works with the relevance feedback process, where there are several labeled training examples that can be used. As for the initial retrieval, since there is only one labeled training example, i.e. the user query, exploiting unlabeled examples is more difficult. Such an extreme setting has not been studied before in the area of learning with unlabeled data. In a recent work [20] we have shown that when the images are with textual annotations, even in the initial retrieval, exploiting unlabeled images to improve the retrieval performance is still feasible. The key is to induce some additional labeled training examples by using Kernel Canonical Component Analysis [4] to exploit the correlations between the visual features and textual annotations. Such an approach can be easily generalized to other cases where there is only one labeled training example but the data have two views.

## 5 Conclusion

Techniques of learning with unlabeled data are helpful in diverse machine learning or data mining applications. This paper shows that how they can be helpful in enhancing the performance of CBIR, which exhibits an encouraging new direction for image retrieval research. Note that here we do not claim that the retrieval performance can be boosted to a level which can ‘make the user satisfied’, which is still a long way to go. In fact, what we have claimed is that

the retrieval performance can be enhanced by exploiting the unlabeled images. Moreover, we believe that CBIR can raise many interesting machine learning research topics, the outputs of which will be not only beneficial to CBIR but also be able to generalize to other learning tasks.

## Acknowledgment

This research was partially supported by FANEDD (200343) and NSFC (60325207).

## References

1. N. Abe and H. Mamitsuka. Query learning strategies using boosting and bagging. In *Proceedings of the 15th International Conference on Machine Learning*, pages 1–9, Madison, WI, 1998.
2. A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *Proceedings of the 11th Annual Conference on Computational Learning Theory*, pages 92–100, Madison, WI, 1998.
3. S. Goldman and Y. Zhou. Enhancing supervised learning with unlabeled data. In *Proceedings of the 17th International Conference on Machine Learning*, pages 327–334, San Francisco, CA, 2000.
4. D. R. Hardoon, S. Szedmak, and J. Shawe-Taylor. Canonical correlation analysis: an overview with application to learning methods. *Neural Computation*, 16(12):2639–2664, 2004.
5. T. Joachims. Transductive inference for text classification using support vector machines. In *Proceedings of the 16th International Conference on Machine Learning*, pages 200–209, Bled, Slovenia, 1999.
6. D. Lewis and W. Gale. A sequential algorithm for training text classifiers. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 3–12, Dublin, Ireland, 1994.
7. D. J. Miller and H. S. Uyar. A mixture of experts classifier with learning based on both labelled and unlabelled data. In M. Mozer, M. I. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems 9*, pages 571–577. MIT Press, Cambridge, MA, 1997.
8. I. Muslea, S. Minton, and C. A. Knoblock. Selective sampling with redundant views. In *Proceedings of the 17th National Conference on Artificial Intelligence*, pages 621–626, Austin, TX, 2000.
9. K. Nigam, A. K. McCallum, S. Thrun, and T. Mitchell. Text classification from labeled and unlabeled documents using EM. *Machine Learning*, 39(2-3):103–134, 2000.
10. Y. Rui, T. S. Huang, M. Ortega, and S. Mehrotra. Relevance feedback: a power tool for interactive content-based image retrieval. *IEEE Transactions on Circuits and Systems for Video Technology*, 8(5):644–655, 1998.
11. H. Seung, M. Opper, and H. Sompolinsky. Query by committee. In *Proceedings of the 5th ACM Workshop on Computational Learning Theory*, pages 287–294, Pittsburgh, PA, 1992.
12. B. Shahshahani and D. Landgrebe. The effect of unlabeled samples in reducing the small sample size problem and mitigating the hughes phenomenon. *IEEE Transactions on Geoscience and Remote Sensing*, 32(5):1087–1095, 1994.

13. A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12):1349–1380, 2000.
14. V. N. Vapnik. *Statistical Learning Theory*. Wiley, New York, 1998.
15. X. S. Zhou and T. S. Huang. Relevance feedback in image retrieval: a comprehensive review. *Multimedia Systems*, 8(6):536–544, 2003.
16. Z.-H. Zhou, K.-J. Chen, and H.-B. Dai. Enhancing relevance feedback in image retrieval using unlabeled data. *ACM Transactions on Information Systems*, 24(2), 2006.
17. Z.-H. Zhou, K.-J. Chen, and Y. Jiang. Exploiting unlabeled data in content-based image retrieval. In *Proceedings of the 15th European Conference on Machine Learning*, pages 525–536, Pisa, Italy, 2004.
18. Z.-H. Zhou and M. Li. Semi-supervised learning with co-training. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence*, pages 908–913, Edinburgh, Scotland, 2005.
19. Z.-H. Zhou and M. Li. Tri-training: exploiting unlabeled data using three classifiers. *IEEE Transactions on Knowledge and Data Engineering*, 17(11):1529–1541, 2005.
20. Z.-H. Zhou, D.-C. Zhan, and Q. Yang. Semi-supervised learning with a single labeled example. *Unpublished manuscript*.
21. X. Zhu. Semi-supervised learning literature survey. Technical Report 1530, Department of Computer Sciences, University of Wisconsin at Madison, Madison, WI, 2005. [http://www.cs.wisc.edu/~jerryzhu/pub/ssl\\_survey.pdf](http://www.cs.wisc.edu/~jerryzhu/pub/ssl_survey.pdf).

# Learning as Abductive Deliberations

Budhitama Subagdja<sup>1</sup>, Iyad Rahwan<sup>2</sup>, and Liz Sonenberg<sup>1</sup>

<sup>1</sup> Department of Information Systems, University of Melbourne

<sup>2</sup> Institute of Informatics, The British University in Dubai, (Fellow) School of Informatics, University of Edinburgh, UK

**Abstract.** This paper explains an architecture for a BDI agent that can learn based on its own experience. The learning is conducted through explicit procedural knowledge or plans in a goal-directed manner. The learning is described by encoding *abductions* within the deliberation processes. With this model, the agent is capable of modifying its own plans on the run. We demonstrate that by abducting some complex structures of plan, the agent can also acquire complex structures of knowledge about its interaction with the environment.

## 1 Introduction

The BDI (Beliefs, Desires, Intentions) agent model [10] is a design framework commonly used in developing agents that behave both deliberatively and reactively in a complex changing environment. The main principle is to use explicit representations of the agents' own mental attitudes (in terms of attributes such as beliefs, desires, and intentions) to direct their actions and decision of choosing the appropriate predefined plan. To develop the system, the designer would define some initial mental conditions and describing some plans explicitly which correspond to the agents behavior in a repository of plans. Variability in behavior can be attained by the process of deliberation.

However, it is always possible that unforeseen conditions require some modification of the prescribed plans or knowledge instead of just alternating one plan after another. Although, the exhibition of the behavior can be adaptive in a reactive way, plans for directing or guiding the behavior in a BDI agent are still fixed in advance of the system execution. Most existing BDI frameworks are still incapable of modifying plans or *recipes* for actions at runtime.

In this paper, we present a new model of learning in BDI agents. We use *meta-level plans*, expressed in general programming constructs, to enable the agent to specify learning and deliberation steps explicitly. This enables the agent to introspectively monitor its own mental state and update its plans at runtime. The learning is regarded as a kind of deliberation process in which the agent makes plausible hypotheses about expected outcomes and creates (or modifies) plans if the hypotheses are proven. This kind of process is also known as *abduction*, a term which was coined by C.S. Peirce [6]. In this case, the agent is not just selecting the best option available but also expecting useful knowledge to be acquired if the selection fails.

This work advances the state of the art by combining the strengths of learning and BDI agent frameworks in a rich language for describing deliberation processes. In particular, our approach enables domain experts to specify learning processes and strategies explicitly, while still benefiting from procedural domain knowledge expressed in plan recipes (as opposed to generating and learning plans from scratch).

The remainder of this paper is structured as follows. the next section explain the architecture of a BDI agent. The section also describes the concept of deliberation processes as meta-level plans which can accommodate *abductions*. Section 3 then explain how learning can be described explicitly in terms of meta-level plans and deliberation processes. In that section we describe some primitives for learning and some examples of generic strategies for the experience-based plan construction. Section 4 illustrates the characteristic of the learning approach from a case study. Section 5 discusses some related works on learning intentional agents. Finally, the last section concludes the paper.

## 2 BDI Agent Architecture

The BDI architecture works as an interpreter interacting with different data structures. In PRS [4] as the commonly used BDI implementation model, there are four different types of data structure. Firstly, *beliefs* or *belief base* ( $\mathcal{B}$ ) correspond to a model or knowledge about the world which can be updated directly by events captured on the sensors. Secondly, the agent's *desires* or *goals* ( $\mathcal{G}$ ) correspond to conditions the agent wants to fulfill. The *desires* invoke finding ways to achieve them and select one (or some) to act upon. Thirdly, the selected ways to be committed for execution are the *intentions* ( $\Gamma$ ). Lastly, the knowledge of how to achieve certain desires or goals are stored in the *plans* or the *plan library*.

The common process of a BDI interpreter that drives the agent's behavior is an iteration of steps like updating beliefs based on observation in the world, deciding what intention to achieve, choosing a plan to achieve intentions, and executing the plan [12]. The interpreter goes through a control loop which consists of observation, intention filtering, and plan selection. The adopted intention is committed for execution to its end. If something goes wrong with the intention, the agent can reconsider its intention, select another plan as an alternative or just drop the intention and select another intention.

In PRS-like agents, the loop may produce a hierarchical structure of intentions. A selected intention may invoke further deliberations which produce other intentions having sub-ordinate relations with the former one. This hierarchical structure is also called the *intention structure*. The intention structure represents a stack structure consisting of goals, subgoals, and their intentions. The intention structure maintains some information about the state of the agent choices and actions, limits the number of choices to be considered at a deliberation moment, thus reducing computational complexity at every cycle. By using this structure, a goal can be broken down further to be more specific while the agent behaves reactively to changes in the environment.

## 2.1 Plans and Intentions

A plan represents procedural knowledge or know-how. As a knowledge for accomplishing a task, a plan would be used as a *recipe* which guides an agent in its decision making process, hence reducing search through alternative solutions [9]. In classical STRIPS planning [2], a plan consists of a set of operators or actions each with attributes like a list of *preconditions*, an *add list*, and a *delete list*.

**Definition 1.** *An action  $\alpha$  is a tuple of  $\langle A_\alpha, P_\alpha, \Delta_\alpha, \Sigma_\alpha \rangle$  in which  $A_\alpha$  is the action name;  $P_\alpha$  is a list of conditions that must be believed to be true prior to the execution of  $\alpha$ ;  $\Delta_\alpha$  is a list of conditions that must be believed to be false after the performance of  $\alpha$ ; and  $\Sigma_\alpha$  are those that are believed to be true after the performance of  $\alpha$ . Conditions are expressed as literals which can be propositions or predicate logic statements.*

A plan can be considered as an encapsulated description of actions with its consequences and contexts. It may represent just a single action or it can describe a complex relationship between actions. Similar to an action, a plan also has contextual descriptions like preconditions and effects (add or delete lists). In addition, a plan can also have attributes like a trigger (goal) and a body that describes relationships between actions.

**Definition 2.** *A plan  $\pi$  can be defined as a tuple  $\langle \varphi_\pi, P_\pi, \Sigma_\pi, \Delta_\pi, B_\pi \rangle$  where  $P_\pi$ ,  $\Sigma_\pi$ , and  $\Delta_\pi$  are respectively the preconditions, add list, and delete list which have the same meaning as the corresponding symbols in the action definition above. The trigger  $\varphi_\pi$  is the goal that triggers the activation of the plan. The plan body  $B_\pi$  describes actions and their relationships.*

The plan goal states the thing that is wanted or desired by executing the plan. There are two types of goals: **achieve** and **perform**. A plan with an **achieve** goal says that a condition stated in the goal will hold or be true after performing actions described in the plan body. A **perform** goal, on the other hand, tells that actions described in the goal will be performed if the plan is executed. Actions described in a **perform** goal or a plan body are represented in a composite action.

**Definition 3.** *A composite action  $\tau_C[\phi_1, \dots, \phi_n]$  states the relationship between actions.  $\tau_C$  is the type of the relation, in which  $\tau_C \in \Upsilon_C$  and  $\phi_i$  can be an action, a proposition, or another composite action forming a nested structure of actions relationship. If  $\iota_\alpha$  is a composite action,  $\nu \leftarrow \iota_\alpha$  is an assignment of the result of the action  $\iota_\alpha$  to the variable  $\nu$ .*

There can be many types of structure in  $\Upsilon_C$ . Due to space limitations, table 1 only describes some of them which seem to be relevant and important. A variable, once bound to a value, can be used for a later purpose through variables in term parameters. For example, the composite action  $\text{seq}[\text{do}[X \leftarrow \text{select\_object}], \text{do}[\text{grasp}(X)]]$  states that an object  $X$  is selected and then grasped. The object value which is bounded by the variable  $X$  as a result of the selection action is fed into the action **grasp**.

In the deliberation cycle of the BDI interpreter a plan is selected from the plan library based on current goals and intentions. The selected plan is instantiated

**Table 1.** Action structures and relationships

Relation type	Description
<b>do</b> [ $\alpha$ ]	execute a single action $\alpha$
<b>confirm</b> [ $c$ ]	confirm if condition $c$ is true in the agent's beliefs
<b>conclude</b> [ $c$ ]	conclude that the condition $c$ is true by asserting it to the agent's beliefs
<b>wait</b> [ $c$ ]	wait until the condition $c$ is true
<b>subgoal</b> [ $\varphi$ ]	post the goal $\varphi$ as a subgoal
<b>seq</b> [ $\beta_1, \dots, \beta_n$ ]	execute substructures $\beta_1$ to $\beta_n$ consecutively
<b>seq-choices</b> [ $\beta_1, \dots, \beta_n$ ]	try to execute substructures in the list from $\beta_1$ to $\beta_n$ consecutively until a successful execution of one of them
<b>cycle</b> [ $\beta_1, \dots, \beta_n, <$ until $c$ $>$ ]	iteratively execute all substructures in the list based on the order and stop until a condition $c$ (optional) is true

and incorporated as an intention. The intention is put on the intention structure before it is executed later on. A plan instance or an intention stores an index that locates the current selected goal or action in the corresponding plan body. It also maintains information about variable bindings and states of the intention. An intention can be in a **scheduled**, **succeeds**, **fails**, **pushed**, or **wait** state.

If the plan body of a plan instance has a nested structure, then the substructure of the composite action becomes a new intention which is concatenated at the intention of that plan instance. This is also conducted for a composite action that posts a subgoal. Another plan instance for achieving the subgoal will be concatenated at that location in the intention structure.

## 2.2 Meta-level Plans and Abductions

In the previous section we described the model of plans and intentions in a BDI agent architecture. In this section, we explain the use of meta-level plans for controlling the deliberation. This section also shows how meta-level plans can leverage the deliberation process with abductions.

The original PRS model assumes that the deliberation process is handled by the use of meta-level plans [5]. The instance of meta-level plans can obtain information from the intention structure and change it at runtime. A meta-level plan for the deliberation process can be characterized as a plan which contains some *meta-actions* or actions that deal with goals, intentions, and plans. For example, the composite action described below shows some parts of the deliberation process.

```
cycle[
  do[observe],  $G \leftarrow \text{do}[\text{consider\_options}]$ ,  $I \leftarrow \text{do}[\text{filter\_options}(G)]$ ,
   $P \leftarrow \text{do}[\text{select\_plan}(I)]$ ,  $\text{do}[\text{intend}(I, P)]$ ]
```

This structure of composite actions can be put initially in the intention structure. It works as an infinite loop of **observe** for updating belief, **consider\_options** for generating options, **filter\_options** for selecting intentions, **select\_plan** for selecting a plan instance, and **intend** that insert the selected plan to its corresponding intention and put them on the intention structure. Objects passed or

exchanged between actions are goal options ( $G$ ), selected intentions ( $I$ ), and a plan ( $P$ ). The intention execution and reconsideration parts of the loop are skipped for simplification and it is assumed that executing the intention in the intention structure is done by the interpreter as a default process.

Based on the process of deliberation and execution, it is possible to say that the agent decides a plan and selects an action based on its beliefs and goals. We assume that if the agent has a plan for achieving a goal, it means also that the agent believes that executing the actions described in the plan will bring about the goal. This kind of process of selecting and adopting a plan instance can be regarded as a deductive inference. When there is a failure in executing the plan, a re-deliberation or re-selection process can be conducted from the beginning with a refreshed condition of beliefs. A more sophisticated technique is retaining the history of the past failures so that a failed plan instance will never be chosen for the second time.

In this paper, we suggest that *abductions* can incorporate the deliberation process so that the agent may not just re-deliberate to deal with failures and changes but also anticipate what would happen. The agent tries to develop explanations about possible failures while it tries to achieve the goal. In our model, the abduction is activated by the deliberation. The agent decides not just the goal and the intention to fulfill but also some proofs of hypothetical failures. The composite action for the deliberation cycle becomes as follows:

```
cycle[
  do[observe],  $G \leftarrow [\text{do[consider\_options]}]$ ,  $I \leftarrow \text{do[filter\_options}(G)\text{]}$ ,
   $P \leftarrow \text{do[select\_plan}(I)\text{]}$ ,  $H \leftarrow \text{do[make\_hypothesis}(G, I)\text{]}$ ,
   $P' \leftarrow \text{do[select\_testing\_plan}(H, I)\text{]}$ , do[intend}(I, P)], do[intend}(H, P')]]
```

The action **make\_hypothesis** generates hypotheses based on prescribed beliefs about what kind of situation would come up. The hypothesis made can be expressed as a composite action describing the sequence of events or actions that would happen. The plan for testing the hypothesis can be described as a meta-level plan. This meta-level plan involves types of actions that can capture changes in the agent's mental state (e.g. **wait**, **confirm**). The successful execution of testing will produce or revise a belief about something that is hypothesized.

When a test to prove a hypothesis fails, it will just be dropped or removed from the intention structure just like a normal plan execution. The testing plan will be re-activated in the next deliberation cycle if the same goal is still needed to be achieved. The testing plan may also update the beliefs about quality, preference, or confidence levels of a plan so that the plan will have a greater chance of being picked up by the action **select\_plan** in the next cycle of the deliberation loop.

Different proving or testing strategies can be used for testing different hypotheses. For example, a hypothesis testing plan can have a composite action like the following: **seq[wait[done(I)], confirm[success(I)], do[assert\_belief(H, I)]]**. This composite action can be used to prove that the intention  $I$  will be executed successfully so that a new belief about the hypothesis  $H$  in relation with  $I$  can be asserted. The action **wait** waits for  $I$  until its finish before confirming its success.

In another case, a different testing strategy might have a more complex composite action like the following:

```
seq-choices[cycle[wait[done( $I$ )], confirm[not success( $I$ )],  

 $H \leftarrow \text{do}[\text{append}(H, I)], \text{do}[\text{assert\_belief}(H, I)]]$ ]
```

This composite action can be used to prove that actions in the intention  $I$  will eventually succeed or reach the goal if they are repeated for a certain number of times. The first branch of the **seq-choices** captures an unsuccessful attempt and updates the hypothesis with an additional step. The second branch is executed when a successful attempt is found and a belief about the repetition structure can be asserted. This hypothesis testing strategy still has a flaw in dealing with a single alternative of action only. An infinite loop might be produced if the goal can not be attained. However, the structure can trivially be amended by inserting some **confirm** actions on both branches to test if the length of the repetition exceeds a certain limit.

To deal with complex situations and problems, different abduction plans can be given to test different possible structures of actions upon several attempts of goal achievements. The abduction plans can be provided by the domain expert or the agent designer as heuristics for acquiring knowledge.

### 3 Representing Learning Processes

Learning in BDI agents can be defined as abduction-deliberation processes which can result in the improvement of the agent's performance. The approach of improvement suggested in this paper is by modifying or generating plans in the plan library. The hypotheses confirmed through the abduction process are candidate plans. The steps of confirmation are followed by a plan generation or modification. For example, the following composite action can be considered as the body of a learning plan:

```
seq-choices[cycle[wait[done( $I$ )], confirm[not success( $I$ )],  

 $H \leftarrow \text{do}[\text{add\_plan\_step}(H, I)], \text{do}[\text{create\_plan}(H, I)]]$ ]
```

This composite action is similar with the example of a composite action for hypothesis testing mentioned above. It captures repetitive actions for achieving the goal. However, the end result of the testing process is a new plan. In this case, the hypothesis  $H$  is a template of the possible plan. The learning plan can be said as trying to confirm that there is a sequence of repetitive actions that eventually reaches the goal. If a repetition of actions is confirmed, a new plan with a repetition or a sequence structure can be asserted.

### 4 Case Study

This section shows some examples of learning plans and the development of the agent's knowledge when the agent is given certain tasks and is situated in

a certain environment. In order to implement the experiment for studying the characteristic of the learning agent, we have developed a special type of a BDI interpreter which supports introspective plans monitoring and modification at runtime.

#### 4.1 The Rat's World

The Rat's World is an implemented simulation inspired by the psychological experiment of operant conditioning. An artificial rat agent is put on a designated place with some desires of getting some rewards (metaphorically some cheese). To get the reward the agent must select (press) some buttons in a particular order. Assume there are two buttons each with different colors (let say black and white buttons).

If the appropriate order has been setup so that the reward can be obtained by firstly pressing the black button followed by the white one, the rat can learn the combination by pursuing several trials of the same situation and converge to the right sequence (Some reinforcement learning algorithms like Q-learning can learn this kind of task very well). However, a simple modification can make this problem non-trivial. In particular, the situation becomes complicated when the position of the buttons is randomly swapped for every trial.

At one moment, the agent has beliefs about the buttons' positions and its own last action. Following the Markovian model of decision processes, these beliefs represent a state. Let's say, initially, the agent has the following belief based on its initial perception: `button_pos(black, white)` which states that the black button is on the left and the white one is on the right. The predicate `last_act(A)` is used to refer to the last action taken, and is added after the agent do the first action. The `A` can be `press(left)` for pressing the button on the left or `press(right)` for pressing the one on the right.

The agent is provided with some initial plans for getting the reward. One plan contains actions like `do[press(left)]` and the other has `do[press(right)]`. With these initial plans, a deliberation process will cause the agent to press the left or the right button in a random fashion to get the reward. A simple learning plan can be made which is triggered by a drop in its performance level (high rate of failures). The body of the learning plan (hypothesis testing) can be as follows

```

seq[  

  wait[done(I)], S  $\leftarrow$  do[observe], seq-choices[  

    seq[confirm[success(I)], P  $\leftarrow$  do[create_plan(I, S)]],  

    seq[wait[done(I')], confirm[success(I')], P  $\leftarrow$  do[create_plan(I, S)],  

      G  $\leftarrow$  do[obtain_goal(I)], P  $\leftarrow$  do[add_plan_step(P, subgoal(G))]  

    ]]  

  ], do[generate_plan(P)]
]

```

**(Learning Plan 1)**

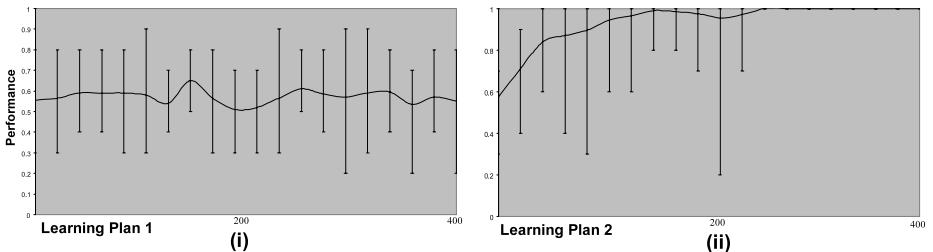
The learning plan monitors events produced by the intention *I* on the intention structure. When *I* succeeds straightaway, a new plan is created by the action `create_plan` with the instance of actions in the intention *I* as its plan

body and the observed state  $S$  as its precondition. Otherwise, it waits for another intention  $I'$  that succeeds straightaway, and a new plan is created with the plan instance of the intention  $I$ , the precondition  $S$ , but a subgoal posting action is appended at the end. Learning Plan 1 produced two types of plan. One type of plan maps a belief state directly to an action. The other type maps a belief state to an action that reach an intermediate state and post the same subgoal recursively. For example, one plan produced has only the following structure in its body `do[press(right)]` and another one with the following structure `seq[do[press(right)], subgoal[get_reward]]`. Each generated plan has a precondition. For example,

`button_pos(black, white)` and `last_act(do[press(right)])`.

The experiment conducted has shown that the learning plan described above is not effective in dealing with the dynamic situation. This is because the agent can not distinguish some observed states when the buttons stay still from states where the buttons have just been swapped. The agent only relies on chances and the probability distribution of the learnt plans in dealing with uncertainties.

Figure 1(i) shows that the performance of learning with Learning Plan 1 is not much different than without learning at all. From 400 learning trials for each 20 cases, the performance on average still stays just slightly above 50% chances with a relatively high level of variability. The performance level is measured by the rate of successful attempts (getting the rewards).



**Fig. 1.** Performance of agent in the Rat's World domain (i) with Learning Plan 1. (ii) with Learning Plan 2

We modified the learning plan by encoding a different hypothesis. The agent is made to wait one more action before producing a sequence of actions. The composite action of the learning plan above is modified as follows

```

seq[
  S ← do[observe], wait[done(I)], wait[done(I')]
  confirm[I ≠ I'], confirm[fails(I)], confirm[success(I')],
  P ← do[create_plan(I, S), P ← do[add_plan_step(P, planbody(I'))],
  do[generate_plan(P)]
]

```

(Learning Plan 2)

This learning plan waits for two consecutive execution of actions from intentions  $I$  and  $I'$ . If both actions are different and the first action failed while the

second one succeeded, then a new plan is generated with the two consecutive actions in the plan body and the observational state as the precondition. The result of the learning is one type of plan with a sequence of different actions as follows  $\text{seq}[\text{do}[\text{press(left)}], \text{do}[\text{press(right)}]]$ . The plan is also provided with appropriate preconditions based on observations. The experiment using Learning Plan 2 produced much better results. Figure 1(ii) shows that successful attempts raise the performance quickly to maximum values. In all cases, it is demonstrated that the performance always reaches the maximum. This can happen because the right interaction between the agent and the buttons can be modeled as a composite action. By hypothesizing a pattern of consecutive actions underlying the right interaction, a significant number of combinations of plan structures to be searched can be pruned.

The experiments in the Rat’s World domain have clearly shown that a simple mapping between an observational state with a single action is not enough to make the agent learn the right model of interaction in a changing situation. The agent should first make assumptions about the model of its interaction with the environment. It is also indicated that applying different patterns of hypotheses influences the capability of the agent to learn something from the environment. Using complex patterns of hypotheses would also make the agent learn complex things.

## 5 Related Works

There is some previous work on making BDI agents learn [1,3] by making learning as a separate process conducted by separate modules. For example, Hernandez et al. [3] apply an inductive decision-tree learning algorithm that learns new plans by feeding in logged events produced by the BDI engine to the learning program. In contrast, the learning processes suggested in this paper are mainly heuristics or knowledge that describe learning and abduction as parts of the BDI architecture. Other works have considered learning as parts of the BDI mechanism [7,8,11], however they still assume the creation of plans based on direct mappings of observational states to corresponding plans or actions. As demonstrated in the last section, this kind of plan creation may lead to the *perceptual aliasing* problem in which the agent would not be able to distinguish one state from another.

The advantage of putting the learning as part of the explicit knowledge in the BDI architecture is that it enables the agent developer to specify learning processes and strategies from domain or expert knowledge easily. In any case, the main feature of the BDI agent architecture is that the behavior of the agent is driven mainly by the procedural domain knowledge which can be prescribed as plan recipes as opposed to generating plans from scratch.

## 6 Conclusion

In this paper, we describe a model of learning in the BDI agent architecture. We use meta-level plans to program learning as an explicit part of the agent’s

behavior. The meta-level plans enable the agent to introspectively monitor their own mental conditions and update their plans at runtime. The learning can be described as a process of abduction that tries to confirm the occurrences of structures of plans. The experiments conducted have shown that the agent needs to make assumptions about its possible interactions with the environment. Direct observations might be insufficient to learn the appropriate model.

Although the heuristics shown in this paper are problem specific, they can still be useful in a range of different situations. By providing a set of different types of heuristics for different classes of problems, the learning might cover various domains of application. Moreover, the learning can be multi-strategic and reactive to the change in the environment. However, further work is still needed for seeking the appropriate set of heuristics so that the approach can be practically useful.

## References

1. C. F. E. Cindy Olivia, Chee-Fon Chang and A. K. Ghose. Case-based BDI agents: an effective approach for intelligent search on the world wide web. In *Proceedings of the AAAI-99 Spring Symposium on Intelligent Agents in Cyberspace Stanford University, USA*, 1999.
2. R. Fikes and N. Nilsson. STRIPS: A new approach to the application of theorem proving to problem solving. In *Artificial Intelligence*, volume 2, pages 189–208. 1971.
3. A. G. Hernandez, A. E. Segruchini, and H. Soldano. BDI multiagent learning based on first-order induction of logical decision trees. In S. O. N. Zhong, J. Liu and J. Bradshaw, editors, *Intelligent Agent Technology: Research and Development*. World Scientific, New Jersey, 2001.
4. F. Ingrand, M. Georgeff, and A. Rao. An architecture for real-time reasoning and system control. In *IEEE Expert*, volume 7(6), pages 34–44. 1992.
5. F. F. Ingrand and M. P. Georgeff. Managing deliberation and reasoning in real-time AI systems. In *Proceedings of the DARPA Workshop on Innovative Approaches to Planning*, San Diego, California, 1990.
6. L. Magnani. *Abduction, Reason, and Science: Processes of Discovery and Explanation*. Kluwer Academic/Plenum Publishers, New York, 2001.
7. E. Norling. Learning to notice: Adaptive models of human operators. In *Second International Workshop on Learning Agents*, Montreal, 2001.
8. E. Norling. Folk psychology for human modelling: Extending the BDI paradigm. In *Proc. of the Third Int. Joint Conf. on Autonomous Agents and Multi Agent Systems (AAMAS-04)*, New York, NY, July 2004.
9. A. S. Rao. A unified view of plans as recipes. In *Contemporary Action Theory*. Kluwer Academic, Netherlands, 1997.
10. A. S. Rao and M. P. Georgeff. BDI agents: From theory to practice. In *Proceedings of the First International Conference on Multi-Agent Systems (ICMAS-95)*. San Francisco, 1995.
11. C. Sioutis and N. Ichalkaranje. Cognitive hybrid reasoning intelligence agent system. In *Knowledge-Based Intelligent Information and Engineering Systems, 9th International Conference, KES 2005, Melbourne, Australia*, volume 3682 of *LNAI*, pages 838–843. Springer, 2005.
12. M. Wooldridge. *Reasoning about Rational Agents*. MIT Press, Cambridge, 2000.

# Using a Constructive Interactive Activation and Competition Neural Network to Construct a Situated Agent's Experience

Wei Peng and John S. Gero

Key Centre of Design Computing and Cognition  
University of Sydney, NSW 2006, Australia  
[{wpeng, john}@arch.usyd.edu.au](mailto:{wpeng, john}@arch.usyd.edu.au)  
<http://www.arch.usyd.edu.au/kcdc/>

**Abstract.** This paper presents an approach that uses a Constructive Interactive Activation and Competition (CIAC) neural network to model a situated agent's experience. It demonstrates an implemented situated agent and its learning mechanisms. Experiments add to the understanding of how the agent learns from its interactions with the environment. The agent can develop knowledge structures and their intentional descriptions (conceptual knowledge) specific to what it is confronted with – its experience. This research is presented within the design optimization domain.

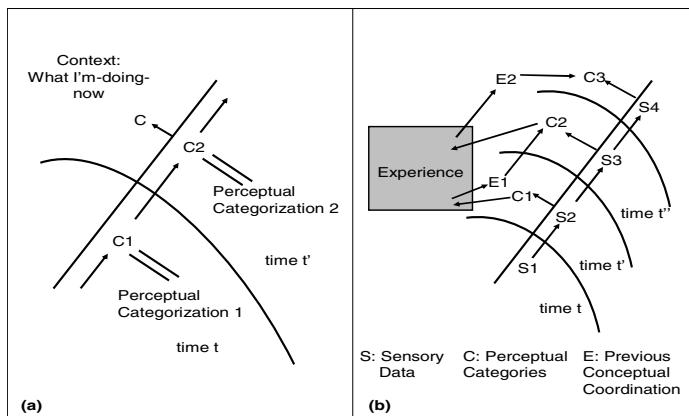
## 1 Introduction

Experience is defined as the accumulation of knowledge or skill that results from direct participation in events or activities [15]. Learning is a process whereby knowledge is created through the transformation of experience [6], [8]. Experience plays a key role in a learning process. Learning, also known as knowledge acquisition, has been widely explored by many artificial intelligence (AI) and machine learning researchers. A broad spectrum of approaches has been developed, including inductive learning methods, explanation-based learning approaches and connectionist algorithms. Another theory of learning emerged from cognitive science domain. “Situatedness” [1], [4] and “situated learning” [9], [13] emphasize the role of social interactions in learning. An agent’s memory can be regarded as a learning process. The notion of “constructive memory” contradicts many views of knowledge as being unrelated to either its locus or application [5]. A constructive memory model [5], [10] provides a conceptual framework for us to utilize the concept of “situatedness” in a software agent. Learning takes place when a learner interacts with, or is stimulated by, an environment [6]. An agent relates its own experience and gives meanings to a situation [7]. These two theories are not incompatible, but complementary, with both addressing the same issue at different levels. An agent that is designed to be situated at a conceptual level can still be implemented using various machine learners. In this paper, we describe how to model a situated agent’s experience using a Constructive Interactive Activation and Competition (CAIC) neural network.

This situated agent is applied in the design optimization domain. Design optimization is concerned with identifying optimal design solutions which meet design objectives while conforming to design constraints. The design optimization process involves some tasks that are both knowledge-intensive and error-prone. Such tasks include problem formulation, algorithm selection and the use of heuristics to improve efficiency of the optimization process. Choosing a suitable optimizer becomes the bottleneck of a design optimization process. Designers rely on their experience to carry out this task. Such a manual process may result in a sub-optimal design solution and hence an inefficient design. Our objective is to construct a computational model which is able to capture the knowledge of using the design optimization tool, and as a consequence, can aid the tool's future use in supporting design. For example, a designer working on optimizing a hospital layout may find that a certain optimizer is more efficient in solving the problem applied. As the same or other designers tackle a similar design task, the same tool constructs memories of a design situation and anticipates the tool's potential use. It can therefore offer helps to designers in their interactions in designing even before they require. Our approach is to use a situated agent that wraps around an existing design optimization tool. A user accesses a design tool via this wrapper, where a situated agent senses the events performed by that user and learns new concepts from the user's interactions with it.

## 2 Situated Agency

Software agents are intentional systems that work autonomously and interact with environments in selecting actions to achieve goals [14]. A situated agent is the software that is founded on the notion of “situatedness”.



**Fig. 1.** (a) shows the conceptual knowledge as a higher order categorization of a sequence (after Fig. 1.6 of Clancey [2]). (b) illustrates a situated learning scenario.

Situatedness involves both the context and the observer's experiences and the interactions between them. Situatedness [1] holds that “where you are when you do what you do matters” [4]. Conceptual knowledge can be learned by taking account of

how an agent orders its experience in time, which is referred as conceptual coordination [2], Fig. 1.

Conceptual knowledge is a function of previously organized perceptual categories and what subsequently occurs, Fig. 1(a). It is generally formed by holding active a categorization that previously occurred (C1) and relating it to an active categorization C2 [2]. Fig. 1(b) illustrates a scenario of a situated concept learning process in which sensory data is augmented into a Gestalt whole. Perceptual category C1 groups sensory sequence “S1 → S2” and activates the agent’s experience to obtain similar organizations. The agent’s experiential response (E1) represents the agent’s hypotheses about what would happen later in the environment. The agent constructs E1 with environmental changes (S3) into current perceptual category C2. This construction involves a validation process in which environmental changes are matched with the agent’s hypothesis. “Valid” means the environmental changes are consistent with the agent’s projection of such changes from a previous time. The grounding process then reinforces a valid experience. For invalid expectations, the agent updates its perceptual category (C2) with the latest environmental changes.

A situated agent contains sensors, effectors, experience and a concept formation engine, which consists of a perceptor, a cue\_Maker, a conceptor, a hypothesizer, a validator and related processes. Sensors gather events from the environment. These events include key strokes of objective functions, the users’ selections of design optimization algorithms, etc. Sense-data takes the form of a sequence of actions and their initial descriptions. For instance, sense-data can be expressed as:

$$S(t) = \{ \dots, "click on objective function text field", \text{key stroke of } "x", "(", "1", ")", "+", "x", "(", "2", ")" \dots \} \quad (1)$$

The perceptor processes sense-data and groups them into multimodal percepts, which are intermediate data structures illustrating environment states at a particular time. Percepts are structured as triplets:

$$P(t) = (\text{Object}, \text{Property}, \text{Values of properties}) \quad (2)$$

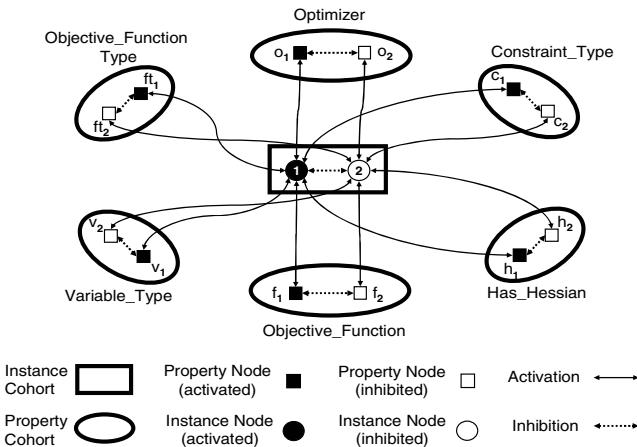
For example, a perceptual data  $P_1$  can be described as (Objective Function Object, Objective\_Function, “ $x(1)+x(2)$ ”).

The cue\_Maker generates cues that can be used to activate the agent’s experience. The conceptor categorizes the agent experience to form concepts. Concepts attach meanings to percepts. The hypothesizer generates hypotheses from the learned concepts. This is where reinterpretation takes place in allowing the agent to learn in a “trial and error” manner. The validator pulls information from the environment and examines whether the environmental changes are consistent with the agent’s responses. An agent needs to validate its hypotheses in interactions to locate a suitable concept for the current situation. An effector is the unit via which the agent brings changes to environments through its actions.

The agent’s experience is structured as a Constructive Interactive Activation and Competition (CIAC) neural network, in which we extend a basic IAC network [11] to accommodate the concept learning process, Fig. 2. An IAC consists of two basic nodes: instance node and property node. The instance node has inhibitory connections

to other instance nodes and excitatory connections to the relevant property nodes. The property nodes encode the special characteristics of an individual instance [12]. Property nodes are grouped into cohorts of mutually exclusive values [12]. Each property node represents the perceptual level experience which is processed from sensory data.

Instance nodes along with the related property nodes describe an instance of a concept. Knowledge is extracted from the network by activating one or more of the nodes and then allowing the excitation and inhibition processes to reach equilibrium [12]. In Fig. 2, the shaded instance node (Ins-1) and related shaded property nodes presents a context addressable memory cued from an environment stimulus, e.g. [Objective\_Function,  $f_1$ ]. Such a response is a dynamic construction in the sense that when environment stimuli change, the agent develops adapted knowledge. This organized experience grounds by weight adaptation and constructive learning.



**Fig. 2.** A CIAC neural network as a representation of the agent's experience; Property nodes are labeled by their value, e.g.  $f_1$  represents a property node in the Objective\_Function cohort with objective function value  $f_1$

### 3 Constructing a Situated Agent's Experience

In this section, we discuss how to construct a situated agent's experience with a CIAC neural net. Table 1 illustrates the formulas that are used to compute network input value ( $N_e$ ) and activation value ( $A_c$ ) for each nodes during activation and competition phrase. Table 1 also describes the formula that is applied to adjust the weights of each excitatory connection of the valid concept during the grounding via weight adaptation process, so that those nodes that fired together become more strongly connected. Weight adaptation is formulated similar to a Hebbian-like learning mechanism [12].

The pseudo code below (Table 2) presents procedures for constructing a situated agent's experience with a CIAC neural network. It shows the relationships between

**Table 1.** Major formulas applied in the CIAC neural net

Items	Formulas
Network Input Values ( $N_e$ ): -- the activations of each of the neurons can be regarded as the “degree of belief” in that hypothesis	$N_e = \mu E + \sum_{i=1}^n W_i A_i$ $\mu$ : excitatory gain for initial network stimulus, set to 4.0 $E$ : initial network stimulus, default 1.0 $W_i$ : inbound weights for a neuron $A_i$ : activation value for each inbound weight $n$ : number of neurons in a network
Activation Values ( $A_c$ ): -- the tendency for units connected by negative weights to turn each other off -- a unit that begins with a small advantage “wins” the competition and becomes active at the expense of the other units in the pool in the end [3].	<i>If</i> $N_e > 0$ $A_c = A_{c-1} + \ell [(A_{\max} - A_{c-1}) N_e - \varphi(A_{c-1} - R)]$ <i>else</i> $A_c = A_{c-1} + \ell [(A_{c-1} - A_{\min}) N_e - \varphi(A_{c-1} - R)]$ $A_c$ : the activation value for each neuron at current cycle $A_{c-1}$ : the activation value for that node at previous cycle $N_e$ : the net input for each node $A_{\max}$ : the maximum permitted activation value, set to 1.0 $A_{\min}$ : the minimum permitted activation value, set to -0.2 $R$ : the initial resting activation value, default -0.1 $\varphi$ : the decay factor, default 1.0 $\ell$ : the learning rate, default 0.1;
Weight Adaptation ( $W_n$ ): -- the process that verifies the usefulness of a related experience in current situation. -- similar to a Hebbian-like learning mechanism	<i>If</i> $W_o > 0$ $W_n = W_o + \ell (W_{\max} - W_o) A_i A_j - \delta W_o$ <i>else</i> $W_n = W_o$ $W_o$ : the weight value before weight-adaptation $W_n$ : the weight value after weight-adaptation $\ell$ : the learning rate, default 0.1 $W_{\max}$ : the maximum permitted weight value, set to 1 $A_i, A_j$ : activation values for neuron $i$ and $j$ $\delta$ : the weight decay factor, set to 0.1

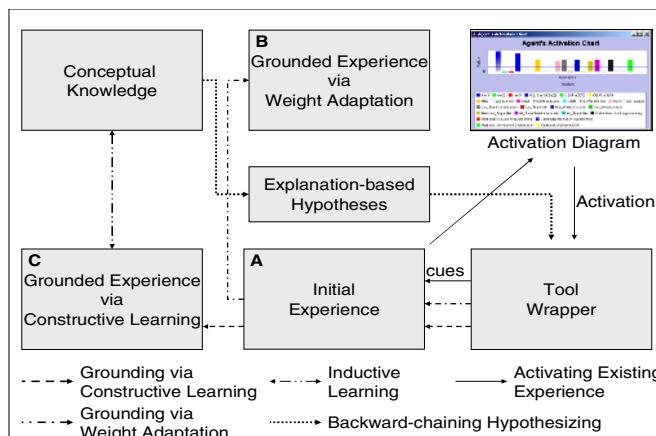
functions in the experience package. Functions related to other packages are denoted by capitals of their package names. For example, Perceptor.generateCue() depicts the method belongs to the Perceptor package.

The implemented prototype system is illustrated in Fig. 3. The tool wrapper interface allows designers to define problems. Sensors gather a user’s actions that comprise a design optimization process and activate a perceptor to create percepts. A percept cues the agent’s initial experience. Activation diagrams output the neurons winning at the equilibrium state, which represent the activated memory.

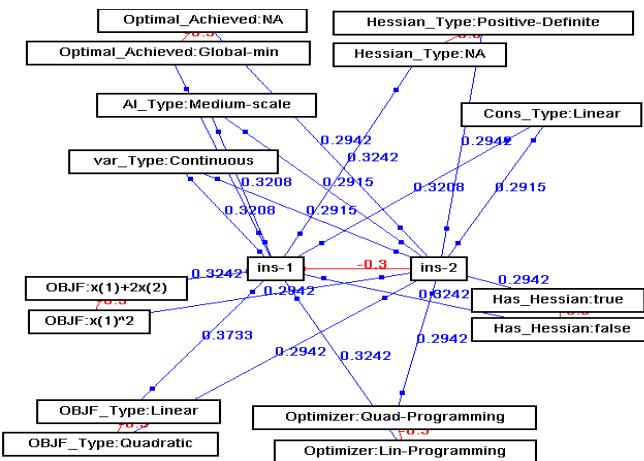
**Table 2.** Constructing a situated agent's experience with a CIAC neural net

1. Initial experience  $\leftarrow$  CIAC net build from a matrix file
2. Sensory data  $\leftarrow$  GUISensor.catcher() //captures a user's actions
3. Percepts  $\leftarrow$  Perceptor.modalitySwitch() //data transform
4. Cues  $\leftarrow$  Perceptor.generateCue() //create memory cues from percepts
5. Activation: set activation values of the cue nodes to 1.0000
6. *Cycling:* while CIAC net not isEquilibrium() //check equilibrium states
  7. for all nodes in the CIAC neural net, compute  $N_e$  for each nodes
  8. for all nodes in the CIAC neural net, update  $A_c$  for each nodes

*End Cycling* //reach equilibrium states
9. Output activated nodes //output nodes with activation value > threshold
10. Percepts  $\leftarrow$  Validator.pull() //check environment changes, transfer data
11. result  $\leftarrow$  Validator.isValid() //check the usefulness of the activated experience
12. If result = true
13. Grounding via weight adaptation() // Hebbian-like learning
- Else {
  14. Percepts  $\leftarrow$  Validator.pull() //update and pull more environment changes
  15. hypotheses  $\leftarrow$  Hypothesizer.hypothesize(Percepts, Concepts, Target)
  16. output hypotheses
  17. result  $\leftarrow$  Validator.isValid() //check hypotheses
  18. If result = true
    19. Grounding() //weight adaptation via Hebbian-like learning
  - Else { Constructive\_learning() //incorporate new experience
    20. Concepts  $\leftarrow$  Conceptor.learnInductiveConcepts() }
21. New experience

**Fig. 3.** Constructing a situated agent's experience

Based on the responses from a CIAC neural net, the agent constructs initial concepts and displays the constructed knowledge in the tool wrapper. The grounding process initiates a validation function which matches the initially constructed concepts with environmental changes. The weight adaptation function increases connection weights of the valid concept and grounds experience A to experience B. The explanation-based learner can be involved to form a new concept if no valid concept has been activated. A percept at runtime can also be developed as a new concept by a constructive learning process. Experience C is learned from constructive learning and the related self conceptual labeling process. Conceptual labels are generalised knowledge that are obtained from applying an inductive learner to the agent's experience. These conceptual knowledge serve as domain theories, from which the agent creates hypotheses. A typical grounded experience is illustrated as below.



**Fig. 4.** A typical experience grounded from an initial experience which has 1 instance node connected to several property nodes with weights “0.3000”. Property nodes are described as property and value pairs, e.g. “OBJF\_Type: Linear” represents a property node with linear objective function type.

## 4 Agent Behaviours and Experiments

This system provides a basis for us to explore the behaviours of a situated agent in various situations. We examine how the agent learns new concepts, in terms of developing knowledge structures and their intentional generalizations in its interactions with the environment. The following five internal states and their changes can be used to study how an agent constructs concepts:

1. The knowledge structure which is a Constructive Interactive Activation and Competition (CIAC) neural network composed of instance nodes connected to a number of property (or feature) nodes;
2. The expectation about environmental changes are generated by the agent experiential responses to environmental cues (shown in the activation diagram in Fig. 3);

3. The validator states show whether an agent's expectation is consistent with the environment changes;
4. Hypotheses depict the agent's reinterpretation about its failures in creating a valid expectation;
5. Concepts are the agent's high-level experiences which are domain theories an agent uses to classify and explain its observations.

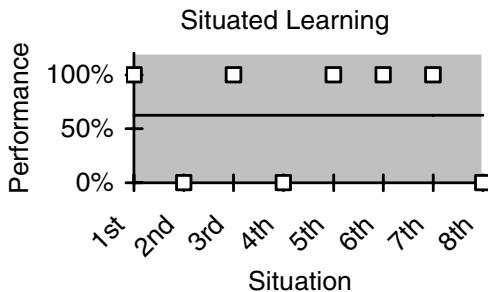
**Table 3.** Experiments with various design optimization scenarios and the agent's behaviours.  $A_c$  denotes activated experience.  $V_1$  represents the validator state for  $A_c$ .  $H_s$  are hypotheses.  $V_2$  describes the validator states for  $H_s$ .  $B_e$  is the abbreviation for the agent's behaviours.  $N_k$  means new knowledge learned. QP stands for a quadratic programming optimizer. NLP is a nonlinear programming optimizer.  $\checkmark$  shows that the agent correctly predicts the situation and  $\times$  shows the otherwise. "Ins-1" stands for design experience instance 1 and "OBJF" is the abbreviation for objective function. "Cons" represents constraints and "HF" is for Hessian function.

Design Scenarios		$A_c$	$V_1$	$H_s$	$V_2$	$B_e$	$N_k$
Identical OBJF to ins-1, Linear Cons		Ins-1	$\checkmark$	N/A	N/A	Grounds Ins-1	Grounded Ins-1
Linear OBJF, No Cons		Ins-1	$\times$	N/A	N/A	Constructs Ins-2	New Experience Ins-2
Linear OBJF, Linear Cons		Ins-1	$\checkmark$	N/A	N/A	Grounds Ins-1 Constructs Ins-3	Grounded Ins-1 and New Ins-3
Quadratic OBJF and No Cons		None	$\times$	N/A	N/A	Constructs Ins-4	New Experience Ins-4
Quadratic OBJF and No Cons		Ins-4	$\checkmark$	N/A	N/A	Grounds Ins-4 Constructs Ins-5	Grounded Ins-4 and New Ins-5
Quadratic OBJF and No Cons		Ins-4,5	$\checkmark$	N/A	N/A	Grounds Ins-4, Constructs Ins-6 Inductive Learning	Grounded Ins-4, 5 and New Ins-6; New Concepts 1,2
Quadratic OBJF, Linear Cons		Ins-4,5,6	$\times$	Is a QP	$\checkmark$	Hypothesize, Grounds Ins-4,5,6 Constructs Ins-7	Grounded Ins-4, 5, 6 and New Ins-7;
Quadratic OBJF, Linear Cons, no HF		Ins-4,5,6,7	$\times$	Is a QP	$\times$	Hypothesize, Constructs Ins-8	New Ins-8, New Concepts 3,4

- New Concept 1:  $\text{OBJF\_Type} = \text{Quadratic} \rightarrow \text{Optimizer} = \text{Quad-Programming};$
- New Concept 2:  $\text{OBJF\_Type} = \text{Linear} \rightarrow \text{Optimizer} = \text{Lin-Programming};$
- New Concept 3:  $\text{OBJF\_Type} = \text{Quadratic}$  and  $\text{Provide\_Hessian} = \text{false} \rightarrow \text{Optimizer} = \text{Nonlin-Programming};$
- New Concept 4:  $\text{OBJF\_Type} = \text{Quadratic}$  and  $\text{Provide\_Hessian} = \text{true} \rightarrow \text{Optimizer} = \text{Quad-Programming}.$

The initial experience of the agent holds one instance of a design optimization scenario using a linear programming algorithm. An experiment has been carried out to study the learning behaviours of a situated agent in heterogeneous design optimization scenarios over time. The performance is defined as the correctness of the system's response to an environment cue, which predicts hence, assists the applied design task. The “0-1” loss function is applied to measure the outcomes of the prediction. The results are illustrated in Table 3. From these results of this experiment, we can see that the agent develops its experience through reorganizing existing experience or constructing a new design experience from its interaction with the environment. It adapts to the environment with the learned knowledge, ranging from detailed design instances to generalizations of these low-level experiences, i.e., new concept 1-4 in Table 3. As shown in Fig. 5, even at the early stage of its learning, the agent achieves a performance of 62.5% in recognizing design optimization problems.

We conjecture one of the reasons for this is the content addressable ability of an CIAC neural net which can generalize across exemplars and provide plausible default values for unknown variables [3]. A situated agent that can inductively learn new concepts and subsequently deduce explanations for environment changes also adds a new level of learning to this CIAC neural net.



**Fig. 5.** Performance of the initial stage of experience learning in a situated agent. The *square dot* shows the performance of the agent in recognizing new situations in the experiment. The *dark black line* represents the mean squared estimation of the agent's performance (62.5%).

## 5 Conclusions

In this paper, we have described an approach that applies a Constructive Interactive Activation and Competition (CIAC) neural network to model a situated agent's experience. We demonstrate an implemented situated agent that uses a constructive memory model to learn new concepts from its interaction with the environment. From

the results obtained from the experiment, we can conclude that the agent develops its knowledge structures and behaviours specific to what it is confronted with – its experience. Based on the conceptual knowledge learned, the agent can further improve its learning behaviour. As a result, designers can integrate their expertise with the knowledge learned from the agent to develop design solutions. Such a system plays a potential role in enhancing the design optimization efficiency.

**Acknowledgments.** This work is supported by a Cooperative Research Centre for Construction Innovation (CRC-CI) Scholarship and a University of Sydney Sesqui R and D grant.

## References

1. Clancey, W.: *Situated Cognition*. Cambridge University Press. Cambridge (1997)
2. Clancey, W.: *Conceptual Coordination: How the Mind Orders Experience in Time*. Lawrence Erlbaum Associates, New Jersey (1999)
3. Dennis, S.: *The Interactive Activation and Competition Network: How Neural Networks Process Information*. <http://www.itee.uq.edu.au/~cogs2010/cmc/chapters/IAC/> (1998)
4. Gero, J.S.: Towards a Model of Designing which includes its Situatedness. In: Grabowski, Rude and Grein (eds.): *Universal Design Theory*. Shaker Verlag, Aachen (1998) 47-56
5. Gero, J.S.: Constructive Memory in Design Thinking. In: Goldschmidt and Porter (eds.): *Design Thinking Research Symposium: Design Representation*, MIT, Cambridge (1999) 29-35
6. Hansen, R.E.: The Role of Experience Learning: Giving Meaning and Authenticity to the Learning Process. *Journal of Technology Education*.11 (2000) 23-32
7. Jarvis, P.: Meaningful and Meaningless Experience: Towards an Analysis of Learning from Life. *Adult Education Quarterly*. 37 (1987) 164-172
8. Kolb, D.A.: *Experiential Learning: Experience as the Source of Learning and Development*. Prentice Hall, Englewood Cliffs (1984)
9. Lave, J. and Wenger, E.: *Situated Learning: Legitimate Peripheral Participation*. University of Cambridge Press, Cambridge (1991)
10. Liew, P.: *A Constructive Memory System for Situated Design Agents*. PhD Thesis, Key Centre of Design Computing and Cognition, University of Sydney, Sydney, Australia (2004)
11. McClelland, J.L.: Retrieving General and Specific Information from Stored Knowledge of Specifics. In: *Proceedings of the Third Annual Meeting of the Cognitive Science Society*, Erlbaum, Hillsdale, NJ (1981) 170-172
12. Medler, D.A.: A Brief History of Connectionism. *Neural Computing Surveys*. 1 (1998) 61-101
13. Reffat, R. and Gero, J.S.: Computational Situated Learning in Design. In: Gero, J. S. (eds.): *Artificial Intelligence in Design'00*. Kluwer Academic Publishers, Dordrecht (2000) 589-610
14. Wooldridge, M. and Jennings, N.R.: Intelligent Agents: Theory and Practice. *Knowledge Engineering Review*. 10 (1995) 115-152
15. Wordnet. <http://wordnet.princeton.edu.au/>

# Rule-Based Agents in Temporalised Defeasible Logic

Guido Governatori<sup>1,\*</sup>, Vineet Padmanabhan<sup>1,\*</sup>, and Antonino Rotolo<sup>2,\*\*</sup>

<sup>1</sup> School of ITEE, The University of Queensland, Australia

{guido, vnair}@itee.uq.edu.au

<sup>2</sup> CIRSFID, University of Bologna, Bologna, Italy

rotolo@cirsfid.unibo.it

**Abstract.** This paper provides a framework based on temporal defeasible logic to reason about deliberative rule-based cognitive agents. Compared to previous works in this area our framework has the advantage that it can reason about temporal rules. We show that for rule-based cognitive agents deliberation is more than just deriving conclusions in terms of their mental components. Our paper is an extension of [5,6] in the area of cognitive agent programming.

## 1 Introduction

There are two main trends in the agent literature for programming cognitive agents in a BDI (*belief, desire, intention*) framework. The first one is *system-based* wherein the main idea is to develop a formal specification language that provides an explicit representation of states and operations on states that underly any BDI implementation [14,13,1]. In this approach the main idea is to formalise the operational semantics of the implemented system. The second one can be termed *rule-based* where rules are used to represent or manipulate an agent's mental attitudes, i.e., an agent consists of a belief base, goal (desire) base, and intention base specified by logic formulas in the form of rules [7,3,4,8,15]. In addition to the three mental attitudes of beliefs, desires and intentions, the works above also include obligations, which are used to denote norms and commitments of social agents and social rationality. There are also works which club these two approaches like in [12]. Here we adopt the rule-based approach of [5,6] and extend it to accommodate temporal defeasible rules.

The main question we try to answer in this paper is: *What does it mean to deliberate for rule/policy-based agents?* (By policy we mean a set of rules.) Of particular concern to us is the reasoning process involved in the deliberation of a rule-based agent wherein the *agent can take a decision at t about what he/she has to do at t' based on her beliefs and policies at t*. In such a set up if no relevant event occurs then she can retain her deliberation at  $t'$ . Consider the following rule

$$p : t_p, \text{OBL } q : t_q \Rightarrow (\text{OBL } p : t_p \Rightarrow_{\text{OBL}} s : t_s) : t_r \quad (1)$$

whose reading is if  $p$  is true at time  $t_p$  and  $q$  is obligatory at time  $t_q$ , then the deontic rule  $\text{OBL } p : t_p \Rightarrow_{\text{OBL}} s : t_s$  is in force at time  $t_r$ . In this work we develop a formal machinery

\* Supported by the Australian Research Council under the Discovery Project No. DP0558854.

\*\* Supported by the European project for Standardised Transparent Representations in order to Extend Legal Accessibility (ESTRELLA, IST-4-027655).

to reason about rules like (1). In general we want to accommodate in our framework rules of the type  $(a : t \Rightarrow b : t' : t'')$  where  $t$  and  $t'$  indicate the time at which  $a$  and  $b$  hold, while  $t''$  is the time of the rule being in force. To incorporate this simple temporal reasoning we have to express whether events and states are permanent or immanent. If we use non-monotonic rules as in some of the works above then deliberation means reasoning about how to derive conclusions in terms of intentions/goals/plans, i.e., just deriving conclusions from a theory. Though [4] proposes a deliberation language, it considers only snapshots of the deliberation process in the sense that the deliberation program is dependent on the agent's current state (one cycle). Put in Bratman's terms *to reason about intentions/goals/plans at  $t$  and decide for  $t$* . A complete solution would require the addition of a temporal variable to allow reasoning about the deliberation process after each time round [9].

A formal framework like the one developed in this paper is useful in the domain of legal information systems to reason about normative conditionals [11,10]. Another domain wherein the framework could be useful is with regard to policy-based rationality as outlined by Bratman [2] in his pursuit for a *temporally extended rational agency*. The principle can be roughly stated as follows:

- At  $t_0$  agent  $A$  deliberates about what policy to adopt concerning a certain range of activities. On the basis of this deliberation agent  $A$  forms a general intention to  $\varphi$  in circumstances of type  $\psi$ .
- From  $t_0$  to  $t_1$   $A$  retains this general intention.
- At  $t_1$   $A$  notes that he/she will be (is) in circumstance  $\psi$  at  $t_2$ , where  $t_2 \geq t_1$ .
- Based on the previous steps  $A$  forms the intention at  $t_1$  to  $\varphi$  at  $t_2$ .

Notice that Bratman is concerned only with policy-based intentions<sup>1</sup> and does not provide any formal framework to show the working aspect of his historical principle. In our model we have temporal rules for beliefs, desires, intentions and obligations as in (1) and a machinery based on defeasible logic (DL) to reason about such temporal rules.

Given the temporal nature of Bratman's historical principle, and the idea that some intentions can be retained from one moment to another, we must then account for two types of temporal deliberations: transient deliberations, which hold only for an instant of time, and persistent deliberations, in which an agent is going to retain them unless some intervening event that forces the agent to reconsider her deliberation occurs. This event can be just a brute fact or it can be a modification of the policy of the agent. Thus an agent must be able to cope with changes of the environment but also of her policies.

Let us consider the following scenario. Our agent (Guido) has sent a paper to the PRICAI-06 conference, and he has the intention to attend the conference to present the paper if accepted. Guido's school policy for funding is that if somebody intends to travel then she has to submit the request for travel funds two weeks (15 days) before the actual travel. This scenario can be represented as follows:

$$r_1 (\text{PRICAI} \text{paperAccepted} : t_1 \Rightarrow_{\text{INT}} \text{Travel} : t_2) : t_0 \quad (2)$$

$$r_2 (\text{INT Travel} : t_X \Rightarrow_{\text{OBL}} \text{Request} : t_{X-15}) : t_0 \quad (3)$$

---

<sup>1</sup> In [2] historical principles for deliberative as well as non-deliberative intentions is outlined. Here we are concerned only with the policy-based aspect.

Rule  $r_1$  states that Guido will form the intention to travel to PRICAI at a certain time in case the paper is accepted and that the rule is in force from  $t_0$  (let us say that  $t_0$  is the time when the paper is submitted to the conference), and  $r_2$  encodes Guido's school travel policy (the policy is in force at time  $t_0$ ). Suppose that Guido, at time  $t_1$ , receives the notification that the paper has been accepted, then at that time he forms the intention to travel to PRICAI at time  $t_2$ . This triggers his obligation to have the travel request submitted two weeks before the date of the conference. Accordingly, he plans to prepare the required paperwork in due time.

Time passes and two important events happen: the School updates the travel policy and Guido is appointed to a *research-only* position. The changes to the travel policy concerns research-only staff and the actual change is that, due to the new accounting software travel funds could be made available in less than one week. Thus the rule encoding the update to the policy is

$$r_3 (Research : t_Y \Rightarrow (\text{INT Travel} : t_X \Rightarrow_{\text{OBL}} \text{Request} : t_{X-7}) : t_4) : t_3$$

Here  $t_3$  is when the new policy has been issued and  $t_4$  is the time the new policy will be effective. Based on the updated policy and the new event, Guido complies with the new obligation if he submit the application for funds one week before travelling. Accordingly, he can change his plans and can postpone to fill all forms to a later time.

## 2 Temporalised DL for Cognitive Agents

We focus on how mental attitudes and obligations jointly interplay in modelling agent's deliberation and behaviour. Such an interplay is modelled within a temporal setting. The logical framework is based on DL, which is a simple and flexible sceptical non-monotonic formalism that has proven able to represent various aspects of non-monotonic reasoning. We extend here the machinery developed in [11,6,5] to represent temporalised motivational attitudes of agents.

The basic language is based on a (numerable) set of atomic propositions  $Prop = \{p, q, \dots\}$ , a set of rule labels  $\{r_1, r_2, \dots\}$ , a discrete totally ordered set of instants of time  $\mathcal{T} = \{t_1, t_2, \dots\}$ , a set of modal operators  $M = \{\text{BEL}, \text{DES}, \text{INT}, \text{OBL}\}$  (belief, desire, intention, and obligation, respectively), and the negation sign  $\neg$ . A plain literal is either an atomic proposition or the negation of it. If  $l$  is a plain literal then, for any  $X \in M$ ,  $Xl$  and  $\neg Xl$  are modal literals. A literal is either a plain literal or a modal literal. Given a literal  $l$ ,  $\sim l$  denotes the complement of  $l$ , that is, if  $l$  is a positive literal  $p$  then  $\sim l = \neg p$ , and if  $l = \neg p$  then  $\sim l = p$ . A temporal literal is a pair  $l : t$  where  $l$  is a literal and  $t \in \mathcal{T}$ . Intuitively, a temporal literal  $l : t$  means that  $l$  holds at time  $t$ .

Knowledge in DL can be represented in two ways: facts and rules. *Facts* are indisputable statements, represented in the form of literal and modal literals. For example, "John is a minor". In the logic, this might be expressed as  $\text{Minor}(John)$ . A *rule* is a relation (represented by an arrow) between a set of premises (conditions of applicability of the rule) and a conclusion. In this paper, conclusions usually correspond to literals, but for a special class of rules they can also be rules themselves; in addition all the conclusions and the premises will be qualified with the time when they hold. We consider four classes of rules: rules for belief, desire, intention and obligation. Each class

of rules is qualified by labelling the arrow with any  $X \in M$  (for belief, desire, intention, and obligation). If  $X \in \{\text{DES}, \text{INT}, \text{OBL}\}$ , applicability of the corresponding rules permits to derive only literals: more precisely, if the consequent of such rules is a literal  $l : t$ , then their applicability leads to obtain the modal literal  $Xl : t$ . For any consequent  $l : t$  obtained through rules for  $X \in \{\text{DES}, \text{INT}, \text{OBL}\}$ ,  $l : t$  is called a temporal goal. Rules for belief play a special role. They constitute the basic inference mechanism of an agent, as they concern the knowledge an agent has about the world. For this reason, their conclusions, if obtained, are not modalised; on the other hand this is the only class of rules for which conclusions can be also rules (rules for  $X \in M$ ).

Besides the above classification, rules can be partitioned according to their strength into *strict rules* (denoted by  $\rightarrow$ ), *defeasible rules* (denoted by  $\Rightarrow$ ) and defeaters (denoted by  $\rightsquigarrow$ ). Strict rules are rules in the classical sense: they are monotonic and whenever the premises are indisputable so is the conclusion. Defeasible rules, on the other hand, are non-monotonic: they can be defeated by contrary evidence. Defeaters are the weakest rules: they do not support directly conclusions, but can be used to block the derivation of opposite conclusions. Henceforth we use  $\hookleftarrow$  as a metavariable for either  $\rightarrow$  when the rule is a strict rule,  $\Rightarrow$  when the rule is a defeasible rule, and  $\rightsquigarrow$  when the rule is a defeater. Thus we define the set of rule, Rules, using the following recursive definition:

- a rule is either a rule for  $X$ ,  $X \in M$  or the empty rule  $\perp$
- If  $r$  is a rule and  $t \in \mathcal{T}$ , then  $r : t$  is a temporalised rule. (The meaning of a temporalised rule is that the rule is valid at time  $t$ .)
- Let  $A$  be a finite set of temporal literals,  $C$  be a temporal literal and  $r$  a temporalised rule, then  $A \hookleftarrow_X C$ ,  $A \hookleftarrow_X r$  are rules for  $X = \text{BEL}$ .
- Let  $A$  be a finite set of temporal literals and  $C$  be a temporal plain literal. Then  $A \hookleftarrow_X C$  is a rule for  $X \in \{\text{DES}, \text{INT}, \text{OBL}\}$ .

For a rule  $r$  labelled with any  $X \in M$  we will use  $A(r)$  to indicate the body or antecedent of the rule and  $C(r)$  for the head or consequent of the rule. It is also possible to have nested rules i.e., rules occurring inside rules for beliefs. However, it is not possible for a rule to occur inside itself. Thus for example, the following is a rule

$$p : t_p, \text{OBL}q : t_q \Rightarrow_{\text{BEL}} (\text{OBL}p : t_p \Rightarrow_{\text{INT}} s : t_s) : t_r \quad (4)$$

(4) means that if  $p$  is true at time  $t_p$  and  $q$  is obligatory at time  $t_q$ , then the intention rule  $\text{OBL}p : t_p \Rightarrow_{\text{INT}} s : t_s$  is valid at time  $t_r$ . Every temporalised rule is identified by its rule label and its time. Formally we can express this relationship by establishing that every rule label  $r$  is a function  $r : \mathcal{T} \mapsto \text{Rules}$ . Thus a temporalised rule  $r : t$  returns the value/content of the rule ‘ $r$ ’ at time  $t$ . This construction allows us to uniquely identify rules by their labels, and to replace rules by their labels when rules occur inside other rules. In addition there is no risk that a rule includes its label in itself. For example if we associate the temporal rule  $(\text{OBL}p : t_p \Rightarrow_{\text{INT}} s : t_s) : t_r$  to the pair  $r_1 : t_r$ , we can concisely rewrite (4) as

$$p : t_p, \text{OBL}q : t_q \Rightarrow_{\text{BEL}} r_1 : t_r \quad (5)$$

It should be noted that we have to consider two temporal dimensions for rules. The first regards the efficacy (effectiveness) of a rule i.e., the capacity of a rule to produce a

desired effect at a certain time point, and the second shows when the rule is valid/comes into force. Consider the following two rules about a hypothetical tax regulation:

$$r_1 : (\text{Income} > 90K : 1\text{Mar} \Rightarrow_{\text{OBL}} \text{Tax10} : 1\text{Jan}) : 1\text{Jan} : 15\text{Jan} \quad (6)$$

$$r_2 : (\text{Income} > 100K : 1\text{Mar} \Rightarrow_{\text{OBL}} \text{Tax40} : 1\text{Jan}) : 1\text{Apr} : 1\text{Feb} \quad (7)$$

Rule  $r_1$  states that if the income of a person is in excess of ninety thousand as of 1st March ( $\text{Income} > 90K : 1\text{Mar}$ ) then he/she is obliged to pay the top marginal tax rate of 10 percent from 1st January ( $\text{Tax10} : 1\text{Jan}$ ) with the policy being in force from 15 January, and effective from 1st January. This means that the norm becomes part of the tax regulation from 15 January, but it is effective from 1st January. Accordingly, the policy covers tax returns lodged after 15 January as well as all tax returns lodged before the validity of the policy itself. The second rule, valid (i.e., part of the tax regulation) from 1st February, establishes a top marginal tax rate of 40% for tax returns lodged after the effectiveness date of 1st April.

The above two rules illustrate the difference between the *effectiveness* and *validity* of a rule. In order to differentiate between the effectiveness and validity of a rule we introduce the notion of *temporalised rule with viewpoint* and 15 Jan in  $r_1$  denotes exactly this. A conclusion or a temporalised rule with viewpoint is an expression  $s@t$ , where  $t \in \mathcal{T}$ , meaning that  $s$  “holds” when the agent reasons using the information available to her at  $t$ . Thus the expression  $r_1 : t_1 @ t_2$  represents a rule  $r_1$  valid at time  $t_2$  and effective at time  $t_1$ . In the case of (6) this could be given as  $s@16\text{Jan}$  where  $s$  is  $(\text{Income} > 90K : 1\text{Mar} \Rightarrow_{\text{OBL}} \text{Tax10} : 1\text{Jan}) : 1\text{Jan}$  and  $t = 16\text{Jan}$ . Thus for an agent intending to lodge a tax return on 16 Jan, there are no alternatives. She has to pay her taxes at the top marginal rate of 10%. However, should she postpone the decision after 1 February, then she has the option to evaluate when and how much tax she has to pay (10% if the tax return is lodged before 1 April and 40% if lodged afterward). Therefore she can plan her actions in order to achieve the most suitable result according to her goals. Hence, an agent equipped with such temporal rules should be able to figure out plans that are applicable at a particular time point. Temporal rules like (7) are more interesting, as they allow the agent to plan using rules having reference to past as well as future time points. We discuss more about *temporalised rule with viewpoint* in section 4. In addition the example shows that in general, unlike other approaches, there is no need to impose constraints on the time instants involved in a rule.

Another issue we need to consider here is that we have two different types of conditionals to derive beliefs and goals (i.e., rules labelled with  $X \in M$ ): conditionals that initiate a state of affairs which persists until an interrupting event occurs, and conditionals where the conclusion is co-occurrent with the premises. To represent this distinction we introduce a further distinction of rules, orthogonal to the previous one, where rules are partitioned in persistent and transient rules. A persistent rule is a rule whose conclusion holds at all instants of time after the conclusion has been derived, unless interrupting events occur; transient rules, on the other hand, establish the conclusion only for a specific instant of time. We use the following notation to differentiate the various types of rules: with  $\hookrightarrow_X^t$  we represent a transient rule for  $X$ , and with  $\hookrightarrow_X^p$  a persistent rule.

Given a set  $R$  of rules, we denote the set of strict rules in  $R$  by  $R_s$ , the set of strict and defeasible rules in  $R$  by  $R_{sd}$ , the set of defeasible rules in  $R$  by  $R_d$ , and the set of

defeaters in  $R$  by  $R_{\text{dfi}}$ .  $R[q : t]$  denotes the set of rules in  $R$  with consequent  $q : t$ . We use  $R^X$  for the set of rules for  $X \in M$ . The set of transient rules is denoted by  $R^{tr}$  and the set of persistent rules by  $R^{per}$ . Finally we assume a set of rule modifiers. A rule modifier is a function  $m : \text{Rules} \times \mathcal{T} \mapsto \text{Rules} \times \mathcal{T}$ .

The above constructions allow us to use rule modifiers on rule labels. Thus  $m(r_1 : t_1) : t_2$  returns the rule obtained from  $r_1$  as such at time  $t_1$  after the application of the modification corresponding to the function  $m$  and the result refers to the content of the rule at time  $t_2$ . Given this basic notion of rule modifier, we can define some functional predicates, i.e. specific rule-modifications. For the sake of brevity, we omit the technical details on how to adapt the basic definition of rule modifier to cover these specific rule modifications: Delete, Update and Add. As we shall see, these functional predicates can only occur in the head of belief rules. For the moment let us see their intuitive reading. The functional predicate  $\text{Delete}(r) : t'$  says that a given rule  $r$  is deleted at  $t'$ . More precisely,  $\text{Delete}(r) : t'$  assigns the empty rule  $r : (\perp) : t'$  to  $r$  as holding at  $t$ . The rule  $r$  is thus dropped at  $t'$  from the system and so, at  $t'$ ,  $r$  is no longer valid. If  $r$  is a rule for  $X \in \{\text{DES}, \text{INT}, \text{OBL}\}$ , let  $\mathcal{A}$  and  $\mathcal{C}$  be a set of temporal literals and a temporal plain literal respectively; if  $r$  is a rule for belief, let  $\mathcal{A}$  be defined as before, while  $\mathcal{C}$  is a temporal plain literal or a temporalised rule. Then

$$\text{Update}(r, \mathcal{A}) : t' \quad \text{Update}(r, \mathcal{C}) : t'$$

say that we operate, at  $t'$  an update of  $r$  which replaces a subset or all components in the antecedent of  $r$  with other appropriate components and the consequent with a new appropriate element of the language. The new version of  $r$  will hold at  $t'$ . Similarly

$$\text{Add}(r', A(r'), C(r')) : t'$$

indicates that a new rule  $r'$  is added at  $t'$  to the system, and that  $r'$  has the antecedent and consequent specified by  $A(r')$  and  $C(r')$ .

### 3 Conflicts Between Rule Modifications

Table 1 summarises the basic conflicts between rule modifications. Notice that conflicts obtain only if the conflicting modifications apply to the same time instant. Deleting a rule  $r$  is incompatible with any update of  $r$  (first and second rows from the top). This is the only case of real conflict. In fact, the third row from the top considers a “residual” but in theory possible conflict between modifications, namely, between those of deleting and adding at the same time a rule  $r$ . This case is marginal essentially because adding a rule  $r$  usually means that  $r$  is not valid in the theory. However, nothing prevents to add a rule  $r$  which is already valid in the system. In this case, the operation is redundant, but, if performed together with deleting  $r$ , we have indeed a conflict between modifications.

**Table 1.** Conflicts

Modifications	Conditions
$\text{Delete}(r) : t'$	$\text{Update}(r, \mathcal{A}) : t''$
$\text{Delete}(r) : t'$	$\text{Update}(r, \mathcal{C}) : t''$
$\text{Delete}(r) : t'$	$\text{Add}(r, A(r), C(r)) : t''$

## 4 Temporalised Rule with View Point

In [11] we showed how to derive temporal literals in a DL framework. But this is of limited use and what we need is a way to derive temporal rules. In this section we extend the framework developed in [11,6,5] with *temporal rules with a view point*. What this means is that we can reason about temporal rules that are valid at a particular instant of time. Suppose that we have a defeasible theory  $D = (\mathcal{T}, F, R, \prec)$  where  $\mathcal{T}$  is discrete totally ordered set of instants of time,  $F$  is a finite set of temporalised literals,  $R$  a finite set of rules (comprising strict, defeasible and defeater rules) and  $\prec$  a ternary relation (superiority relation) over  $R \times R \times \mathcal{T}$ , meaning that one rule is stronger than another rule at a particualr time; for example  $r_1 \prec_t r_2$  means that rule  $r_2$  is stronger than rule  $r_1$  at time  $t$ . Conclusions in DL can have one of the following four forms (where  $X$  ranges over  $M$ ):

- $+ \Delta_X @ t q : t'$  meaning that  $q$  is definitely provable with mode  $X$ , at time  $t'$  with viewpoint  $t$ , in  $D$  (i.e., using only facts and strict rules).
- $- \Delta_X @ t q : t'$  meaning that we have proved that  $q$  is not definitely provable with mode  $X$ , at time  $t'$  with viewpoint  $t$ , in  $D$ .
- $+ \partial_X @ t q : t'$  meaning that  $q$  is defeasibly provable with mode  $X$ , at time  $t'$  with viewpoint  $t$ , in  $D$
- $- \partial_X @ t q : t'$  meaning that we have proved that  $q$  is not defeasibly provable with mode  $X$ , at time  $t'$  with viewpoint  $t$ , in  $D$ .

For example,  $+ \text{OBL} @ t_1 q : t_0$  means that we have a defeasible proof for  $\text{OBL}q$  at  $t_0$ , or, in other words, that  $\text{OBL}q$  holds at time  $t_0$  when we use the rules in force in the system at time  $t_1$ . However, these tags do not take care whether a conclusion  $q : t$  is obtained via transient rules (that is,  $q$  holds only at time  $t_0$ ) or via persistent rules, in such a case for every  $t'$  such that  $t_0 < t'$ , the property  $q$  persists at time  $t'$ , unless we have other evidence on the contrary, i.e., a piece of evidence that terminates the property  $q$ . To reflect these issues, we introduce auxiliary proof tags for persistent and transient conclusions. Formally,  $+ \Delta_X @ t p$  means that either  $+ \Delta_X^{tr} @ t p$  or  $+ \Delta_X^{pr} @ t p$ , i.e., either  $p$  is transient at  $t$  or it is persistent at  $t$ ;  $- \Delta_X @ t p$  means both  $- \Delta_X^{tr} @ t p$  or  $- \Delta_X^{pr} @ t p$ , i.e., it is not true that  $p$  is transient at  $t$  and that  $p$  is not persistent at  $t$ .

The proof tags are labelled with the mode used to derive the rule, according to their appropriate proof conditions. It is not possible to give the complete set of proof conditions in this paper. Here we concentrate only on the proof conditions to derive defeasible persistence of both rules with belief mode, and literals. The proof conditions given here are extensions of those given in [11] for the temporal aspects and can be used for goals and planning as in [6,5]. The proof conditions missing in this paper can be obtained from the corresponding conditions of [11,6,5] using the same intuition on which the proof conditions we are going to show illustrate.

Provability is based on the concept of a *derivation* (or proof) in  $D$ . A derivation is a finite sequence  $P = (P(1), \dots, P(n))$  of tagged literals satisfying the proof conditions (which correspond to inference rules for each of the kinds of conclusion).  $P(1..n)$  denotes the initial part of the sequence  $P$  of length  $n$ . A strict derivation (i.e., a conclusion tagged with  $\Delta$ ) is a monotonic derivation using forward chaining of rules, i.e., modus ponens. In DL a defeasible derivation, on the other hand, has three phases. In the first

phase we propose an argument in favour of the conclusion we want to prove. In the simplest case this consists of an applicable rule for the conclusion (a rule is applicable if the antecedent of it has already been proved). Then in the second phase we examine all possible counter-arguments (rules for the opposite conclusion). Finally we have to rebut the counter-arguments. Thus we have to provide evidence against the counter-argument. Accordingly, we can demonstrate that the argument is not as such (i.e., some of its premises are not provable), or we can show that the counter-argument is weaker than an argument for the conclusion.

For persistent conclusions we have another method. We can use a derivation of the conclusion at a previous time provided that no terminating event occurred in between. In [11] the rules are given, but here rules are can also be derived. Thus in the proof conditions we have to cater for this option. Accordingly, we have to give conditions that allows us to derive rules instead of literals. For the sake of simplicity we will assume that all rules in  $R$  can be overruled/modified. Then we have to extend the notation  $R[x : t]$  to the case where  $x$  is a rule label (and rule-modifiers). Given a set of belief rules  $R$  and a set of rule modifiers  $M = \{m_1, \dots, m_n\}$ , then

$$R[r : t_r] = \{s \in R : A(s) = m_i(v : t_v) \text{ and } m_i(v : t_v) = r : t_r\}$$

$R[r : t_r]$  gives the set of nested rules whose head results in the rule  $r : t_r$  after the application of the rule modifier; and

$$R[\sim r : t_r] = \{s \in R : A(s) = m_i(r : t_r) \text{ and } m_i(r : t_r) \text{ is in conflict with } r : t_r\}$$

The set  $R[\sim r : t_r]$  gives the set of rules that modify  $r : t_r$  and the modification is in conflict with the  $r : t_r$ , see Table 1 for such conflicts.

We can now give the proof conditions for  $+ \partial^{pr}$  to derive a rule.

If  $P(n+1) = + \partial_{\text{BEL}}^{pr} @ t r : t_r$  then

- 1a)  $r : t_r @ t \in R^{\text{BEL}}$  or
- 1b)  $\exists s : t_s \in R^{\text{BEL}}[r : t_r]$  such that  $+ \partial_{\text{BEL}} @ t s : t_s \in P(1..n)$  and  
 $\forall Y_a a : t' \in A(s), + \partial_{Y_a} @ t a : t' \in P(1..n);$  and
- 2)  $\forall v : t_v \in R^{\text{BEL}}[\sim r : t_r]$  if  $+ \partial_{\text{BEL}} @ t v : t_v \in P(1..n)$ , then either
  - 2.1)  $\exists Y_b b : t'' \in A(v)$  such that  $- \partial_{Y_b} @ t b : t'' \in P(1..n)$  or
  - 2.2 a)  $v : t_v \prec_t r : t_r$  if 1a obtain or b)  $v : t_v \prec_t s : t_s$  if 1b obtain; or
- 3)  $+ \partial_{\text{BEL}}^{pr} @ t' r : t_r \in P(1..n), t' < t$  and
  - 3.1)  $\forall t'', t' \leq t'' < t, \forall s : t_s \in R[\sim r : t_r]$  if  $+ \partial_{\text{BEL}} @ t'' s : t_s \in P(1..n)$ , then
    - 3.1.1)  $\exists Y_a a : t_a \in A(s), - \partial_{Y_a} @ t'' a : t_a \in P(1..n)$  or  $t_s < t_r$ ; and
- 4)  $+ \partial_{\text{BEL}}^{pr} @ t r : t'_r \in P(1..n), t'_r < t$  and
  - 4.1)  $\forall t', t'_r \leq t'' < t_r, \forall s : t_s \in R[\sim r : t_r]$  if  $+ \partial_{\text{BEL}} @ t' s : t_s \in P(1..n)$ , then
    - 4.1.1)  $\exists Y_a a : t_a \in A(s), - \partial_{Y_a} @ t' a : t_a \in P(1..n)$  or  $t_s < t'_r$ .

Let us briefly examine the above proof conditions. To prove a rule at time  $t$ , the rule must be in force at time  $t$ , i.e., the rule must be one of the given rules (condition 1a). There is a second possibility that the rule is derived from another rule. The second rule must be provable and applicable at  $t$  (condition 1b). However, this is not enough since there could have been modifications to the rule effective at  $t$ . Thus we have to show that either all eventual modifications were not applicable (2.1) or the modifications

were not successful since they were defeated (2.2a and 2.2b). Finally the rule could be provable because it was persistent, i.e., it was persistently in force before (3), and no modification occurred in between. The possible modifications in force after the rule was in force were not applicable to the rule. Or (4) the rule was persistently effective before, and its effectiveness was not revoked.

The conditions for positive persistent defeasible proofs are as follows:

If  $P(n+1) = +\partial_X^{pr} @ t q : t'$  then

1)  $+ \Delta_X^{pr} @ t q : t' \in P(1..n)$ , or

2)  $- \Delta_X @ t \sim q : t' \in P(1..n)$ , and

2.1)  $\exists r : tr \in R_{sd}^{X,pr}[q : t'] : +\partial_{BEL} @ t r : tr \in P(1..n)$ , and

$\forall Y_a a : ta \in A(r : tr), +\partial_{Y_a} @ t a : ta \in P(1..n)$  and

2.2)  $\forall s : ts \in R^X[\sim q : t]$ : if  $+ \partial_{BEL} @ t s : ts$ , then either

2.2.1)  $\exists Y_a a : ta \in A(s : ts), -\partial_{Y_a} @ t a : ta \in P(1..n)$ ; or

2.2.2)  $\exists w : tw \in R^X[q : t] : +\partial_{BEL} @ t w : tw \in P(1..n)$  and

$\forall Y_a a : ta \in A(w : tw), +\partial_{Y_a} @ t a : tw \in P(1..n)$  and  $w \succ s$ ; or

3)  $\exists t'' \in \mathcal{T} : t'' < t$  and  $+ \partial_X^{pr} @ t'' q : t' \in P(1..m)$  and

3.1)  $\forall t''' t'' < t''' \leq t \forall s : ts \in R^X[\sim q : t']$ : if  $+ \partial_{BEL} @ t''' s : ts \in P(1..n)$ , then

3.1.1)  $\exists Y_a a : ta \in A(s : ts), -\partial_{Y_a} @ t''' a : ta \in P(1..n)$  or

3.1.2)  $\exists v : tv \in R^X[q : t'] : +\partial_{BEL} @ t''' v : tv \in P(1..n)$  and

$\forall Y_b b : tb \in A(v : tv) + \partial_{Y_b} @ t''' b : tb \in P(1..n)$  and  $s : ts \prec_{t'''} v : tv$ ; or

4)  $\exists t'' \in \mathcal{T} : t'' < t'$  and  $+ \partial_X^{pr} @ t q : t'' \in P(1..m)$  and

4.1)  $\forall t''' t'' < t''' \leq t' \forall s : ts \in R^X[\sim q : t''']$ : if  $+ \partial_{BEL} @ t s : ts \in P(1..n)$ , then

4.1.1)  $\exists Y_a a : ta \in A(s : ts), -\partial_{Y_a} @ t a : ta \in P(1..n)$  or

4.1.2)  $\exists v : tv \in R^X[q : t'''] : +\partial_{BEL} @ t v : tv \in P(1..n)$  and

$\forall Y_b b : tb \in A(v : tv) + \partial_{Y_b} @ t b : tb \in P(1..n)$  and  $s : ts \prec_{t'''} v : tv$ .

Clause 1 of the above proof condition allows us to infer a defeasible persistent conclusion from a strict persistent conclusion with the same mode. Clause 2 requires that the complement of the literal we want to prove is not definitely provable (or definitely provable for  $-\partial$ ), but it does not specify whether it is persistent or transient: remember that what we want to achieve is to see whether the literal or its complement are provable at  $t$  but not both; in the same way, and for the same reason,  $q$  can be attacked by any rule for the complement of  $q$  (clauses 2.2.1). An important issue in all clauses of this proof condition is that each time we have to use a rule (either to support the conclusion (2.1), to attack it (2.2) or to rebut the attack (2.2.2)) we must have that the rule is provable at time  $t$  of the derivation (@ $t$ ). Clauses 3 and 4 are the clauses implementing persistence (i.e., the conclusion has been derived at a previous time and carries over to the current time). Essentially clause 3 ensures that the conclusion has been derived at a previous time  $t''$  and no interrupting event occurred between  $t''$  and  $t$ ; while clause 4 takes care of the case where  $q$  is derived persistently for a time before  $t'$ , and that no interrupting event will occur between the effectiveness of  $q$  and the time  $q$  is expected to hold according to the current derivation.

## 5 Summary

In this paper we combined and extended the approaches presented in [11] and [6,5]. In particular we have extended the programming cognitive agents approach with

temporalised literals. This makes the resulting logic more expressive and more suitable for the task at hand. In addition we have introduced the notion of view-point. The deliberation of an agent based on a policy depends not only on the environment but also on the rules in force in the policy at the time of deliberation and at the time when the plan resulting from the deliberation will be executed. These two aspects are neglected in the literature on agent planning. In addition the framework we propose can handle revision of theories in the same way the framework is inspired to handle complex modification of normative codes [10]. An aspect we did not consider here is how to extend the temporal framework to reason with actions and their duration. This matter is left for future work.

## References

1. R.H. Bordini, A.L. C. Bazzan, R. de O. Jannone, D.M. Basso, R.M. Vicari, and V.R. Lesser. Agentspeak(xl): efficient intention selection in BDI agents via decision-theoretic task scheduling. In *AAMAS'02*, pp. 1294–1302, 2002.
2. M.E. Bratman. *Intentions, Plans and Practical Reason*. Harvard University Press, 1987.
3. J. Broersen, M. Dastani, J. Hulstijn, and L. van der Torre. Goal generation in the BOID architecture. *Cog. Sc. Quart.*, 2:428–447, 2002.
4. M. Dastani, F. de Boer, F. Dignum, and J.-J. Meyer. Programming agent deliberation. In *AAMAS'03*, pp. 91–104, ACM Press, 2003.
5. M. Dastani, G. Governatori, A. Rotolo, and L. van der Torre. Preferences of agents in defeasible logic. In *Australian AI05*, pp. 695–704. Springer, 2005.
6. M. Dastani, G. Governatori, A. Rotolo, and L. van der Torre. Programming cognitive agents in defeasible logic. In *LPAR 2005*, pp. 621–636. Springer, 2005.
7. M. Dastani and L.W.N. van der Torre. Programming BOID-plan agents: Deliberating about conflicts among defeasible mental attitudes and plans. In *AAMAS 2004*, pp. 706–713, 2004.
8. M. Dastani, B. van Riemsdijk, F. Dignum, and J.-J. Meyer. A programming language for cognitive agents: Goal directed 3APL. In *ProMAS'03*, pp. 111–130. Springer, 2003.
9. A.S. d'Avila Garcez and L.C. Lamb. Reasoning about time and knowledge in neural symbolic learning systems. In Sebastian Thrun, Lawrence Saul, and Bernhard Schölkopf, editors, *Advances in Neural Information Processing Systems 16*. MIT Press, 2004.
10. G. Governatori, M. Palmirani, R. Riveret, A. Rotolo, and G. Sartor. Norm modifications in defeasible logic. In *JURIX05*, pp. 13–22. IOS Press, 2005.
11. G. Governatori, A. Rotolo, and G. Sartor. Temporalised normative positions in defeasible logic. In *ICAIL*, pp. 25–34. ACM Press 2005.
12. K.V. Hindriks, F.S. De Boer, Hoek W. van der, and J.-J. Meyer. Agent programming in 3(apl). *Autonomous Agents and Multi-Agent Systems*, 2(4):357–401, 1999.
13. M.D. Inverno and M. Luck. Engineering agentspeak(l): A formal computational model. *Journal of Logic and Computation*, 8:1–27, 1998.
14. A.S. Rao. Agentspeak(L): BDI agents speak out in a logical computable language. Technical report, Australian Artificial Intelligence Institute, 1996.
15. M. Birna van Riemsdijk, M. Dastani, and J.-J. Meyer. Semantics of declarative goals in agent programming. In *AAMAS 2005*, pp. 133–140. ACM Press, 2005.

# Compact Preference Representation for Boolean Games

Elise Bonzon, Marie-Christine Lagasquie-Schiex, and Jérôme Lang

IRIT, UPS, F-31062 Toulouse Cedex 9, France

{bonzon, lagasq, lang}@irit.fr

**Abstract.** Boolean games, introduced by [15,14], allow for expressing compactly two-players zero-sum static games with binary preferences: an agent’s strategy consists of a truth assignment of the propositional variables she controls, and a player’s preferences is expressed by a plain propositional formula. These restrictions (two-players, zero-sum, binary preferences) strongly limit the expressivity of the framework. While the first two can be easily encompassed by defining the agents’ preferences as an arbitrary  $n$ -uple of propositional formulas, relaxing the last one needs Boolean games to be coupled with a propositional language for compact preference representation. In this paper, we consider generalized Boolean games where players’ preferences are expressed within two of these languages: prioritized goals and propositionalized CP-nets.

## 1 Introduction

The framework of Boolean games, introduced by [15,14], allows for expressing compactly two-players zero-sum static games with binary preferences: an agent’s strategy consists of a truth assignment of the propositional variables she controls, and a player’s preferences is expressed by a plain propositional formula. Arguably, these three restrictions (two-players, zero-sum, binary preferences) strongly limit the expressivity of the framework. The first two can be easily encompassed by defining the agents’ preferences as an arbitrary  $n$ -uple of propositional formulas (see [3], who addresses complexity issues for these binary  $n$ -players Boolean games). In this paper we focus on the third one, which needs considerably more work to be dealt with. The starting point of our paper is that whereas a single propositional formula (goal)  $\varphi$  cannot express more than a binary preference relation on interpretations (models of  $\varphi$  are strictly better than models of  $\neg\varphi$ ), expressing arbitrary (non-binary) preferences within a propositional framework is possible, making use of a *propositional language for compact preference representation*. The study of such languages has been a very active issue for a few years in the AI community. Several classes of languages based on propositional logic have been proposed and studied (see for instance [16,8] for an overview of these languages).

A first question has to be addressed before going further: should agents’ preferences be expressed in a numerical way or in an ordinal way? This depends a lot on the notions we want to deal with. While some notions (such as pure Nash equilibria and dominated strategies) can be defined in a purely ordinal setting, other ones (such as mixed strategy Nash equilibria) need quantitative (real-valued) preferences. Here we choose to stick to ordinal settings (we leave numerical preferences in Boolean games for further

work – see Section 5), and we successively integrate Boolean games with two of these languages: first, *prioritized goals*, and then (*propositionalized*) *CP-nets*.

In Section 2, some background is given and we define  $n$ -players, non zero-sum Boolean games with binary preferences. Boolean games are then enriched with prioritized goals in Section 3, and with propositionalized CP-nets in Section 4. Section 5 addresses related work and further issues.

## 2 $n$ -Players Boolean Games

Let  $V = \{a, b, \dots\}$  be a finite set of propositional variables and  $L_V$  be the propositional language built from  $V$  and the usual connectives as well as the Boolean constants  $\top$  (*true*) and  $\perp$  (*false*). Formulas of  $L_V$  are denoted by  $\varphi, \psi$ , etc. A *literal* is a formula of the form  $x$  or of the form  $\neg x$ , where  $x \in V$ . A *term* is a consistent conjunction of literals.

$2^V$  is the set of the interpretations for  $V$ , with the usual meaning that an interpretation  $M$  gives the value *true* to a variable  $x$  if  $x \in M$ , and the value *false* otherwise.  $\models$  denotes classical logical consequence. Let  $X \subseteq V$ .  $2^X$  is the set of  $X$ -interpretations. A *partial interpretation* (for  $V$ ) is an  $X$ -interpretation for some  $X \subseteq V$ . Partial interpretations are denoted by listing all variables of  $X$ , with a  $\bar{\phantom{x}}$  symbol when the variable is set to false: for instance, let  $X = \{a, b, d\}$ , then the  $X$ -interpretation  $M = \{a, d\}$  is denoted  $a\bar{b}d$ . If  $\{V_1, \dots, V_p\}$  is a partition of  $V$  and  $\{M_1, \dots, M_p\}$  are partial interpretations, where  $M_i \in 2^{V_i}$ ,  $(M_1, \dots, M_p)$  denotes the interpretation  $M_1 \cup \dots \cup M_p$ .

Given a set of propositional variables  $V$ , a Boolean game on  $V$  [15,14] is a zero-sum game with *two players* (1 and 2), where the actions available to each player consist in assigning a truth value to each variable in a given subset of  $V$ . The utility functions of the two players are represented by a propositional formula  $\varphi$  formed upon the variables in  $V$  and called *Boolean form* of the game<sup>1</sup>.  $\varphi$  represents the goal of Player 1: her payoff is 1 when  $\varphi$  is satisfied, and 0 otherwise. Since the game is zero-sum<sup>2</sup>, the goal of Player 2 is  $\neg\varphi$ . This simple framework can be extended in a straightforward way to non zero-sum  $n$ -players games (see [3], especially for complexity issues): each player  $i$  has a goal  $\varphi_i$  (a formula of  $L_V$ ). Her payoff is 1 when  $\varphi_i$  is satisfied, and 0 otherwise.

**Definition 1.** A  $n$ -players Boolean game is a 4-uple  $(A, V, \pi, \Phi)$ , where  $A = \{1, 2, \dots, n\}$  is a set of players,  $V$  is a set of propositional variables  $\pi : A \mapsto V$  is a control assignment function and  $\Phi = \langle \varphi_1, \dots, \varphi_n \rangle$  is a collection of formulas of  $L_V$ .

The control assignment function  $\pi$  associates every player with the variables that she controls. For the sake of notation, the set of all the variables controlled by  $i$  is written  $\pi_i$  instead of  $\pi(i)$ . We require that each variable be controlled by one and only one agent, i.e.,  $\{\pi_1, \dots, \pi_n\}$  forms a partition of  $V$ . The original definition by [15,14] is a special case of this more general framework, obtained by letting  $n = 2$  and  $\varphi_2 = \neg\varphi_1$ .

<sup>1</sup> The original definition in [15,14] is inductive: a Boolean game consists of a finite dynamic game. We use here the equivalent, simpler definition of [11], who showed that this tree-like construction is unnecessary.

<sup>2</sup> Stricto sensu, the obtained games are not zero-sum, but constant-sum (the sum of utilities being 1) – the difference is irrelevant and we use the terminology “zero-sum” nevertheless.

**Definition 2.** Let  $G = (A, V, \pi, \Phi)$ . A **strategy**  $s_i$  for a player  $i$  is a  $\pi_i$ -interpretation. A **strategy profile**  $S$  for  $G$  is an  $n$ -uple  $S = (s_1, s_2, \dots, s_n)$  where for all  $i$ ,  $s_i \in 2^{\pi_i}$ .

In other words, a strategy for  $i$  is a truth assignment for all the variables  $i$  controls. Remark that since  $\{\pi_1, \dots, \pi_n\}$  forms a partition of  $V$ , a strategy profile  $S$  is an interpretation for  $V$ , i.e.,  $S \in 2^V$ .  $\Omega$  denotes the set of all strategy profiles for  $G$ .

The following notations are usual in game theory. Let  $G = (A, V, \pi, \Phi)$ ,  $S = (s_1, \dots, s_n)$ ,  $S' = (s'_1, \dots, s'_n)$  be two strategy profiles for  $G$ .  $s_{-i}$  denotes the projection of  $S$  on  $A \setminus \{i\}$ :  $s_{-i} = (s_1, s_2, \dots, s_{i-1}, s_{i+1}, \dots, s_n)$ . Similarly,  $\pi_{-i}$  denotes the set of the variables controlled by all players except  $i$ :  $\pi_{-i} = V \setminus \pi_i$ . Finally,  $(s_{-i}, s'_i)$  denotes the strategy profile obtained from  $S$  by replacing  $s_i$  with  $s'_i$  without changing the other strategies:  $(s_{-i}, s'_i) = (s_1, s_2, \dots, s_{i-1}, s'_i, s_{i+1}, \dots, s_n)$ .

**Example 1.** We consider here a Boolean  $n$ -players version of the well-known prisoners' dilemma.  $n$  prisoners (denoted by  $1, \dots, n$ ) are kept in separate cells. The same proposal is made to each of them: “Either you cover your accomplices ( $C_i$ ,  $i = 1, \dots, n$ ) or you denounce them ( $\neg C_i$ ,  $i = 1, \dots, n$ ). Denouncing makes you freed while your partners will be sent to prison (except those who denounced you as well; these ones will be freed as well). But if none of you chooses to denounce, everyone will be freed.<sup>3</sup>” This can be expressed much compactly by the following  $n$ -players Boolean game  $G = (A, V, \pi, \Phi)$ :  $A = \{1, 2, \dots, n\}$ ;  $V = \{C_1, \dots, C_n\}$ ; and for every  $i \in \{1, \dots, n\}$ ,  $\pi_i = \{C_i\}$  and  $\varphi_i = (C_1 \wedge C_2 \wedge \dots \wedge C_n) \vee \neg C_i$ . Here is the representation of this game in normal form for  $n = 3$ , where in each  $(x, y, z)$ ,  $x$  – resp.  $y$ , resp.  $z$  – represents the payoff of player 1 – resp. 2, resp. 3.

		strategy of 3: $C_3$		strategy of 3: $\neg C_3$	
		$C_2$	$\neg C_2$	$C_2$	$\neg C_2$
		$C_1$	$\neg C_1$	$\neg C_1$	$C_1$
1	2	(1, 1, 1)	(0, 1, 0)	(0, 0, 1)	(0, 1, 1)
	$C_1$	(1, 0, 0)	(1, 1, 0)	(1, 0, 1)	(1, 1, 1)

The explicit representation of this game in normal form would need exponential space, which illustrates the succinctness power of a representation by Boolean games.

Each player  $i$  has two possible strategies:  $s_{i1} = \{C_i\}$ ,  $s_{i2} = \{\neg C_i\}$ . There are 8 strategy profiles for  $G$ . Consider  $S_1 = (C_1, C_2, C_3)$  and  $S_2 = (\neg C_1, \neg C_2, \neg C_3)$ . Under  $S_1$ , players 1, 2 and 3 have their goal satisfied, while  $S_2$  satisfies only Player 1's goal.

This choice of binary utilities (where agents can only express plain satisfaction or plain dissatisfaction, with no intermediate levels) is a real loss of generality. We would like now to allow for associating an arbitrary preference relation on  $\Omega$  with each player. A preference relation  $\succeq$  is a reflexive and transitive binary relation (not necessarily complete) on  $\Omega$ . The strict preference  $\succ$  associated with  $\succeq$  is defined as usual by  $S_1 \succ S_2$  if and only if  $S_1 \succeq_i S_2$  and not  $(S_2 \succeq_i S_1)$ .

A generalized Boolean game will be a 4-uple  $G = (A, V, \pi, \Phi)$ , where  $A = \{1, \dots, n\}$ ,  $V$  and  $\pi$  are as before and  $\Phi = \langle \Phi_1, \dots, \Phi_n \rangle$ , where for each  $i$ ,  $\Phi_i$  is a compact representation (in some preference representation language) of the preference relation  $\succeq_i$  of agent  $i$  on  $\Omega$ . We let  $Pref_G = \langle \succeq_1, \dots, \succeq_n \rangle$ .

<sup>3</sup> The case where everyone will be freed if everyone denounces the others is a side effect of our simplification of the prisoners' dilemma.

A pure strategy Nash equilibrium (PNE) is a strategy profile such that each player's strategy is an optimum response to the other players' strategies. However, PNEs are classically defined for games where preferences are complete, which is not necessarily the case here. Therefore we have to define *two* notions of PNEs, a weak one and a strong one (they are equivalent to the notion of maximal and maximum equilibria in [14]).

**Definition 3.** Let  $G = (A, V, \pi, \Phi)$  and  $\text{Pref}_G = \langle \succeq_1, \dots, \succeq_n \rangle$  the collection of preference relations on  $\Omega$  induced from  $\Phi$ . Let  $S = (s_1, \dots, s_n) \in \Omega$ .

$S$  is a **weak PNE (WPNE)** for  $G$  iff  $\forall i \in \{1, \dots, n\}$ ,  $\forall s'_i \in 2^{\pi_i}, (s'_i, s_{-i}) \not\succ_i (s_i, s_{-i})$ .

$S$  is a **strong PNE (SPNE)** for  $G$  iff  $\forall i \in \{1, \dots, n\}$ ,  $\forall s'_i \in 2^{\pi_i}, (s'_i, s_{-i}) \preceq_i (s_i, s_{-i})$ .

$NE_{\text{strong}}(G)$  and  $NE_{\text{weak}}(G)$  denote respectively the set of strong and weak PNEs for  $G$ .

Clearly, any SPNE is a WPNE, that is,  $NE_{\text{strong}}(G) \subseteq NE_{\text{weak}}(G)$ .

### 3 Boolean Games and Prioritized Goals

The preferences of a single player in this framework are expressed by a set of goals ordered by a priority relation:

**Definition 4.** A **prioritized goal base**  $\Sigma$  is a collection  $\langle \Sigma^1; \dots; \Sigma^p \rangle$  of sets of propositional formulas.  $\Sigma^j$  represents the set of goals of priority  $j$ , with the convention that the smaller  $j$ , the more prioritary the formulas in  $\Sigma^j$ .

In this context, several criteria can be used in order to generate a preference relation  $\succeq$  from  $\Sigma$ . We recall below the three most common ones. In the following, if  $S$  is an interpretation of  $2^V$  then we let  $\text{Sat}(S, \Sigma^j) = \{\varphi \in \Sigma^j \mid S \models \varphi\}$ .

**Definition 5.** Let  $\Sigma = \langle \Sigma^1; \dots; \Sigma^p \rangle$ , and let  $S$  and  $S'$  be two interpretations of  $2^V$ .

**Discrimin preference relation** [7,13,2]  $S \succ^{\text{disc}} S'$  iff  $\exists k \in \{1, \dots, p\}$  such that:

$$\text{Sat}(S, \Sigma^k) \supset \text{Sat}(S', \Sigma^k) \text{ and } \forall j < k, \text{Sat}(S, \Sigma^j) = \text{Sat}(S', \Sigma^j)$$

**Leximin preference relation** [10,2,17]  $S \succ^{\text{lex}} S'$  iff  $\exists k \in \{1, \dots, p\}$  such that:

$$|\text{Sat}(S, \Sigma^k)| > |\text{Sat}(S', \Sigma^k)| \text{ and } \forall j < k, |\text{Sat}(S, \Sigma^j)| = |\text{Sat}(S', \Sigma^j)|.$$

**Best-out preference relation** [10,2] Let  $a(s) = \min\{j \text{ such that } \exists \varphi \in \Sigma^j, S \not\models \varphi\}$ , with the convention  $\min(\emptyset) = +\infty$ . Then  $S \succeq^{\text{bo}} S'$  iff  $a(S) \geq a(S')$ .

Note that  $\succeq^{\text{bo}}$  and  $\succeq^{\text{lex}}$  are complete preference relations, while  $\succeq^{\text{disc}}$  is generally a partial preference relation. Moreover, the following implications hold (see [2]):

$$(S \succ^{\text{bo}} S') \Rightarrow (S \succ^{\text{discr}} S') \Rightarrow (S \succ^{\text{lex}} S') \quad (1) \quad (S \succeq^{\text{discr}} S') \Rightarrow (S \succeq^{\text{lex}} S') \Rightarrow (S \succeq^{\text{bo}} S') \quad (2)$$

**Definition 6.** A **PG-Boolean game** is a 4-uple  $G = (A, V, \pi, \Phi)$ , where  $\Phi = (\Sigma_1, \dots, \Sigma_n)$  is a collection of prioritized goals bases. We denote  $\Sigma_i = \langle \Sigma_i^1, \dots, \Sigma_i^p \rangle$ , that is,  $\Sigma_i^j$  denotes the stratum  $j$  of  $\Sigma_i$ , or equivalently, the (multi)set of goals of priority  $j$  for player  $i$ .

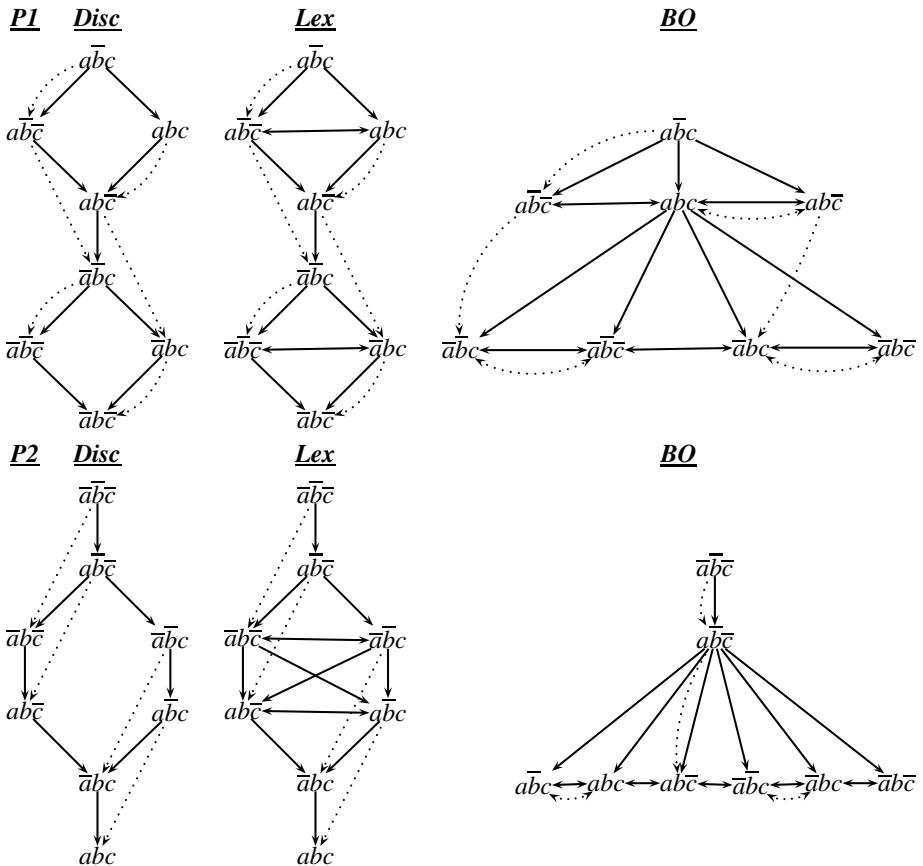
Note that the assumption that the number of priority levels is the same ( $p$ ) for all players does not imply a loss of generality, as adding empty strata to a prioritized base does not change the induced preference relation.

We make use of the following notations:

- if  $G$  is a PG-boolean game and  $c \in \{disc, lex, bo\}$  then  $Pref_G^c = \langle \succeq_1^c, \dots, \succeq_n^c \rangle$ .
- $NE_{weak}^c(G)$  and  $NE_{strong}^c(G)$  denote respectively the sets of all weak and strong Nash equilibria for  $Pref_G^c$ .

**Example 2.** Let  $G = (A, V, \pi, \Phi)$  with  $A = \{1, 2\}$ ,  $V = \{a, b, c\}$ ,  $\pi_1 = \{a, c\}$ ,  $\pi_2 = \{b\}$ ,  $\Sigma_1 = \langle a; (\neg b, c) \rangle$ ,  $\Sigma_2 = \langle (\neg b, \neg c); \neg a \rangle$ .

For each of the three criteria  $c \in \{lex, disc, bo\}$ , we draw the corresponding preference relations  $Pref_G^c = \langle \succeq_1^c, \dots, \succeq_n^c \rangle$ . The arrows are oriented from more preferred to less preferred strategy profiles ( $S_1$  is preferred to  $S_2$  is denoted by  $S_1 \rightarrow S_2$ ). To make the figures clearer, we do not draw edges that are obtained from others by transitivity. The dotted arrows indicate the links taken into account in order to compute Nash equilibria.



- **Discrimin and Leximin:**  $NE_{weak}^{disc}(G) = NE_{strong}^{disc}(G) = \{\bar{abc}\}$
- **Best Out:**  $NE_{weak}^{bo}(G) = NE_{strong}^{bo}(G) = \{abc, \bar{abc}\}$

**Lemma 1.** Let  $\succeq = \langle \succeq_1, \dots, \succeq_n \rangle$  and  $\succeq' = \langle \succeq'_1, \dots, \succeq'_n \rangle$  be two collections of preference relations, and let  $S$  be a strategy profile.

1. If  $\succeq$  is contained in  $\succeq'$  and if  $S$  is a SPNE for  $\succeq$ , then  $S$  is a SPNE for  $\succeq'$ .
2. If  $\succ$  is contained in  $\succ'$  and if  $S$  is a WPNE for  $\succ'$ , then  $S$  is a WPNE for  $\succ$ .

This lemma enables us to draw the following:

**Proposition 1.** Let  $G = (A, V, \pi, \Phi)$  be a PG-boolean game and  $\text{Pref}_G^c = \langle \succeq_1^c, \dots, \succeq_n^c \rangle$ .  $\text{NE}_{\text{strong}}^{\text{disc}}(G) \subseteq \text{NE}_{\text{strong}}^{\text{lex}}(G) \subseteq \text{NE}_{\text{strong}}^{\text{bo}}(G)$  and  $\text{NE}_{\text{weak}}^{\text{lex}}(G) \subseteq \text{NE}_{\text{weak}}^{\text{disc}}(G) \subseteq \text{NE}_{\text{weak}}^{\text{bo}}(G)$ .

We may now wonder whether a PG-boolean game can be *approximated* by focusing on the first  $k$  strata of each player. Here, the aim is double: to obtain a simpler (for PNE computation) game and to increase the possibility to find a significant PNE taking into account the most prioritized strata.

**Definition 7.** Let  $G = (A = \{1, \dots, n\}, V, \pi, \Phi)$  be a PG-boolean game, and  $k \in \{1, \dots, p\}$ .  $G^{[1 \rightarrow k]} = (A, V, \pi, \Phi^{[1 \rightarrow k]})$  denotes the  **$k$ -reduced game** of  $G$  in which all players' goals in  $G$  are reduced in their  $k$  first strata:  $\Phi^{[1 \rightarrow k]} = \langle \Sigma_1^{[1 \rightarrow k]}, \dots, \Sigma_n^{[1 \rightarrow k]} \rangle$ .

**Lemma 2.** Let  $G$  be a PG-boolean game. Then for every  $k \leq p$ ,  $c \in \{\text{discr}, \text{lex}, \text{bo}\}$ , and every  $i \in A$ , we have:  $S \succeq_i^{c, [1 \rightarrow k]} S' \Rightarrow S \succeq_i^{c, [1 \rightarrow k-1]} S'$  and  $S \not\succ_i^{c, [1 \rightarrow k]} S' \Rightarrow S \not\succ_i^{c, [1 \rightarrow k-1]} S'$ .

**Proposition 2.** Let  $G$  be a PG-boolean game and  $c \in \{\text{discr}, \text{lex}, \text{bo}\}$ . If  $S$  is a SPNE (resp. WPNE) for  $\text{Pref}_{G^{[1 \rightarrow k]}}^c$  of the game  $G^{[1 \rightarrow k]}$ , then  $S$  is a SPNE (resp. WPNE) for  $\text{Pref}_{G^{[1 \rightarrow (k-1)]}}^c$  of the game  $G^{[1 \rightarrow (k-1)]}$ .

This proposition leads in an obvious way that if  $G^{[1]}$  for  $\text{Pref}_{G^{[1]}}^c$  does not have any SPNE (resp. WPNE), then the game  $G$  for  $\text{Pref}_G^c$  does not have any SPNE (resp. WPNE) whatever the criteria used. The converse is false, as shown in the following example.

**Example 3.** Let  $G$  with  $A = \{1, 2\}$ ,  $V = \{a, b\}$ ,  $\pi_1 = \{a\}$ ,  $\pi_2 = \{b\}$ ,  $\Sigma_1 = \langle a \rightarrow b; b \rightarrow a \rangle$ ,  $\Sigma_2 = \langle a \leftrightarrow \neg b; \neg b \rangle$ . We check that  $\text{NE}_{\text{weak}}^{\text{bo}}(G) = \text{NE}_{\text{strong}}^{\text{bo}}(G) = \emptyset$ . Let us now focus on the 1-reduced game  $G^{[1]} = (A, V, \pi, \Phi^{[1]})$  of  $G$ . We have  $\Sigma_1^{[1]} = \langle a \rightarrow b \rangle$ ,  $\Sigma_2^{[1]} = \langle a \leftrightarrow \neg b \rangle$ . We check that for any criterion  $c$ ,  $\text{NE}_{\text{weak}}^c(G^{[1]}) = \text{NE}_{\text{strong}}^c(G^{[1]}) = \{\bar{a}b\}$ .

This example shows us that Proposition 2 can be used to find the right level of approximation for a PG-game. For instance, we may want to focus on the largest  $k$  such that  $G^{[1 \rightarrow k]}$  has a SPNE, and similarly for WPNEs.

## 4 Boolean Games and CP-Nets

A problem with prioritized goals is the difficulty for the agent to express his preferences (from a cognitive or linguistic point of view). In this Section we consider another very popular language for compact preference representation on combinatorial domains, namely CP-nets. This graphical model exploits conditional preferential independence in order to structure decision maker's preferences under a *ceteris paribus* assumption. They were introduced in [6] and extensively studied in many subsequent papers, especially [4,5].

Although CP-nets generally consider variables with arbitrary finite domains, for the sake of simplicity (and homogeneity with the rest of the paper) here we consider only “propositionalized” CP-nets, that is, CP-nets with binary variables (note that this is not a real loss of generality, as all our definitions and results can be easily lifted to the more general case of non-binary variables).

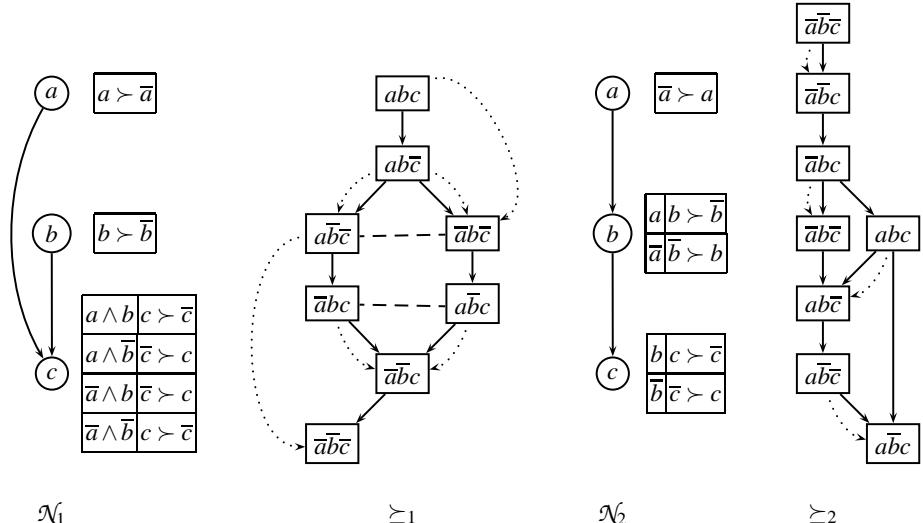
**Definition 8.** Let  $V$  be a set of propositional variables and  $\{X, Y, Z\}$  a partition of  $V$ .  $X$  is **conditionally preferentially independent** of  $Y$  given  $Z$  if and only if  $\forall z \in 2^Z$ ,  $\forall x_1, x_2 \in 2^X$  and  $\forall y_1, y_2 \in 2^Y$  we have :  $x_1y_1z \succeq x_2y_1z$  iff  $x_1y_2z \succeq x_2y_2z$ .

For each variable  $X$ , the agent specifies a set of *parent variables*  $\text{Pa}(X)$  that can affect her preferences over the values of  $X$ . Formally,  $X$  and  $V \setminus (\{X\} \cup \text{Pa}(X))$  are conditionally preferentially independent given  $\text{Pa}(X)$ . This information is used to create the CP-net:

**Definition 9.** Let  $V$  be a set of variables.  $\mathcal{N} = \langle \mathcal{G}, \mathcal{T} \rangle$  is a **CP-net on  $V$** , where  $\mathcal{G}$  is a directed graph over  $V$ , and  $\mathcal{T}$  is a set of conditional preference tables  $CPT(X_j)$  for each  $X_j \in V$ . Each  $CPT(X_j)$  associates a total order  $\succ_p^j$  with each instantiation  $p \in 2^{\text{Pa}(X_j)}$ .

**Definition 10.** A **CP-boolean game** is a 4-uple  $G = (A, V, \pi, \Phi)$ , where  $A = \{1, \dots, n\}$  is a set of players,  $V = \{x_1, \dots, x_p\}$  is a set of variables and  $\Phi = \langle \mathcal{N}_1, \dots, \mathcal{N}_p \rangle$ . Each  $\mathcal{N}_i$  is a CP-net on  $V$ .

**Example 4.**  $G = (A, V, \pi, \Phi)$  where  $A = \{1, 2\}$   $V = \{a, b, c\}$   $\pi_1 = \{a, b\}$ ,  $\pi_2 = \{c\}$ ,  $\mathcal{N}_1$  and  $\mathcal{N}_2$  are represented on the following figure.



Using these partial pre-orders, Nash equilibria are:  $NE_{\text{strong}} = NE_{\text{weak}} = \{abc\}$ .

The first property concerns a very interesting case where the existence and the unicity of PNE hold:

**Proposition 3.** Let  $G = (A, V, \pi, \Phi)$  be a CP-boolean game such the graphs  $\mathcal{G}_i$  are all identical ( $\forall i, j, \mathcal{G}_i = \mathcal{G}_j$ ) and acyclic. Then  $G$  has one and only one strong PNE.

The proof of this result makes use of the *forward sweep* procedure [6,4] for outcome optimization (this procedure consists in instantiating variables following an order compatible with the graph, choosing for each variable its preferred value given the value of the parents).

The point is that in general the graphs  $\mathcal{G}_i$  for  $i \in \{1, \dots, n\}$  may not be identical. However, they may be *made* identical, once remarked that a CP-net  $\langle \mathcal{G}, \mathcal{T} \rangle$  can be expressed as a CP-net  $\langle \mathcal{G}', \mathcal{T}' \rangle$  as soon as the set of edges in  $\mathcal{G}$  is contained in the set of edges in  $\mathcal{G}'$ . We may then take as common graph  $\mathcal{G}$  (to all players) the graph whose set of edges is the *union* of the set of edges of  $\mathcal{G}_1, \dots, \mathcal{G}_n$ . The only problem is that the resulting graph may not be acyclic, in which case Proposition 3 is not applicable. Formally:

**Definition 11.** Let  $G$  be a CP-boolean game. For each player  $i$ ,  $\mathcal{G}_i$  is denoted by  $(V, Arc_i)$ , with  $Arc_i$  being the set of edges of  $i$ 's CP-net. The **union graph** of  $G$  is defined by  $\mathcal{G} = (V, Arc_1 \cup \dots \cup Arc_n)$ . The **normalized game equivalent to  $G$** , denoted by  $G^* = \{A, V, \pi, \Phi^*\}$ , is the game obtained from  $G$  by rewriting, where the graph of each player's CP-net has been replaced by the graph of the union of CP-nets of  $G$  and the CPT of each player's CP-net are modified in order to fit with the new graph, keeping the same preferences (formally, if  $\succ_i^y$  denotes the relation associated with  $CPT_i(y)$  for Player  $i$ 's CP-net in  $G$ , then we have for  $G^*$ :  $\forall x \in V$  such that  $x$  is a parent of  $y$  in  $G^*$  but not in  $G$ ,  $\succ_{i,x}^y = \succ_{i,\bar{x}}^y = \succ_i^y$ ).

The following lemma is straightforward:

**Lemma 1.** Let  $G$  be a CP-boolean game and  $G^*$  its equivalent normalized game. Then  $G^*$  and  $G$  define the same preference relations on strategy profiles.

Therefore, if  $G^*$  is acyclic, then Proposition 3 applies, therefore  $G^*$  has one and only one SPNE. Now, since  $G$  and  $G^*$  define the same pre-orders on  $\Omega$ , the latter is also the only SPNE of  $G$  (on the other hand, if the graph of  $G$  is cyclic, neither the unicity nor the existence of SPNEs is guaranteed).

**Proposition 4.** Let  $G = (A, V, \pi, \Phi)$  be a CP-boolean game. If the union graph of  $G$  is acyclic then  $G$  has one and only one strong PNE.

**Example 4, continued:** Players' preferences in the normalized game  $G^*$  (equivalent to  $G$ ) are represented by the CP-nets given on Figure 1. The union graph is acyclic, therefore Proposition 3 can be applied and  $G$  has one and only one strong PNE (abc).

There is a last condition (less interesting in practice because it is quite strong) guaranteeing the existence and the unicity of a SPNE. This condition states that any variable controlled by an agent is preferentially independent on variables controlled by other agents (in other words, the parents of any variable controlled by a player  $i$  are also controlled by  $i$ ). In this case, each agent is able to instantiate her variables in an unambiguously optimal way, according to her preferences.

**Proposition 5.** Let  $G = (A, V, \pi, \Phi)$  be a CP-boolean game such that for every player  $i \in A$  and for every  $v \in \pi_i$ , we have  $Pa(v) \subseteq \pi_i$ . Then  $G$  has one and only one SPNE.

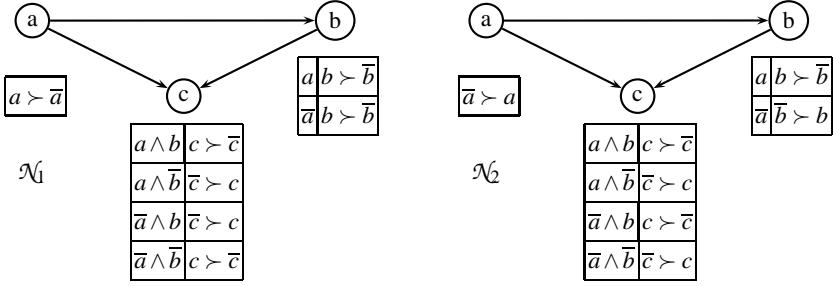


Fig. 1. CP-net of Players 1 and 2’s preferences for  $G^*$

## 5 Related Work and Conclusion

Apart of previous work on Boolean games [15, 14, 11], related work includes a few papers where games are expressed within ordinal preferences within well-developed AI frameworks.

In [12], a game in normal form is mapped into a *logic program with ordered disjunction* (LPOD) where each player owns a set of clauses that encode the player’s preference over her possible actions given every possible strategy profile of other players. It is shown that PNE correspond exactly to the most preferred answer sets. The given translation suffers from a limitation, namely its size: the size of the LPOD is the same as that of the normal form of the game (since each player needs a number of clauses equal to the number of possible other strategy profiles for other players). However, this limitation is due to the way LPODs are induced from games and could be overwhelmed by allowing to express the players’ preferences by any LPODs (in the same spirit as our Section 3).

In [9], a strategic game is represented using a *choice logic program*, where a set of rules express that a player will select a “best response” given the other players’ choices. Then, for every strategic game, there exists a choice logic program such that the set of stable models of the program coincides with the set of Nash equilibria of the game. This property provides a systematic method to compute Nash equilibria for finite strategic games.

In [1], CP-nets are viewed as games in normal form and vice versa. Each player  $i$  corresponds to a variable  $X_i$  of the CP-net, whose domain  $D(X_i)$  is the set of available actions to the player. Preferences over a player’s actions given the other players’ strategies are then expressed in a conditional preference table. The CP-net expression of the game can sometimes be more compact than its normal form explicit representation, provided that some players’ preferences depend only on the actions of a subset of other players. A first important difference with our framework is that we allow players to control an arbitrary set of variables, and thus we do not view players as variables; the only way of expressing in a CP-net that a player controls several variables would consist in introducing a new variable whose domain would be the set of all combination of values for these variables—and the size of the CP-net would then be exponential in the number of variables. A second important difference, which holds as well for the

comparison with [12] and [9], is that players can express arbitrary preferences, including extreme cases where the satisfaction of a player's goal may depend only of variables controlled by other players. A last (less technical and more foundational) difference with both lines of work, which actually explains the first two above, is that we do not *map* normal form games into anything but we *express* games using a logical language.

Further work includes the investigation of other notions (such as dominated strategies) within the two frameworks proposed in this paper, as well as the integration of other preference representation languages within Boolean games.

## References

1. K. R. Apt, F. Rossi, and K. B. Venable. CP-nets and Nash equilibria. In Elsevier, editor, *Proc. CIRAS 2005 (Third International Conference on Computational Intelligence, Robotics and Autonomous Systems)*, Singapore, December 13-16 2005.
2. S. Benferhat, C. Cayrol, D. Dubois, J. Lang, and H. Prade. Inconsistency management and prioritized syntax-based entailment. In *Proc. of the 13th IJCAI*, pages 640–645, 1993.
3. E. Bonzon, MC. Lagasquie-Schiex, J. Lang, and B. Zanuttini. Boolean games revisited. available at <ftp://ftp.irit.fr/pub/IRIT/RPDMF/ecai06.ps.gz>, 2006.
4. C. Boutilier, R. Brafman, C. Domshlak, H. Hoos, and D. Poole. Cp-nets: A tool for representing and reasoning with conditional *Ceteris Paribus* preference statements. *Journal of Artificial Intelligence Research*, 21:135–191, 2004.
5. C. Boutilier, R. Brafman, C. Domshlak, H. Hoos, and D. Poole. Preference-based constrained optimization with cp-nets. *Computational Intelligence*, 20(2):137–157, 2004. Special Issue on Preferences.
6. C. Boutilier, R. I. Brafman, H. H. Hoos, and D. Poole. Reasoning with conditional *ceteris paribus* preference statements. In *Proc. of UAI*, 1999.
7. G. Brewka. Preferred subtheorie : An extended logical framework for default reasoning. In *Proc. of the 11th IJCAI*, pages 1043–1048, 1989.
8. S. Coste-Marquis, J. Lang, P. Liberatore, and P. Marquis. Expressive power and succinctness of propositional languages for preference representation. In *Proc. of the 9th KR*, pages 203–212, 2004.
9. M. De Vos and D. Vermeir. Choice logic programs and Nash equilibria in strategic games. In Jorg Flum and Mario Rodriguez-Artalejo, editors, *Computer Science Logic (CSL'99)*, volume 1683, pages 266–276, 1999.
10. D. Dubois, J. Lang, and H. Prade. Inconsistency in possibilistic knowledge bases: To live with it or not live with it. In *Fuzzy Logic for the Management of Uncertainty*, pages 335–351. Wiley and sons, 1992.
11. P.E. Dunne and W. van der Hoek. Representation and complexity in boolean games. In *Proc. of JELIA, LNCS 3229*, pages 347–359, 2004.
12. N. Foo, T. Meyer, and G. Brewka. LPOD answer sets and Nash equilibria. In M. Maher, editor, *Proceedings of the 9th Asian Computer Science Conference (ASIAN 2004)*, pages 343–351. Chiang Mai, Thailand, Springer LNCS 3321, 2004.
13. H. Geffner. Default reasoning: causal and conditional theories. *MIT Press*, 1992.
14. P. Harrenstein. *Logic in Conflict*. PhD thesis, Utrecht University, 2004.
15. P. Harrenstein, W. van der Hoek, J.J Meyer, and C. Witteveen. Boolean games. In *Proc. of TARK*, pages 287–298, 2001.
16. J. Lang. Logical preference representation and combinatorial vote. *Annals of Mathematics and Artificial Intelligence*, 42:37–71, 2004.
17. D Lehmann. Another perspective on default reasoning. *Annals of Mathematics and Artificial Intelligence*, 15:61–82, 1995.

# Agent-Based Flexible Videoconference System with Automatic QoS Parameter Tuning

Sungdoke Lee<sup>1</sup>, Sanggil Kang<sup>2</sup>, and Dongsoo Han<sup>1</sup>

<sup>1</sup> School of Engineering, Information and Communication University  
P.O. Box 77, Yuseong, Daejeon, 305-600 Korea  
[{sdlee, dshan}@icu.ac.kr](mailto:{sdlee, dshan}@icu.ac.kr)

<sup>2</sup> Computer Science, Engineering of Information Technology  
The University of Suwon  
San 2-2, Wau-ri, Bongdam-eup, Hwaseong, Gyeonggi-do, 445-743 Korea  
[sgkang@suwon.ac.kr](mailto:sgkang@suwon.ac.kr)

**Abstract.** In this paper, we propose a new agent-based flexible videoconference system (AVCS) by modifying videoconference manger (VCM) agent in a conventional flexible videoconference system (FVCS). The proposed AVCS can more flexibly cope with changes in working conditions during videoconferencing than the conventional FVCS. It is because an automatic parameter tuning algorithm is imbedded to VCM dynamically adapt QoS (Quality of Service) parameters in the sense that the current working condition can meet with the desired working condition of the user, which can change in time during videoconferencing. In the experimental section, we design a new structure of the VCM with the automatic parameter tuning module, imbed to the prototype of FVCS and implement the new AVCS. Also, it is shown that the proposed AVCS outperforms the existing FVCS in the experiment.

## 1 Introduction

To maintain a stable conference session during videoconferencing [1]-[3] on various computer systems and network environments, users have to consider the working condition such as status of system resources on not only his/her side but also the other end, the variation on network service, and etc. A manual adjustment of system parameters for keeping the conference stable sometimes burdens to novice users. In order to reduce these burdens, the flexible videoconference system (FVCS) [4]-[7], which is a user support environment for videoconference based on a multi-agent system, has been studied by a few experts including us, to our best knowledge. FVCS can be modeled by adding some flexible features to a traditional videoconference system (VCS). FVCS can flexibly cope with the changes in user's requirement and system/network environment by adapting its quality of service (QoS). Existing adaptive QoS control mechanisms adjust the QoS parameters<sup>1</sup> such as smoothness, quality, and resolution by the strategic

---

<sup>1</sup> In this paper, "QoS parameter" and "parameter" are exchangeable for the convenience.

knowledge stored in an agent. We can catch a several drawbacks from the mechanisms such as; one is that their problem solving fashion is static because they cannot repeat the process of the adjustment at the same session. The other is that they cannot automatically cope with the changes of a user's preference of the parameters during videoconferencing.

In order to overcome the drawbacks, we modify the architecture of the video-conference manager (VCM) in conventional FVCS and develop an automatic parameter tuning module (APTM) which can reside in VCM. The APTM dynamically tunes the parameters in the sense that the current working condition can meet with the desired working condition of the user which usually changes in time during videoconferencing. Using the gradient method [8], the parameters are trained in the space of parameters until a set of optimal values of the parameters is obtained. In APTM, the parameters are differently tuned according to the order of users' preference of the parameters. In the experimental section, we design a new structure of the VCM with APTM imbedded to the prototype of FVCS and implement the proposed agent-based flexible videoconference system (AVCS). Also, it is shown that of the proposed AVCS outperforms the existing FVCS in the experiment.

The remainder of this paper is organized as follows. Section 2 overviews the background of videoconference systems and related work, which can help to understand our system. Section 3 explains our proposed architecture of VCM in the AVCS. In Section 4, we demonstrate our automatic parameter tuning algorithm. In Section 5, we design the proposed AVCS and analyze the operational situation. We also demonstrate the experimental results by comparing with existing FVCS. In Section 6, we draw our conclusion.

## 2 Background and Related Work

### 2.1 Videoconference Systems

In traditional VCS, there are a couple of main methods for QoS control on application level: one is IVS [1] and the other is framework-based [2]. IVS was developed for the purpose of videoconferencing over Internet. It controls outgoing data transmission rate based on information about change in network condition by adjusting the parameters of video coders. Also, it can accept simple user requirement by defining policy on QoS control. Framework-based approach provides two fundamental challenges for constructing "network aware" application: one is how to find out dynamic changes of quality in network service, the other is how to transfer network-centric quality measures to application-centric ones.

The two methods explained above are simple to execute, however there are two critical limitations in them as following: (L1) Flexibility - They are designed for a specific network environment and their QoS control mechanism is simple, which are the main reasons why they have a difficulty in flexibility for an unexpected situation. (L2) Cooperation - In the traditional VCS, both sides of users do not exchange the information about their working condition such as the status of CPU resource, network condition, user requirement of QoS, and etc. at the

other side. By cooperating during videoconferencing, a stable conference can be flexibly maintained from help of the other side in the case that the computer in his/her side can not adapt for an unexpected situation. The cooperation between two sides can be done by the cooperation protocol [5].

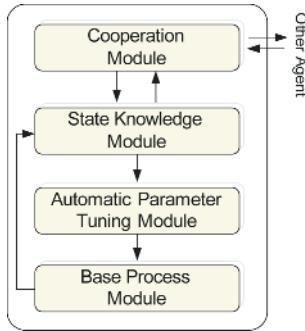
In order to overcome the limitations of the traditional QoS control mechanism, the multiagent-based flexible videoconference system has been developed, which is explained in the following section.

## 2.2 Flexible Videoconference System

Conventional FVCS has been promoted aiming at providing a user-centered communication environment based on agent-based computing technology [4]-[7]. The objective of the FVCS is to reduce lots of users' overloads in utilizing traditional VCSs by effective use of traditional VCS software and expertise of designers/operators of VCSs. To lighten users' burdens of VCSs, FVCS is attained by embedding the following functionality to the existing VCSs, i.e., (F1) service configuration function at the start of a session and (F2) service tuning function during the session. Here, (F1) composes the most suitable service configuration of VCS automatically by selecting best software modules and deciding their set-up parameters under the given conditions of environments and users. (F2) adjusts the QoS autonomously according to the changes of network/computational environments or user requirements against the QoS. The function can be realized by two phase tuning operations, i.e., parameter operation tuning for small-scale changes and reconfiguration of videoconference service for large-scale changes.

In the FVCS, there are four parts such as Videoconference Manager (VCM) Agent, User Agent, Sensor Agent, and Service Agent. User Agent monitors the requirement of a user. The Sensor Agent monitors the status of network environment and CPU resource. The Service Agent executes the videoconference services such as video, audio, whiteboard, etc., in accordance with the requirements from the User Agent or the VCM Agent. The VCM Agent plays a role of management of other agents, e.g., analysis of other agents' working conditions, control of message flow among agents, cooperation between two VCM agents (Video-Conf-Manager-A and Video-Conf-Manager-B), and tuning QoS parameters (F2). In general, the QoS parameter tuning is driven by the following steps: 1) some changes during videoconference are detected by the Sensor Agents or the User Agents, and it is reported to the VCM Agents. 2) The VCM Agents negotiate each other to decide suitable operation against the videoconference process agents. 3) The videoconference process agents set parameters. 4) The Sensor Agents check recovery status and report it to the VCM. 5) Repeat from 2) to 4) until the changes are recovered. In existing the VCM Agents, the rule-based programming language method [4]-[7] is used for adjusting the QoS parameters.

The rule-based programming language method is performed by the knowledge base selection algorithm. It selects an appropriate value of each parameter with the trial and error fashion in the knowledge database. There are several drawbacks from the knowledge base selection algorithm (KBSA) such as: 1) Designers must know ahead of time all possible domain knowledge, i.e., the combination



**Fig. 1.** The model of VCM Agent

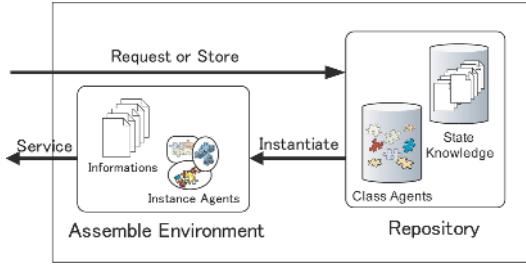
of the values of the QoS parameters. 2) The process of selecting QoS parameters cannot be repeated at the same session. 3) The process cannot automatically cope with the change of a user's preference of QoS parameters during videoconferencing. In order to solve the problems, we develop an automatic parameter tuning algorithm and design a new VCM architecture in accordance with our algorithm which will be explained in detail in the following section.

### 3 Architecture of VCM Agent

In this section, we design a model of the VCM agent in accordance with developing the automatic parameter tuning algorithm explained in detail in the following. As viewed in Fig. 1, the VCM is composed of four parts such as Cooperation Module (CM), State Knowledge Module (SKM), Automatic Parameter Tuning Module (APTM), and Base Process Module (BPM).

The Cooperation Module (CM) exchanges the information of the working conditions such as available CPU resource, bandwidth, user requirement, etc. with the SKM on a regular base with specific protocols. According to the information from SKM, the CM decides whether it requests the cooperation with the CM at the other user side. If the CM is not able to resolve the change in working condition for itself in a time limit, it sends message for cooperation to the CM in the other VCM Agent at the other end using a specific protocol.

The State Knowledge Module (SKM) plays a role on not only detecting any change in working network environment but also storing it to the repository and informing it to the CM. If a change is detected on the working environment (or the status of computer resource) in its own computer, it notifies the change to the APTM and lets the APTM activate. Also, a user's specific requirement for quality service is propagated to the APTM via the SKM. As seen in Fig. 2, the SKM is composed of two parts: one is the assemble environment and the other is the repository. The assemble environment sets up the initialization of the videoconferencing, based the information of working environment obtained from the previous videoconferences stored in the repository. During the videoconferencing, it can detect any change in the environment and send it to the



**Fig. 2.** The structure of State Knowledge Module

repository. If the QoS parameters need to be adjusted, the information of the status of the environment is sent to the APTM. Then, the APTM tunes the QoS parameters in order to maintain an appropriate working environment by tuning the parameters. The algorithm of the automatic tuning algorithm is explained in the following section.

The Base Process Module (BPM) plays a role on controlling the service agents including video, audio, and whiteboard.

## 4 Automatic Parameter Tuning Algorithm

Based on the information sent from the SKM, the APTM activates its parameter tuning algorithm. The parameters are differently tuned according to the order of user's preference of the parameters in the sense that the working condition (CPU resource, network bandwidth, and etc.) meets to the desired working condition which can changes by users during videoconferencing. Using the gradient ascent or descent method, the parameters are trained in the space of parameters until a set of optimal values of parameters is obtained. The training is done with respect to obtaining an acceptable working condition as iterating the tuning process over the parameter space, whenever the current working condition changes. The adjustment is done by an automatic scheme demonstrated in the following.

Let  $d(t)$  and  $y(t)$  denote a desired working condition (or an acceptable working condition) and the current working condition at time  $t$ , respectively.  $y(t)$  can be expressed in a function of a set of QoS parameters because the current resource depends on the values of the parameters as in Eq. (1). The available resource decreases if the values of parameters increase, and vice versa.

$$y(t) = f(A(t)) \quad (1)$$

where  $A(t) = [a_1(t), a_2(t), \dots, a_i(t), \dots, a_n(t)]$  which is a set of parameters and  $a_i(t)$  is parameter  $i$  at time  $t$ . The error, denoted as  $e(t)$ , between the desired working condition and that of current working condition at time  $t$  can be expressed as:

$$e(t) = |d - y(t)| \quad (2)$$

As seen in Eq. (2), the value of  $e(t)$  depends on the value of  $y(t)$  depending on the value of  $A(t)$ . The parameters have to be adjusted to the values with which the error becomes to zero or close to it. The adjusted parameters at the first iteration can be expressed as  $A(t+1) = [a_1(t+1), a_2(t+1), \dots, a_i(t+1), \dots, a_n(t+1)]$ , where "1" in the parenthesis means the first iteration of the adjustment and  $a_i(t+1)$  can be expressed as below:

$$\Delta a_i(t+1) = a_i(t) + \Delta a_i(t) \quad (3)$$

where  $\Delta a_i(t)$  is the amount of adjustment for parameter  $a_i(t)$  at the iteration. For the convenience,  $\Delta a_i(t)$  is set to a predetermined value of ratio with respect to  $a_i(t)$ , i.e.,

$$\Delta a_i(t+1) = \rho_i \cdot a_i(t) \quad (4)$$

where  $\rho_i$  is a scaling factor. The size of the scaling factor implies the preference of the parameters by the user. If the value of the scaling factor of a parameter is larger, the user requires more high quality of service than others. For example, if the order of the preference of the parameters from the lowest to the highest is set to  $a_1(t), a_2(t), \dots, a_i(t), \dots, a_n(t)$  then their scaling factors are ordered in  $\rho_n, \rho_{n-1}, \dots, \rho_i, \dots, \rho_1$ . The value of scaling factor according to the preference of the parameters can be obtained from exhaustive empirical experience, which is demonstrated in the experimental section. Also, as seen in Eq. (3) and (4), the sign of the scaling factor implies a search direction in the parameter space. If the sign of a scaling factor is the positive, the parameter is adaptive to the direction in which the value of the parameter increases, and vice versa. During the tuning process, the sign can be varied, which make our tuning algorithm be capable to tune dynamically the parameters to desired working condition varying more than one time at one session of videoconference. Once the first iteration is done, the current working condition and the error can be expressed as

$$y(t+1) = f(A(t+1)) \quad (5)$$

and

$$e(t+1) = |d(t) - y(t+1)| \quad (6)$$

If  $e(t+1)$  is reached to the satisfactory value then it ends the tuning process, otherwise the process of the adjustment continues until the error is tolerable. As seen in the equations from (4) to (6), the value of the scaling factor is the critical for the tuning process. If the scaling factor is too large then the speed of the tuning can be fast but the tuning process can be oscillated from the optimal points of the parameters as iteration goes on. If the scaling factor is too small then the speed of the tuning can be too slow but it can converge to a set of optimal values of the parameters. In order to avoid the problem, the amount of adjustment decreases as the iteration goes as seen in Eq. (7).

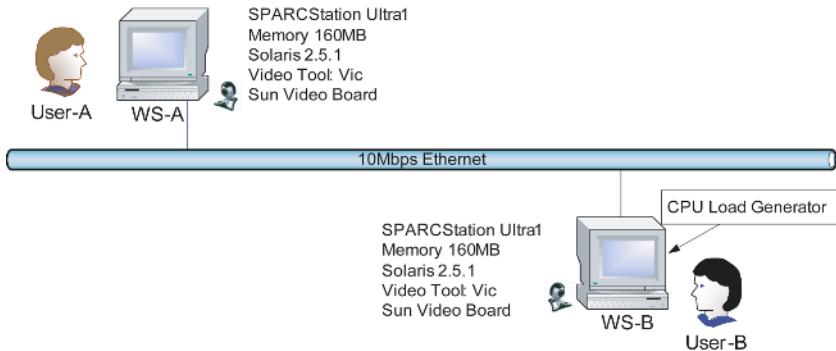
$$\Delta a_i(t+k) = \Delta a_i(t+k-1)/2^{k-1} \quad (7)$$

where  $k$  is the number of iteration.

## 5 Experiment and Analysis

### 5.1 Experimental Environment

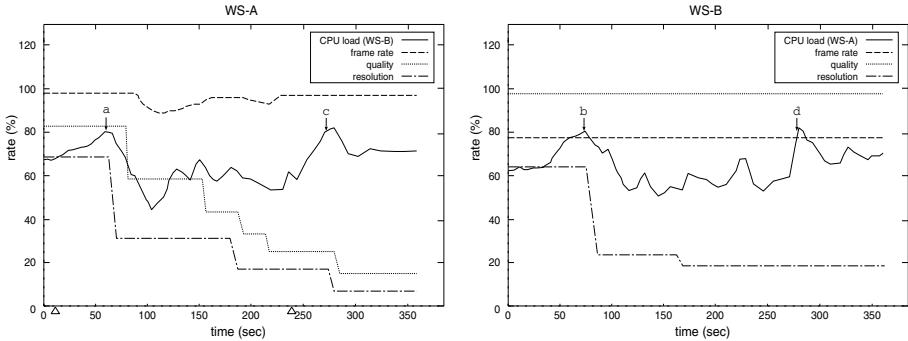
The proposed architecture of the VCM based on the automatic parameter tuning algorithm is embedded to the AVCS and is implemented in the experimental environment viewed in Fig. 3. We designed a new VCM agent by modifying the parts of the repository and the knowledge base in ADIPS Framework [9] as an agent-based computing infrastructure. In BPM, videoconference-purpose vic [3] is used for controlling the service agents in accordance with the tuned QoS parameters such as frame rate, quality, and resolution. Video-A agent and Video-B agent out of service agents provide the information of vic to the VCM agent for the APTM to execute the tuning process. Also, the CPU-Monitor agent uses Unix OS standard command "sar" for obtaining the information of CPU in order to investigate the status of CPU. The obtained information is sent to the VCM agent. The VCM agent and Sensor agent, and Service agent were written in TCI/Tk [10]. Also, SPARCstation Ultra1 (CPU clock 200MHz) was used as the hardware of the agent workspace and the videoconference terminal. The total size of the agents is about 1,800 lines.



**Fig. 3.** The experimental environment

### 5.2 Experimental Results by Variation of CPU Resources

In this section, we observe the behaviors of the proposed AVCS and the conventional FVCS according to CPU resource or user requirement change during videoconferencing. We add some extra load on CPU of WS-B at the side of User-B by a CPU load generator and observe the changes of QoS parameters (frame rate, encoding quality and resolution) of video processes serviced to User A or User B. For WS-A on the side of User-A, frame rate in movement of video is the highest priority of the preference parameter, video quality is the second highest priority, and video resolution is the lowest priority. On the other hand, for WS-B on the side of User-B, video quality is the highest priority, frame rate is the

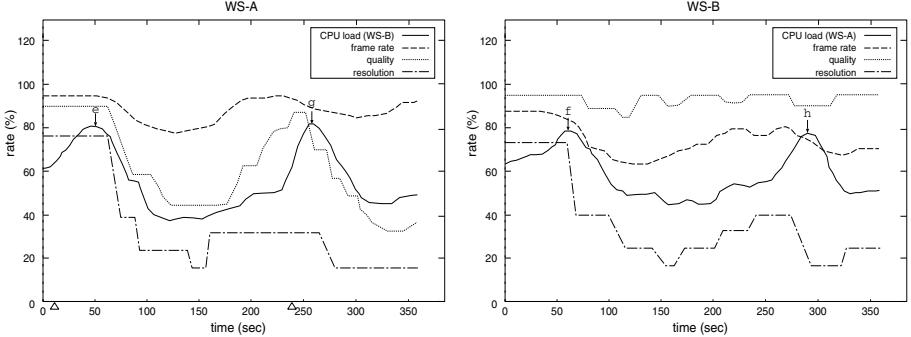


**Fig. 4.** The behavior of the conventional FVCS according to the variation of CPU load:  
(a) Change of QoS at User-A, (b) Change of QoS at User-B

second highest priority, and resolution is the lowest priority. Fig. 4 and 5 represent the transition of the parameters' values controlled by VCM agents. In the graph, x-axis represents the time (second) and y-axis represents each parameter value observed at the recipient site. The parameter values can be expressed in terms of percentage in the case that the following values are regarded as 100%; CPU load: 100%, Frame rate: 30-fps, Quality: 32-level, Resolution: 5-level. The parameter tuning algorithms are supposed to activate when CPU load is over 80%. The acceptable level of CPU load is set to below 60%.

Fig. 4 represents the experimental result using the rule-based programming language method applied to the conventional FVCS [4]-[7] by varying CPU load. From Fig. 5(a), when CPU load is increased by the CPU load generator on the side of WS-B at around 10 seconds (the first  $\Delta$ ) after starting videoconference, CPU load reaches to around 80% at around 60 seconds (point a) on the side of WS-A. At the point, the resolution, whose order of priority is the lowest, is first adjusted for degradation. It causes CPU load to decrease. From Fig. 5(b), quality at point b on the side of WS-B is adjusted for degradation. Also, frame rate, whose order of priority is the highest, is adjusted for small degradation. CPU load is decreased to some extent up to around 240 seconds (the second  $\Delta$ ). At 240 seconds, CPU load starts to increase again on the side of WS-B and reaches to over 80% at point c in Fig. 4(a) and at point d in Fig. 4(b). However, the parameters are not tuned anymore because of the characteristics of the conventional FVCS. From the result, it is revealed that the tuning method of the conventional FVCS is static so it is not able to flexibly cope with dynamical environment changes during one session, due to lack of the state knowledge stored in VCM. In addition, the protocols for cooperating between agents might not be appropriate in some cases.

Fig. 5 represents the experimental result using our proposed AVCS in which APTM is imbedded to VCM. The experimental condition is the same as that of the conventional FVCS. In this experiment, the parameters are tuned according to the priority of the parameters. As seen in Eq. (4), for WS-A, the initial value of the scaling factor for each parameter is set as  $\rho_{a_1} = 0.196$  for frame



**Fig. 5.** The behavior of the proposed AVCS according to the variation of CPU load:  
(a) Change of QoS at User-A, (b) Change of QoS at User-B

rate,  $\rho_{a_2} = 0.324$  for quality, and  $\rho_{a_3} = 0.525$  for resolution, respectively. For WS-B,  $\rho_{a_1} = 0.182$  for quality,  $\rho_{a_2} = 0.351$  for frame rate, and  $\rho_{a_3} = 0.463$  for resolution, respectively. The initial values were obtained by our exhaustive empirical experience. From Fig. 5(a), when CPU load is increased by the CPU load generator on the side of WS-B at around 10 seconds (the first  $\Delta$ ) after starting videoconference, CPU load reaches to around 80% at around 55 seconds (point e) on the side of WS-A. At the point, the parameters start to be tuned with the initial scaling factors by the automatic parameter tuning algorithm in APTM. However, the parameters start to be tuned at around 65 seconds (point f) on the side of WS-B, as seen in Fig. 5(b). CPU load is decreased to around up to 40% at around 240 seconds (the second  $\Delta$ ). In the meanwhile, the parameters are tuned to increase by automatically changing the direction of tuning from the negative to the positive, in order to efficiently use the 20% margin of CPU from the acceptable level (60%) of CPU. At 240 seconds, CPU load starts to increase again on the side of WS-B and reaches to over 80% at point g in Fig. 5(a) and at point h in Fig. 5(b). By the same process as the first parameter tuning step, the parameters start to be tuned and reach to the optimal points in which CPU load stays on an acceptable level.

## 6 Conclusion

In this paper, we have proposed a new AVCS by designing a new VCM in which the automatic parameter tuning module (APTM) is imbedded. From the experimental section, it can be concluded that the proposed AVCS is can more flexibly cope with the changes in working conditions, compared to the conventional FVCS. The AVCS adjusts the QoS parameters in a static fashion because it cannot repeat the process of the adjustment at the same session. However, the APTM can maintain a stable videoconferencing, by automatically tuning QoS parameters using the gradient method.

Despite encouraging results from our method, a couple of further studies are needed. In the experiment, we obtained the values of the scaling factors from exhaustive empirical experience. However, it does not guarantee that the values are the optimal initial scaling factors. In order to provide more automatic parameter tuning, an algorithm, which can decide the initial values of the scaling factors, is needed. Not only the proposed AVCS but also the conventional FVCS were developed for only two-user videoconference. We need to expand our method to multi-user videoconference because the multi-user videoconference is becoming more popular.

## References

1. Turletti, T., Huitema, C.: Videoconferencing on the Internet. IEEE/ACM Trans. on Networking, Vol.4, No.3 (1996) pp.340-351.
2. Bolliger, J., Gross, T.: A framework-based approach to the development of network-aware applications. IEEE Trans. on Software Engineering, Vol.24, No.5 (1998).
3. MaCanne, S., Jacobson, V.: Vic: a flexible framework for packet video. ACM Multimedia Nov. (1995) pp.511-522.
4. Suganuma, T., Kinoshita, T., Sugawara, K., Shiratori, N.: Flexible videoconference system based on ADIPS framework. Proc. of the 3rd International Conference and Exhibition on the Practical Application of Intelligent Agents and Multi-Agent Technology (pAAM98) (1998) pp.83-100.
5. Lee, S. D., Karahashi, T., Suganuma, T., Kinoshita, T., Shiratori, N.: Construction and evaluation of agent domain knowledge for flexible videoconference system. IEICEJ, Vol.J83-B (2000) pp.195-206.
6. Suganuma, T., Imai, S., Kinoshita, T., Shiratori, N.: A QoS control mechanism using knowledge-based multiagent framework. IEICE Trans. Information and Systems, Vol.E86-D, No.8 (2003) pp.1344-1355.
7. Lee, S. D., Han, D. S.: Multiagent based adaptive QoS control mechanism in flexible videoconference system. ICACT 2004, Vol.II, Feb. (2004) pp.745-750.
8. Kang, S. G., Lim, J. Y., Kim, M. C.: Modeling the user preference on broadcasting contents using Bayesian networks. Journal of Electronic Imaging, Vol.14, No.2 [Online]: 0230221-10 (2005).
9. Kinoshita, T., Sugawara, K.: ADIPS framework for flexible distributed systems. Springer-Verlag Lecture Notes in AI, 1599 (1998) pp.18-32.
10. Ousterhout, J. K.: Tcl and the Tk Toolkit. Addison-Wesley (1994).

# Kalman Filter Based Dead Reckoning Algorithm for Minimizing Network Traffic Between Mobile Game Users in Wireless GRID

Seong-Whan Kim and Ki-Hong Ko

Department of Computer Science, Univ. of Seoul, Jeon-Nong-Dong, Seoul, Korea  
Tel.: +82-2-2210-5316, fax: +82-2-2210-5275  
swkim7@uos.ac.kr, jedigo@venus.uos.ac.kr

**Abstract.** Whereas conventional GRID service is static, wireless GRID supports mobility, and it should maintain geographic position to support efficient resource sharing and routing. When the devices are highly mobile, there will be much traffic to exchange the geographic position information of each mobile node, and this makes adverse effect on efficient battery usage and network congestion. To minimize the network traffic between mobile users, we can use dead reckoning (DR) algorithm for each mobile nodes, where each node uses the algorithm to estimates its own movement (also other node's movement), and when the estimation error is over threshold, the node sends the UPDATE (including position, velocity, etc) packet to other devices. As the estimation accuracy is increased, each node can minimize the number of UPDATE packet transmission. To improve the prediction accuracy of DR algorithm, we propose Kalman filter based DR approach, and we also propose the adaptive Kalman gain control to minimize the number of UPDATE packet to distant device. To experiment our scheme, we implement a popular network game (BZFlag) with our scheme added on each mobile node, and the results show that we can achieve better prediction accuracy and reduction of network traffic by 12 percents.

**Keywords:** Dead reckoning, Kalman filter, Wireless GRID.

## 1 Introduction

Conventional GRID service supports no mobility, and results in many drawbacks such as continuous connection, waste of bandwidth, and service overloading. Wireless GRID supports mobility and it should consider geographic position to support efficient resource sharing and routing [1]. However, if the device in the GRID is highly mobile, there will be much traffic to manage the geographic position of each mobile node, and this make adverse effect on efficient battery usage. To minimize the network traffic between networking mobile devices, dead reckoning (DR) technique is used [2]. Each mobile device uses the algorithm to estimates its movement and other devices' movement, thereby, each device can minimize the transmission of its information (position, velocity, etc) to other entities. R. Gossweiler and R. J. Laferriere

introduced the DR algorithm for the multi-user game [2], and S. Aggarwal and H. Banavar proposed the use of globally synchronized clocks among the participating players and a time-stamp augmented DR vector that enables the receiver to render the entity accurately [3]. In addition, W. Cai and F. B.S. Lee proposed a multi-level threshold scheme that adaptively adjusted, based on the relative distance between entities to reduce the rate of transmitting UPDATE packets [4].

To improve the prediction accuracy of DR algorithm, we propose the Kalman filter based DR approach. To simulate the mobility of mobile device scenarios in wireless GRID, we use a simple analogy, network game (BZFlag). In section 2, we review the DR and Kalman filter. In Section 3, we propose a Kalman filter based DR algorithm. In Section 4, we apply our Kalman approach on BZFLAG game; show the experimental results with minimized UPDATE packets between game players. We conclude in section 5.

## 2 Related Works

In the wireless mobile GRID, overall GRID area is partitioned into several squares in GRID region. Each GRID region elects its own leader (gateway), and the leader performs location aware (exploit of geographic information). For Geographic location aware routing, we can use GEOCAST, GEOTORA, and GEOGRID [10].

- **GEOCAST:** GEOCAST sends a message to all mobile devices within a designated geographic area (so called a geo-cast region). It uses forwarding zone and multicast region. In the forwarding zone, the data packet is sent by unicast to each other's device, and in the multicast region, the data packet is sent by multicast to each other's device.
- **GEOTORA:** GEOTORA derives from TORA (temporally ordered routing algorithm). TORA maintains a DAG (directed acyclic graph) with the destination device as sink; the data packet is forwarded by the DAG's direction to sink. GEOTORA divides each region into TORA (DAG region) and GEOCAST region. In GEOCAST regions, mobile devices perform the flooding, and in the DAG region, mobile devices perform an anycast from the source to any host.
- **GEOGRID:** GEOGRID uses two methods such as the flooding-based geo-casting and ticket-based geo-casting. The flooding-based geo-casting allows any grid leader in the forwarding zone to rebroadcast the messages, and the ticket-based geo-casting allows only ticket-holding grid leaders to rebroadcast.

For these location-aware routing schemes, it is important to maintain precise location information on other devices. To maintain exact geographic information for all other device, it requires frequent location information exchange between devices, and it can easily make network congestion. To minimize the exact geographic information without heavy processing overhead is to use a kind of location prediction on each device.

When mobile devices are physically distributed, each mobile device should maintain other device's geographic state for efficient resource sharing and routing. As the number of mobiles increased, they should exchange their geographic information update request to each other, so that it can generate a large amount of communication and thus saturate network bandwidth. To reduce the number of UPDATE packets, the DR technique is used [4]. In addition to the high fidelity model that maintains the accurate position about its entities, each mobile device also has a DR model that estimates the position of all entities (both local and remote). Therefore, instead of transmitting state UPDATE packets, the estimated position of a remote mobile device can be readily available through a simple and localized computation [4]. Each mobile device M compares its real position with DR estimated position. If the difference between them is greater than a threshold, the M informs other mobile devices to update their geographic information of M [2]. Simple DR algorithm can be described as follows.

**Algorithm: Dead Reckoning**

```

for every received packet of other device do
    switch received packet type {
        case UPDATE
            fix position information of other device
            break;
    }

    [Extrapolation] Extrapolate all device (including
    me)'s geographic information based on the past state
    information;

    if (my true position - my extrapolated position) >
    Threshold {
        Broadcast an UPDATE packet to all the other device
    }
}

```

### 3 Kalman Filter Based Dead Reckoning Algorithm

In wireless GRID environment, each mobile device is geographically distributed. A technique referred to as DR is commonly used to exchange information about movement among the mobile devices [6, 7, 8]. Each mobile device sends information about its movement as well as the movement of the entities it controls to the other mobile devices using a DR update packets.. A update packets typically contains information about the current position of the entity in terms of x, y and z coordinates (at the time the update packet sent) as well as the trajectory of the entity in terms of the velocity component in each of the dimensions [3].

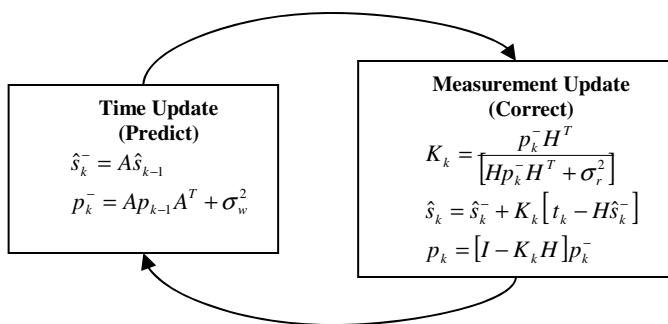
In this paper, we use the mobility of network game users to simulate the real geo-graphically distributed mobile device environments. For the network game, we

present a Kalman filter based DR to optimize the network traffic. A Kalman filter is a recursive procedure to estimate the states  $s_k$  of a discrete-time controlled process governed by the linear stochastic difference equation, from a set of measured observations  $t_k$ . The mathematical model is shown in Equation (1) and Equation (2).

$$s_k = As_{k-1} + w_{k-1} \quad (1)$$

$$t_k = Hs_k + r_k \quad (2)$$

The NxN matrix A represents an state transition matrix,  $w_k$  is an Nx1 process noise vector with  $N(0, \sigma_w^2)$ ,  $t_k$  is Mx1 measurement vector, H is MxM measurement matrix, and  $r_k$  is Mx1 measurement noise vector with  $N(0, \sigma_r^2)$ . To estimate the process, Kalman filter uses a form of feedback control as shown in Figure 1 [5]. We define  $\hat{s}_k^-$ ,  $\hat{s}_k$ ,  $p_k^-$  and  $p_k$  as the priori state estimate, posteriori state estimate, priori estimate error covariance, and posteriori estimate error covariance, respectively.  $K$  is the Kalman gain.



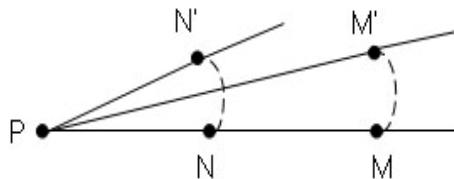
**Fig. 1.** Kalman filter cycle [5]

DR technique disseminates both location and movement models of mobile entities in the network so that every entity is able to predict or track the movement of every other entity with a very low overhead. The accuracy of location information affects the system performance; however, frequent transmission of DR UPDATE packets for increased accuracy can make network bandwidth overuse. To tradeoff the accuracy of location information and network bandwidth overuse, we can use the following consideration, as used in [11].

- As distance between devices increases, the accuracy requirement of geographic information gets decreased (“distance effect”).

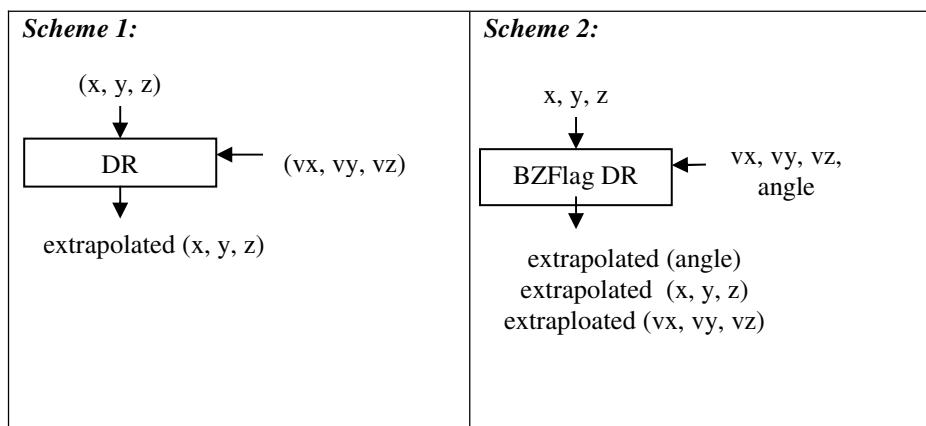
Figure 2 shows the distance effect. Entity N moves the same distance as entity M does. However, entity M is much far from P than N, so that entity P does not need to

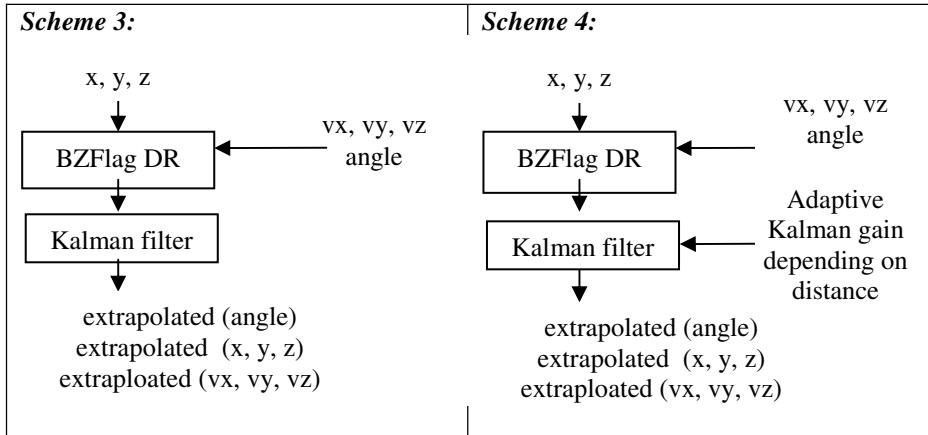
send UPDATE packet to M to maintain accurate geographic information in M. Layer concept is used to distinguish the accuracy of location information as distance [11]. Each layer applies different thresholds to classify the accuracy of location. However it requires processing overhead in each device because it should filter UPDATE packets considering distance between device P and destination device. Instead of using layer concept, we maintain the distance between the device P and every other device, and we set different Kalman gain (the allowable measurement noise) depending on the distance. For the device such as N and M as shown in Figure 2, we set higher observation error allowance on N. In other words, the measurement noise's variance for nearby entity N is higher than the measurement noise's variance for far away entity M. Kalman filter defines measurement noise's variance as Kalman gain, thereby we can set Kalman gain (measurement noise's variance) to achieve distance effects.



**Fig. 2.** Tradeoff between accuracy of geographic information and network traffic: (a) distance effect, and (b) density effect

To evaluate our scheme, we experimented with simple DR scenarios (scheme 1) and optimized DR algorithm for game logic (scheme 2). From scheme 2, we added Kalman filter for better prediction (scheme 3), and added adaptive Kalman filter gain control over scheme 3 (scheme 4). The overall schemes are shown in Figure 3.





**Fig. 3.** Kalman filter approach for DR algorithm

**Scheme 1:** We compute the extrapolated position using last position, last velocity, and time step as follows. We performed the extrapolation until the difference between the extrapolated position and the true position is under threshold.

```
Extrapolated position = last position + last velocity * time step;
```

**Scheme 2:** To get a new extrapolated position, the scheme uses two equations depending on the game entity's motion type as follows. We performed the extrapolation until the difference between the extrapolated position and the true position is under threshold.

```
if (linear motion) {
    extrapolated position = last position + last
    velocity * time step;
} else {
    extrapolated position = BZFlag function(angle);
}
```

**Scheme 3:** Scheme 3 adds Kalman filter after computing the extrapolated (position, velocity, and angle) as scheme 1. Our DR algorithm (scheme 3) is described as follows.

```
float speed = (vx * cosf(angle)) + (vy * sin(angle));
// speed relative to the tank's direction
radius = (speed / angular_velocity);

float inputTurnCenter[2]; // tank turn center
float inputTurnVector[2]; // tank turn vector
inputTurnVector[0] = +sin(last_angle) * radius;
inputTurnVector[1] = -cos(last_angle) * radius;
```

```

inputTurnCenter[0] = last_position-inputTurnVector[0];
inputTurnCenter[1] = last_position-inputTurnVector[1];

// compute new extrapolated angle using Kalman filter
float angle = Kalman (time step * angular_velocity);
float cos_val = cosf(angle);
float sin_val = sinf(angle);

// compute new extrapolated position
const float* tc = inputTurnCenter;
const float* tv = inputTurnVector;
new_x = tc[0]+((tv[0] * cos_val) - (tv[1] * sin_val));
new_y = tc[1]+((tv[1] * cos_val) + (tv[0] * sin_val));
new_z = last_position + (vz * time step);

// compute new extrapolated velocity
float vx = Kalman ((vx * cos_val) - (vy * sin_val));
float vy = Kalman ((vy * cos_val) + (vx * sin_val));
float vz = Kalman (vz);

```

**Scheme 4:** Scheme 4 considers distance effect to minimize the number of DR UPDATE packets. We achieve this minimization by controlling Kalman gain. Kalman gain depends on the allowable variance of measurement noise, and we set smaller Kalman gain as the distance between my device and another device gets longer. For example, the allowable variance is set to 0.001 for distant devices, 0.01 for middle distance, and 0.1 for nearby devices.

## 4 Experimental Results

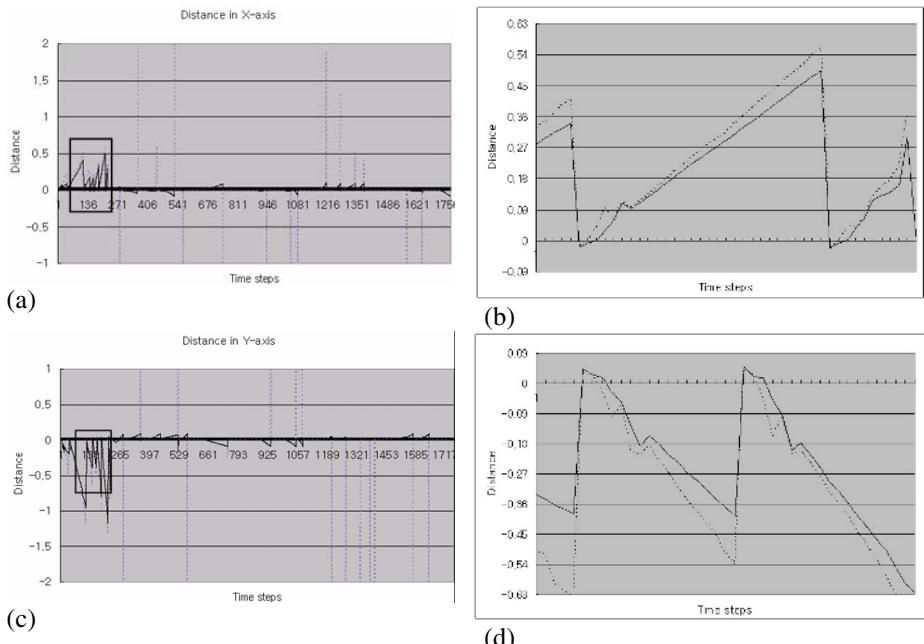
In this paper, we used a popular on-line game BZFlag (Battle Zone Flag) to experiment our scheme. BZFlag is a first-person shooting game, where each player in a team drives tanks within a battlefield. The aim of the players is to navigate and capture flags belonging to the other team and bring them back to their own area. The players shoot other player's tanks using "shooting bullets" The movements of the tanks (players) as well as that of the shots (entities) exchanged among the players using DR vectors [3, 9]. The position and the velocity value of each player are sampled from real BZFlag game plays. We used 8301 samples, and set the UPDATE threshold to 0.09. Noticeably, BZFlag game uses the game optimized DR algorithm, which means that it considers two additional vectors (orientation and angle) to predict the position more accurately. This game optimized DR works very well, and it performs much better than usual DR schemes. To show the Kalman filter approach over DR, we compared the number of DR UPDATE packet transmission and the average prediction error E as shown in Equation (3). (x, y, z) represent the true position, (newx, newy, newz) represent the extrapolated position, and (n) represent the number of sample.

$$E = \frac{\sum_{i=1}^{n=8301} \sqrt{(x_i - newx_i)^2 + (y_i - newy_i)^2 + (z_i - newz_i)^2}}{n} \quad (3)$$

Table 1 shows the experimental result. It shows that the number of DR UPDATE packet transmission of scheme 1 (simple DR scheme) can be decreased using game optimized DR scheme as in scheme 2. Incorporating Kalman filter (scheme 3) makes more accurate prediction, and it decreases the number of DR UPDATE packets. Finally, we can use distance effect to decrease the number of DR UPDATE packets to distant devices.

**Table 1.** Comparision of the # of DR UPDATE packet transmissions and prediction error E

	Scheme 1	Scheme 2	Scheme 3	Scheme 4
# of DR UPDATE packet transmission	4658	700	644	634
E	4.511	2.563	0.363614	0.363484



**Fig. 4.** Error in X prediction: (a) errors in X direction, (b) enlarged for box in (a), (c) Error in Y prediction, and (d) enlarged for box in (c)

Figure 4 compares the prediction errors in X and Y coordinates of scheme 2 and scheme 3. Scheme 2 shows fluctuations, and BZFlag clients should send DR UPDATE packets when the prediction error is over than threshold (we set 0.9). Minimizing DR packets also minimizes network latency and the game responses time. Figure 4 (a) and (c) show the overall prediction errors, and Figure 4 (b) and (d) show the detailed view on the prediction errors. Even in the detailed view, the prediction errors of scheme 4 are smaller than the prediction errors of scheme 3.

Scheme 3 does not consider distance effect. It assures the same prediction accuracy of geographic information for nearby and distant mobile devices. However, as a mobile device gets far away from my device, it is hardly observable, and it does not require the same accuracy requirement as for the nearby mobile device. Scheme 4 considers the distance effect by controlling Kalman gain. Kalman gain depends on the variance of measurement noise, and we can assume large allowance for variance on the distant devices. For distant devices, the number of DR UPDATE packets is minimized; however the prediction error can be increased. However, as shown in Table 2, scheme 4 can even decreases the prediction error for distant de devices.

**Table 2.** Comparison of scheme 3 and scheme 4 in # of DR packer transmission and errors

	# of DR UPDATE packet transmission			E		
	Near	Middle	Far	Near	Middle	Far
Scheme 3	11	615	18	0.025864	0.613829	0.020523
Scheme 4	11	607	16	0.025864	0.613677	0.020373

## 5 Conclusions

In this paper, we propose the Kalman filter approach to improve the DR algorithm for geographically oriented networking between mobile nodes in wireless GRID environments. Our scheme improves the accuracy of DR prediction, and minimizes the network traffic among the mobile devices, so that it can make efficient battery usage. Instead of experimenting geographically distributed mobile devices, we use a popular on-line game BZFlag, and compare our scheme with the state-of-the-art DR algorithm optimized for game logic. Our Kalman filter based DR scheme reduces more than 10% of network traffic over game optimized DR algorithms. More traffic reduction is achieved with adaptive Kalman gain control depending on the distance between my device and other device.

## References

1. Zhang W., Zhang J., Ma D., Wang B., Chen Y.: Key technique research on GRID mobile servie. Proc. 2<sup>nd</sup> Int. Conf. Information Technology (2004)

2. Gossweiler, R., Laferriere, R.J., Keller, M.L., Pausch, R.: An introductory tutorial for developing multi-user virtual environments. *Tele-operators and Virtual Environments*, vol. 3. no. 4 (1994) 255-264
3. Aggarwal, S., Banavar, H., Khandelwal, A., Mukherjee, S., Rangarajan, S.: User experience: accuracy in dead-reckoning based distributed multi-player games. Proc. ACM SIGCOMM 2004 Workshops on Net-Games. Network and System Support for Games (2004)
4. Cai, W., Lee, F.B.S., Chen, L.: An auto-adaptive dead reckoning algorithm for distributed interactive simulation. Proc. of the thirteenth Workshop on Parallel and Distributed Simulation (1999)
5. Welch, G., Bishop, G.: An introduction to the Kalman filters. available in <http://www.cs.unc.edu/~welch/Kalman/index.html>
6. Gautier, L., Diot, C.: Design and Evaluation of MiMaze, a Multiplayer Game on the Internet. Proc. IEEE Multimedia. ICMCS (1998)
7. Mauve, M.: Consistency in Replicated Continuous Interactive Media. Proc. of the ACM Conference on Computer Supported Cooperative Work (2000) 181–190
8. Singhal, S.K., Cheriton, D.R.: Exploiting Position History for Efficient Remote Rendering in Networked Virtual Reality. *Tele-operators and Virtual Environments*. vol. 4. no. 2 (1995) 169-193
9. Schoeneman, C., Riker, T.: BZFlag (Battle Zone capture Flag), available in <http://www.bzflag.org>
10. Tseng, Y.-C., Wu, S.-L., Liao, W.-H., Chao, C.-M.: Location awareness in ad hoc wireless mobile networks. *IEEE Computer*. vol. 34, no. 6, (2001) 46-52
11. Vijay K., Asmir R.D.: Performance of Dead Reckoning-Based Location Service for Mobile Ad Hoc Networks. *Wireless Communications and Mobile Computing Journal*. vol. 4, no. 2, (2004) 189-202

# Affective Web Service Design

Insu Song and Guido Governatori

School of Information Technology & Electrical Engineering  
The University of Queensland, Brisbane, QLD, 4072, Australia  
`{insu, guido}@itee.uq.edu.au`

**Abstract.** In this paper, we propose that, in order to improve customer satisfaction, we need to incorporate communication modes (e.g., speech act) in the current standards of web services specifications. We show that with the communication modes, we can estimate various affects on service consumers during their interactions with web services. With this information, a web-service management system can automatically prevent and compensate potential negative affects, and even take advantage of positive affects.

**Keywords:** E-commerce and AI Human computer interaction.

## 1 Introduction

In this paper, we discuss an important factor with regard to creating a successful Web services user experience that has been largely ignored. We propose that, in order to counter (detect, prevent, and resolve) negative affects that can be caused by Web services, we need to reconsider some of Web standards (e.g., WSDL) to include information about communication modes of interfaces (e.g., speech acts). Communication modes are important information for checking whether or not Web services are *well-behaved*. A Web service violating the well-behavedness can cause negative affects on its users. For instance, when a directive operation is invoked by a user (e.g., ‘add item A to my shopping cart’), it is a common knowledge that the user is expecting an acknowledgement within a certain time frame. A violation of this common knowledge can cause negative affects on the users.

Despite great deal of efforts on semantic web, this factor has been largely ignored. Thus, currently a large amount of development time is spent on making sure that Web services behave as intended. For example, in most e-commerce environments, the following problems frequently arise:

1. Customers often feel ignored or uncertain because prospective events, such as delivery notices, are not informed properly.
2. For customer services personals, it is hard to feel how customers are affected by the overall processes. Therefore, it is difficult to provide more adaptive and reasonable services and differences in customers’ situations are often ignored.

3. Promises are often not full-filled. E-commerce Web-sites often promise customers for certain behaviors of their services in their Web site, such as promotions, but actually certain behaviors do not meet such information.

We propose that Web services can be guaranteed to meet certain acceptable quality by incorporating affective computing, and thus avoiding these common problems.

Affective computing is not a new concept. In designing Web applications (e.g., e-commerce applications) that carry out a certain set of goals for human users, such as purchase orders, the importance of the affects of such applications has already been put forward to system designers' attention (e.g., [24,12]). Therefore, there has been much research into evaluating human emotions [11,6], expressing emotions [17,4], and the effectiveness of such approaches to improving human-computer interactions [16,5,3,25]. Business communities also have been aware of the significance of customer satisfaction in measuring business performances (e.g., American Consumer Satisfaction Index [1]) because for most companies, repeat customers are major contributors to profit [1]. Preventing negative affects on computer users is also one of the primary goals of HCI communities [20].

However, there are still no well defined languages to represent or account for affects of Web services on human users. As a result, current Web service design approaches do not have means for representing and estimating affects on users. Thus, it is impossible for Web service management systems to prevent possible negative affects or to take advantage of positive affects.

In this paper we define well-behaved protocols of Web-services based on speech act theory; and a method to evaluate various affects on the users when the Web-services violate such protocols based on cognitive appraisal theories of emotion. The rest of the paper is organized as follows. In the next section we discuss the issue of incorporating speech acts in Web service specifications. In Section 3, we define well-behavedness of Web service interfaces. In Section 4, we develop a method to evaluate affects on users during their interactions with Web services. Finally we conclude with comparisons with other approaches and some remarks.

## 2 Web Service Definition

Let us investigate the information that we can obtain from web service specifications in WSDL [7] which is a W3C standard for defining Web services. A WSDL document defines a set of interfaces through which consumers can interact with a Web service. Although WSDL standard defines four operation (interface) primitives that a service can support, in order to make the presentation more readable, in this paper we consider only two types of one-way operations: (One-way) the service receives a message, (Notification) the service sends a message. Since other primitive operation types defined in WSDL can be modelled abstractly using the two one-way messages, this is not a big limitation.

We represent a Web service  $W$  as a structure:

$$W = (OP_i, OP_o, M, DB, PL)$$

where  $OP_i$  is a set of input operation types,  $OP_o$  is a set of output operation types,  $M$  is a set of message types,  $DB$  is a database definition (e.g., DTD specifications for XML views of relational databases),  $PL$  is a set of plans (or procedures) each of which achieves certain goals.  $OP_i$ ,  $OP_o$ , and  $M$  can be obtained directly from WSDL specifications of Web services.

Let us represent each message type in  $M$  as follows:

$$\text{MessageType}(Part_1, \dots, Part_n)$$

which is just an alternative representation of WSDL message definitions. We also represent each operation type in  $OP_i$  and  $OP_o$  as  $OpName(m)$  where  $OpName$  is the operation name of the operation and  $m$  is a message type.

Unfortunately, WSDL specifications do not tell us much about the meanings of the operations. In particular, there is no way to tell what are the speech acts of the messages exchanged between Web services and service consumers. Without the speech act information of a message, it is impossible to know the intention of the message. Thus, we need to provide speech act information for each operations.

According to the speech act theory of Searle [26], each communication interaction is an attempt to perform an illocutionary act such as a request and an assertion. Therefore, interactions can be classified into five types (illocutionary points) of illocutionary acts, but we find that mostly only four types (directive, assertive, commissive, declarative) are used in user-service communications. We define an operation-to-speech-act mapping function as follows:

$$MSmap : O \rightarrow U \times F \times Q$$

where  $O$  is a set of operations,  $U$  is a set of users,  $F = \{\text{directive, assertive, declarative, commissive}\}$  is a set of illocutionary points,  $Q$  is a set of XPath queries. A query is a specification for checking the achievement of the operation.  $MSmap(op)$  returns a triple  $(u, f, q)$  for an operation instance  $op$ . We now suppose that this mapping function is given.

*Example 1.* Let us consider a shopping cart Web service example defined as follows:

$$\begin{array}{ll} OP_i = \{\text{AddItem(Item)}\} & OP_o = \{\text{Response(Item)}\} \\ M = \{\text{Item(name)}\} & PL = \{\text{AddItem(Item)} \rightarrow p1\} \end{array}$$

$PL$  has only one plan,  $p1$ , whose goal is  $\text{AddItem(Item)}$ . Suppose  $DB$  is an XML view of a relational database and its schema is defined by the following DTD specification:

```
root = basket
R={basket ← customerName, basketItem*;
   basketItem ← itemName;
   customerName ← #PCDATA;
   itemName ← #PCDATA;}
```

The root item ‘basket’ represents a shopping basket of a customer in an online shopping Web service. It can have zero or more items. Now, let’s suppose the

process receives the following request message from a user  $u$  through one-way operation AddItem: `AddItem(Item('Book'))`; and suppose the mapping function returns a triple  $(u, f, q)$  with the following values:

```
f= directive
q="/basket[customerName=u]/
basketItem[itemName='Book']/itemName/text()"
```

Then, the service upon receiving the message executes plan  $p_1$  for goal `AddItem(Item('Book'))`. The plan performs some actions that will eventually lead to an update to the database so that the XPath query  $q$  will return 'Book'.

An instance  $op$  of incoming operation can have one of the following goal types depending on the illocutionary point of the operation:

1. If  $f$  is a *directive*, the goal is  $op$  meaning that the service achieves  $op$ . For example, `AddItem(Item('Book'))`.
2. If  $f$  is an *assertive* or *declarative*, the goal is  $Bel(s, op)$  meaning that the service  $s$  believes  $op$ . For example, `Details(PhoneNo('5555'))`.
3. If  $f$  is a *commissive*, the goal is  $Bel(s, Int(u, op))$  meaning that the service  $s$  believes that the customer  $u$  intends to achieve  $op$  for the service.

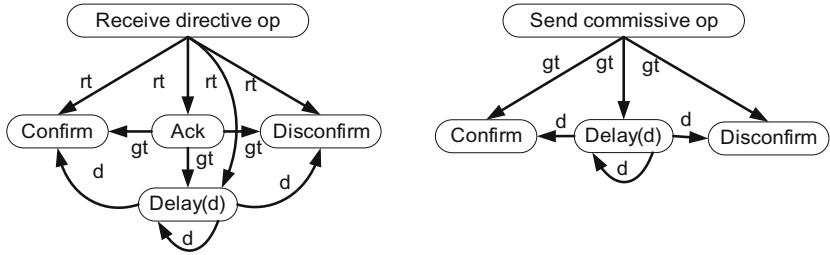
where  $Bel(s, p)$  means  $s$  believes  $p$ ,  $Int(u, p)$  means  $u$  intends  $p$ , and thus  $Bel(s, Int(u, p))$  means  $s$  believes that  $u$  intends  $p$ .

For outgoing messages  $op$ , if they are commissive, the goal is just  $op$  meaning that the service promises to achieve  $op$ . Other types of outgoing messages do not create goals. They are either assertive (informing messages) or directive and treated as actions produced by services.

### 3 Well-Behaved Web Service Interface

With the information of illocutionary point of each message, we can now define how a Web service should interact with its users. In this paper, we consider two cases: directive one-way operations and commissive notification operations. When a user of a Web service knows that the service has received a message containing the user's goal and the service is responsible for achieving it, the user expects the service to inform the user an acknowledgement or whether the goal is achieved or not. The acknowledgement means that (a) the message is received; and (b) the goal will be achieved within a certain time limit; and (c) if the goal is achieved, the user will be informed. If the goal is a promised goal, acknowledgement is not necessary. In both cases, if the goal cannot be achieved in a certain time limit, the process must send a delay message telling the user to wait for a certain time. These are just basic protocols that are expected by most human users. Similar protocols are also defined in an agent communication language called FIPA-ACL [14].

Let us call Web service interfaces conforming to the above descriptions *well-behaved-interfaces*. Figure 1 shows state flow diagrams for well-behaved-interfaces



**Fig. 1.** Well behaved web-service interface state flow diagrams for a directive one-way operation and a commissive notification operation. It is assumed that the response time,  $rt$ , is smaller (earlier) than the goal achievement time,  $gt$ :  $rt \leq gt$ .

for both a directive one-way operation and a commissive notification operation. In the figure, upon receiving a directive operation, the service must respond within a certain response time  $rt_g$  with one of the following messages:

1. a confirming message  $confim_g$  or
2. a disconfirming message  $disconfirm_g$  or
3. a delay message  $delay_g(d)$  or
4. an acknowledgement message  $ack_g$ .

If the response is either  $confim_g$  or  $disconfirm_g$ , the communication ends. But, if the response is  $delay_g(d)$  or  $ack_g$ , the user is expecting one of the above messages (except  $ack_g$ ) again within a certain goal achievement time,  $gt_g$ . If the following response is  $delay_g(d)$ , this time the user is expecting one of the above messages (except  $ack_g$ ) within  $d$ . The response behavior for a commissive notification operation is similar to a directive operation except that there is no acknowledgement as shown in Figure 1.

## 4 Detecting Affects on Users

Now, we consider how human users might be affected (or felt) if Web services are not well-behaved. There are many reasons that Web services cannot conform to the definitions of well-behaved-interface – the Internet is not always reliable and there are situations that Web-service designers have not anticipated or they are beyond the control of the services such as delay of back order items and natural disasters. We propose a simple and effective method to estimate affects on the users during their interactions with Web services. The method relies on prospective events which are main derivers of emotional states of human users according to cognitive appraisal theories of emotion (e.g., Ortony, Collins and Clore (OCC) [23] ).

### 4.1 Prospective Events of Web Services

We consider two classes of goals: *directive goals* created by directive one-way operations and *promised goals* created by commissive notification operations.

**Table 1.** Prospective events for a goal  $g$  and OCC classification of the events

Event Names	Symbols
goal failure time event	$gfe_g$
response failure time event	$rfe_g$
confirming event	$ce_g$
disconfirming event	$dce_g$
delay event	$dlye_g$
acknowledgement event	$acke_g$

OCC Types	Symbols
Prospective	all
Unexpected	-
Desirable	$ce_g$
Undesirable	$gfe_g, rfe_g, dce_g$
Unconfirmed	$dlye_g, acke_g$
Confirming	$ce_g$
Disconfirming	$dce_g$

**Table 2.** Interface based classification of events

Event Types	Symbols
User side time events: $TE_g$	$gfe_g, rfe_g$
Informing events: $IE_g$	$ce_g, dce_g, dlye_g, acke_g$

Given the two classes of goals, we can now enumerate the events that are relevant to these goals. But first, we make two important assumptions. When a user interacts with the service, it is reasonable to assume the followings:

1. Users know the meaning of each interface;
2. Users are aware of (or accustomed to) all prospective events related with the goals.

Based on these assumptions, we obtain the *user side time events* listed in Table 2. The service responsible for the goals must struggle to prevent these events occurring by producing *informing events* listed in the same table. Table 1 shows the descriptions of the event symbols. Table 1 also shows Ortony, Collins and Clore (OCC) [23] classification of these events. OCC have proposed a classification of events and their affects on (causes emotions) on communication participants.

As shown in Figure 1, for a directive goal  $g$ , there will be the time  $rt_g$  to inform of the acknowledgement of the acceptance of the goal to the user and the time  $gt_g$  to fulfill the goal before the user aware of undesirable events. However, when the user is not informed of the achievement of the goal within the goal achievement time  $gt_g$ , a goal failure time event  $gfe_g$  fires. When neither the achievement nor an acknowledgement is informed to the user within the response time  $rt_g$ , a response failure event  $rfe_g$  fires. As shown in Figure 1, for promised goals,  $rt_g$  is not necessary and only goal failure event  $gfe_g$  will occur.

Any user side time events  $\{gfe_g, rfe_g\}$  can cause negative emotions on the user. Thus, the process must create appropriate informing messages to prevent the user side time events occurring. According to Figure 1, there are four possible types of informing events: confirming events  $ce_g$ , disconfirming events  $dce_g$ , delay events  $dlye_g$ , and acknowledgement events  $acke_g$ . Each of these events occurs when the service sends the corresponding notification messages. The following production rules (whose conclusions are evaluated when their conditions are satisfied) summarizes the event firing policies for the user side time events:

$$\begin{aligned}
 r_1 : & \text{directive}_g \wedge (rt_g \leq t) \wedge \neg(acke_g \wedge dlye_g \wedge ce_g \wedge dce_g) \rightarrow rfe_g \\
 r_2 : & (gt_g \leq t) \wedge \neg(ce_g \wedge dce_g) \rightarrow gfe_g \\
 r_3 : & delay_g(d) \rightarrow (gt_g = gt_g + d)
 \end{aligned}$$

Rule  $r_1$  says that if the response time has passed and there have been no responses at all for a directive goal  $g$ , a response failure time event  $rfe_g$  occurs.  $r_2$  says that if the goal achievement time has passed and there have been neither a confirming message nor a disconfirming message, then a goal failure time event  $gfe_g$  occurs.  $r_3$  says that a delay message resets the goal achievement time.

If a failure event occurs, a new promising goal  $g'$  should be created in order to compensate the failure. The promising goal can only be formulated if we know the affect of the failure on the user.

## 4.2 Estimating Emotional States

This section describes how emotional states can be deduced from the prospective events of Web services based on the work of the Ortony, Collins and Clore (OCC) [23] cognitive appraisal theory of emotion which is one of the most widely accepted emotion models. The OCC model defines twenty-two emotion types, but we only describe six of them in this paper: hope, satisfied, fear, fears-confirmed, disappointment, and reproach. These emotions are prospective-based emotions that are emotions in response to expected and suspected states and in response to the confirmation or disconfirmation of such states [23].

Although the events we have described provide significant information to estimate users' emotional states, there can be always many other sources that can affect the users. Thus, we cannot use strict rules to capture relations between the events and emotional states. We use a fragment of Defeasible Logic (DL) [22] which is a popular nonmonotonic logic that is simple and computationally efficient. In DL, A defeasible rule  $L \Rightarrow c$  consists of its antecedent  $L$  which is a finite set of literals, an arrow, and its consequent  $c$  which is a literal. A literal is an atom  $p$  or its negation  $\neg p$ . A defeasible rule  $a_1, \dots, a_n \Rightarrow c$  can be expressed in the following logic program (without the monotonic kernel and the superiority relation of DL):

$$\begin{aligned}
 supported(c) &:- conclusion(a_1), \dots, conclusion(a_n). \\
 conclusion(c) &:- supported(c), \text{not } supported(\neg c), \text{not } strictly(\neg c).
 \end{aligned}$$

where  $\sim c$  is the complement of literal  $c$ ;  $conclusion(c)$  denotes that  $c$  is defeasibly provable;  $strictly(c)$  denotes that  $c$  is strictly provable. Then, the following defeasible rules roughly capture the relations between the events and emotional states:

- |     |  |
|-----|--|
| R0. | $\Rightarrow \neg rfe_g, \quad \Rightarrow \neg gfe_g$   |
| R1. | $ce_g \Rightarrow \neg hope_g, \quad dce_g \Rightarrow \neg hope_g, \quad gfe_g \Rightarrow \neg hope_g$ |
| R2. | $ce_g \Rightarrow \neg fear_g, \quad dce_g \Rightarrow \neg fear_g, \quad gfe_g \Rightarrow \neg fear_g$ |
| R3. | $directive_g, \neg rfe_g \Rightarrow hope_g$   |
| R4. | $directive_g, rfe_g \Rightarrow fear_g$  |
| R5. | $directive_g, \neg rfe_g, \neg gfe_g, ce_g \Rightarrow satisfied_g$                                      |

- R6.  $\text{commissive}_g, \neg gfe_g, ce_g \Rightarrow satisfied_g$
- R7.  $\text{directive}_g, rfe_g, gfe_g \Rightarrow fearconfirm_g$   
 $\text{directive}_g, rfe_g, dce_g \Rightarrow fearconfirm_g$
- R8.  $\text{directive}_g, \neg rfe_g, gfe_g \Rightarrow disappoint_g$   
 $\text{directive}_g, \neg rfe_g, dce_g \Rightarrow disappoint_g$
- R9.  $\text{commissive}_g, gfe_g \Rightarrow disappoint_g$   
 $\text{commissive}_g, dce_g \Rightarrow disappoint_g$
- R10.  $\text{directive}_g, rfe_g, ce_g \Rightarrow relieved_g$

Rules in R0 are assumptions that response failure events and goal failure events are not occurred. Rules in R1 and R2 say that if a communication is ended, a user usually feels neither hope nor fear. R3 says that a user usually feels hope over a desired goal  $g$  if no fear prospect ( $rfe_g$ ) is triggered. R4 says that a user usually feels fear if a desirable goal seem to be failing ( $rfe(g)$ ). Rules in R5 and R6 say that a user usually feels satisfied if a desirable goal (*directive* or *commissive*) is fulfilled without a trouble. Rules in R7 say that a user usually feels fear-confirmed if a desirable goal ( $\text{directive}(g)$ ) that seems to be failing is actually failed. Rules in R8 and R9 say that a user is usually disappointed if a desirable goal (*directive* or *commissive*) is failed. R10 says that a user is usually relieved if the user has had fear over a desirable goal (*directive*), but it is actually achieved.

The rules can be used to predict and estimate various affects that a Web service can cause on its users. With this information, a Web-service management system can prevent potential negative effects, compensate negative effects (e.g., sending an apology gift when a delivery is delayed), and even take advantage of positive affects (e.g., advertising when goods are successfully delivered without any troubles). This is a perfect technology that can tell when it is acceptable to send spam messages or to show pop-up advertisements.

## 5 Conclusion

Intelligent agent research communities have been working on various agent communication languages (e.g., KQML[13], FIPA-ACL[14]) [21,28] based on speech act theory. Speech act theory [27,8,9,10] has helped defining the types of messages based on the concept of illocutionary point, which constraints the semantics of the communication act itself [18, p.87]. It is also used as a basic ontology in organizational management systems [15] and in a conversation model for Web services [2].

However, Web service development communities have largely ignored the semantic issues of interactions. Currently, most works on Web service focus on design tools, infrastructure, and Web service composition. Thus, the standards developed for Web services (e.g., WDSL) mainly focus on the syntaxes of the description languages (e.g., WSDL), structural issues, or operational semantics (e.g., BPEL<sup>1</sup>) largely ignoring various service quality issues. Thus, there are

---

<sup>1</sup> <http://www.ibm.com/developerworks/webservices/library/ws-bpel/>

currently no standard ways to represent necessary data for quality management; it is difficult to compose Web services that meets the minimum requirement for *well-behaved Web services* for human users.

In contrast to usability evaluation methods (e.g., [19]), we only focus on the affects on the Web services users rather than efficiency oriented issues such as cognitive workload, performance, easy of use, and easy to learn. We should also note that our work does not require any of direct measurements of users such as brain activity, facial expression unlike existing approaches of affective computing. All information required is provided by the illocutionary points of operations and the emotion generation rules because our work only account for affects related to the goals that the services promise to deliver. However, actually these affects are the main issues that must be addressed.

This paper proposed that we need to incorporate communication modes in the current standards of Web-service specifications in order to improve customer satisfaction. We showed that with the communication modes, we can define well-behavedness of Web-service interfaces and estimate various affects on customers during their interactions with Web services. The result is important, since more businesses are relying on Web services as their primary contacts of customers.

## References

1. ACSI. American customer satisfaction index. <http://www.theacsi.org>, 2004.
2. L. Ardissono, A. Goy, and G. Petrone. Enabling conversations with Web services. In *Proc. AAMAS 2003*, pages 819–826, 2003.
3. Lesley Axelrod and Kate Hone. E-emotional advantage: performance and satisfaction gains with affective computing. In *CHI '05: CHI '05 extended abstracts on Human factors in computing systems*, pages 1192–1195, New York, NY, USA, 2005. ACM Press.
4. Alberto Battocchi, Fabio Pianesi, and Dina Goren-Bar. A first evaluation study of a database of kinetic facial expressions (dafex). In *Proc. ICMI '05: the 7th international conference on Multimodal interfaces*, pages 214–221, New York, NY, USA, 2005. ACM Press.
5. Timothy W. Bickmore and Rosalind W. Picard. Establishing and maintaining long-term human-computer relationships. *ACM Trans. Comput.-Hum. Interact.*, 12(2):293–327, 2005.
6. Carlos Busso, Zhigang Deng, Serdar Yildirim, Murtaza Bulut, Chul Min Lee, Abe Kazemzadeh, Sungbok Lee, Ulrich Neumann, and Shrikanth Narayanan. Analysis of emotion recognition using facial expressions, speech and multimodal information. In *Proc. ICMI '04: the 6th international conference on Multimodal interfaces*, pages 205–211, New York, NY, USA, 2004. ACM Press.
7. Erik Christensen, Francisco Curbera, Greg Meredith, and Sanjiva Weerawarana. Web services description language (WSDL). *W3C*: <http://www.w3.org/TR/wsdl>, 2001.
8. Philip R. Cohen and Hector J. Levesque. Performatives in a rationally based speech act theory. In *Meeting of the Association for Computational Linguistics*, pages 79–88, 1990.
9. Philip R. Cohen and Hector J. Levesque. Rational interaction as the basis for communication. In *Intentions in Communication*, pages 221–255. MIT Press, Cambridge, Massachusetts, 1990.

10. Philip R. Cohen and Hector J. Levesque. Communicative actions for artificial agents. In *Proc. ICMAS '95*, pages 65–72, San Francisco, CA, USA, 1995. The MIT Press: Cambridge, MA, USA.
11. Erica Costantini, Fabio Pianesi, and Michela Prete. Recognising emotions in human and synthetic faces: the role of the upper and lower parts of the face. In *Proc. IUI '05*, pages 20–27, New York, NY, USA, 2005. ACM Press.
12. F.N. Egger. Affective design of e-commerce user interfaces: How to maximise perceived trustworthiness. In *Proc. CAHD2001: Conference on Affective Human Factors Design*, pages 317–324, 2001.
13. Tim Finin, Richard Fritzson, Don McKay, and Robin McEntire. Kqml as an agent communication language. In *Proc. CIKM '94: the third international conference on Information and knowledge management*, pages 456–463, New York, NY, USA, 1994. ACM Press.
14. FIPA [Fondation for Intelligent Physical Agents]. FIPA ACL message structure specification. <http://www.FIPA.org>, 2002.
15. Fernando Flores, Michael Graves, Brad Hartfield, and Terry Winograd. Computer systems and the design of organizational interaction. *ACM Trans. Inf. Syst.*, 6(2):153–172, 1988.
16. Kiel M Gilleade and Alan Dix. Using frustration in the design of adaptive videogames. In *Proc. ACE '04: the 2004 ACM SIGCHI International Conference on Advances in computer entertainment technology*, pages 228–232, New York, NY, USA, 2004. ACM Press.
17. Lisa Gralewski, Neill Campbell, Barry Thomas, Colin Dalton, David Gibson, and University of Bristol. Statistical synthesis of facial expressions for the portrayal of emotion. In *Proc. GRAPHITE '04: the 2nd international conference on Computer graphics and interactive techniques in Australasia and South East Asia*, pages 190–198, New York, NY, USA, 2004. ACM Press.
18. Michael N. Huhns and Larry M. Stephens. Multiagent systems and societies of agents. In *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*, chapter 2, pages 79–120. The MIT Press, Cambridge, MA, USA, 1999.
19. Melody Y. Ivory and Marti A. Hearst. The state of the art in automating usability evaluation of user interfaces. *ACM Computing Surveys*, 33(4):470–516, December 2001.
20. Jonathan Klein, Youngme Moon, and Rosalind W. Picard. This computer responds to user frustration. In *Proceedings of ACM CHI 99*, volume 2, pages 242–243, 1999.
21. Yannis Labrou, Tim Finin, and Yun Peng. Agent communication languages: The current landscape. *IEEE Intelligent Systems*, 14(2):45–52, March/April 1999.
22. M. J. Maher and G. Governatori. A semantic decomposition of defeasible logics. In *AAAI '99*, pages 299–305, 1999.
23. A. Ortony, G. L. Clore, and A. Collins. *The Cognitive Structure of Emotions*. Cambridge University, Cambridge, 1988.
24. Rosalind W. Picard. *Affective Computing*. MIT Press, 1997.
25. Rosalind W. Picard. Affective computing: challenges. *International Journal of Human-Computer Studies*, 59(1/2):55–64, 2003.
26. John Searle. *Expression and Meaning*. Cambridge Univesity Press, Cambridge, England, 1979.
27. Munindar P. Singh. A semantics for speech acts. In *Readings in Agents*, pages 458–470. Morgan Kaufmann, San Francisco, CA, USA, 1997. (Reprinted from *Annals of Mathematics and Artificial Intelligence*, 1993).
28. Michael Wooldridge. *An Introduction to Multiagent Systems*. John Wiley Sons, 2001.

# An Empirical Study of Data Smoothing Methods for Memory-Based and Hybrid Collaborative Filtering

Dingyi Han, Gui-Rong Xue, and Yong Yu

Shanghai Jiao Tong University, No. 800, Dongchuan Road, Shanghai, 200240, China  
{handy, grxue, yyu}@sjtu.edu.cn

**Abstract.** Collaborative Filtering (CF) techniques are important in the e-business era as vital components of many recommender systems, for they facilitate the generation of high-quality recommendations by leveraging the similar preferences of community users. However, there is still a major problem preventing CF algorithms from achieving better effectiveness, the sparsity of training data. Lots of ratings in the training matrix are not collected. Few current CF methods try to do data smoothing before predicting the ratings of an active user. In this work, we have validated the effectiveness of data smoothing for memory-based and hybrid collaborative filtering algorithms. Our experiments show that all these algorithms achieve a higher accuracy after proper smoothing. The average mean absolute error improvements of the three CF algorithms, Item Based, k Nearest Neighbor and Personality Diagnosis, are 6.32%, 8.85% and 38.0% respectively. Moreover, we have compared different smoothing methods to show which works best for each of the algorithms.

## 1 Introduction

With the ability of predicting user preferences, recommending systems can help business companies to better analyze their product markets in today's e-business era. As the core techniques, collaborative filtering (CF) algorithms have proved their abilities of generating high-quality recommendations by leveraging the similar preferences of community users.

There are two major classes of prediction algorithms: model-based algorithms and memory-based algorithms. Model-based algorithms first learn a descriptive model of users, items and/or ratings, appealing to which they generate recommendations. Generally, they are memory economical but time-consuming to build and update the models. Algorithms within this category include Bayesian network approach [1], clustering approach [2,3], the aspect models[4], etc. On the other hand, memory-based algorithms just maintain a table of all user ratings for all items, across which they perform some computation for each prediction. Compared to model-based ones, they are usually memory-consuming but accurate. The notable examples include the Pearson-Correlation based approach [5], the vector similarity based approach [1] and the extended generalized vector-space model [6].

However, there is still a major problem preventing CF algorithms from achieving better effectiveness, the sparsity of training data. Lots of ratings in the training matrix are not collected. It is mainly caused by the cost consuming factors in time and effort of the data collection process. Take the popular data set Each-Movie, as an example. It is an explicit voting example set using data from the EachMovie collaborative filtering site<sup>1</sup> deployed by Digital Equipment Research Center from 1995 through 1997<sup>2</sup>. It includes over 2.1 million ratings (ranged in value from 0 to 5) of 1,623 movies by 61,265 users. Although the collected rating number is great, the density of the data set is still only 2.1%.

Few current CF methods try to smooth the sparse training matrix before predicting the ratings of an active user. Most of them analyze the training data directly to find similar user/item or to construct a model. Though a hybrid memory-and-model-based approach proposed by Pennock et al. in 2000 [7], named Personality Diagnosis (PD), does induct a simple smoothing method and attains a better results, it just assigns a uniform distribution over ratings to the blanks and does not exploit the information in the training data.

Therefore, to find some effective smoothing methods becomes meaningful to recommending systems as they may improve the prediction accuracy. Meanwhile, smoothing methods can also lower the cost of collecting data for training. Since they can fill in the blank elements in the rating matrix, the training data may not be needed as much as before. i.e. CF may work with less ratings.

In this work, we have validated the effectiveness of data smoothing methods for memory-based and hybrid collaborative filtering algorithms. In our experiments, the average mean absolute error improvements of the three CF algorithms, Item Based, k Nearest Neighbor and Personality Diagnosis, are 6.32%, 8.85% and 38.0% respectively. Moreover, we have compared different smoothing methods to show which works best for each of the algorithms.

The rest of the paper is organized as follows: We first discuss the related work in Section 2. Section 3 introduces some background knowledge, including our notation, problem definition and the three CF algorithms in brief. It is in Section 4 that the smoothing methods are described. The experimental results and analysis are given in Section 5. Section 6 is the conclusion part.

## 2 Related Work

There are currently three kinds of algorithms to deal with the sparsity problem. The first one is to do dimension-reduction. The second one is to acquire additional information. The last and latest one is to do data smoothing.

Dimension-reduction methods aims at reducing the dimensionality of the user-item matrix directly. Principle Component Analysis (PCA) [8] and information retrieval techniques such as Latent Semantic Indexing (LSI) [9,10] are used. Zeng proposed to compute the user similarities by a matrix conversion method [11]. These approaches address the sparsity problem by removing unrepresentative or

---

<sup>1</sup> <http://www.eachmovie.com/>.

<sup>2</sup> For more information, please visit <http://research.compaq.com/SRC/eachmovie/>.

insignificant users of items to condense the user-item matrix. However, some potentially useful information might also be removed during the reduction process.

Content-boosted CF [12,13] approaches require additional information regarding items as well as a metric to compute meaningful similarities among them. In [14], Popescul et al. also proposed a unified probabilistic model for integrating content information to solve the sparse-data problem. Most previous studies have demonstrated significant improvement in recommendation quality. However, such information may be difficult or expansive to acquire in practice.

Gui-Rong et al. have proposed a novel cluster-based smoothing method to solve the sparsity problem as well as the scalability problem [15]. They try to use smoothing methods to solve the sparsity problem and achieve a higher accuracy when using KNN algorithm to predict ratings. However, whether the method works on other CF algorithms are left unknown.

### 3 Background

#### 3.1 Smoothing Method Definition

Let  $U$  be the user set,  $n$  the user number,  $m$  the item number. Denote the  $n \times m$  matrix of all the ratings as  $\mathbf{R}$ . The rating of user  $i$  for item  $j$  is  $R_{ij}$ .  $\mathbf{R}_{i*}$  is user  $i$ 's rating vector.  $\mathbf{R}_{*j}$  is item  $j$ 's user rating vector. The value of  $R_{ij}$  maybe an provided integer  $R_{ij}^{\text{true}}$  or a blank (marked as  $\perp$ ).

The CF problem thus can be described as the follows. Given a rating matrix  $\mathbf{R}$  and some rating  $\mathbf{R}_{i*}^{\text{true}}$  of an active user  $i$ . CF is a function  $cf$  that can predict a blank rating  $R_{ij} = \perp$  of user  $i$ . i.e.  $R_{ij} = cf(\mathbf{R}, \mathbf{R}_{i*}^{\text{true}})$ . The closer to  $R_{ij}^{\text{true}}$  the predicted  $R_{ij}$  is, the better the CF algorithm will be.

A smoothing method  $s$  is to assign new values to blank value  $R_{ij}$  ( $= \perp$ ) in the training matrix  $\mathbf{R}$ . We use superscript ' $s$ ' to mark a smoothed rating.

$$R_{ij}^s = \begin{cases} R_{ij} & \text{if } R_{ij} \neq \perp, \\ s(\mathbf{R}^{\text{true}}, i, j) & \text{else.} \end{cases}$$

Thus, a smoothing method is a function to replace the  $\perp$ 's in a sparse matrix  $\mathbf{R}$  with new values  $s(\mathbf{R}^{\text{true}}, i, j)$ .

#### 3.2 Memory-Based and Hybrid CF Algorithms

Model-based CF algorithms are divided into two categories: user-based and item-based, which are different in the perspective of similarity measurement. The representative algorithms are respectively  $k$ -nearest neighbors (KNN) [11] and item-based CF (IB) [16]. KNN's basic idea is to compute the active user's predicted vote on an item as a weighted average of the votes given to that item by other users [11]. It first measures the similarity as the Pearson correlation of two rating vectors. The predictions are computed as the weighted average of deviation from the  $k$  nearest neighbors' mean. IB just makes predictions in the other dimension of the rating matrix. The best similarity method is adjusted

cosine similarity[16]. It predicts the active user rating by weighted average of the user's know rating.

Personality Diagnosis (PD) is a hybrid memory-and-model-based approach proposed by Pennock et al. [7] in 2000. It assumes that users report ratings for movies with Gaussian noise and the distribution of personality types or rating vectors in the matrix is representative of the distribution of personalities in the target population of users. By applying Bayes' rule, the probability that the active user is of the same personality type as any other can be computed. After computing this quantity for each user, the probability distribution for the active users rating of an unseen item can be calculated.

## 4 Smoothing Methods

There are many underlying scenarios of smoothing methods. One user may probably vote an item with similar value to most of the other users. Therefore, we may use the mean rating values of the item to replace the blanks. Some users may vote high rates to most movies and others may vote low rates according to their tastes. Hence, we may use the mean rating value of the user to replace those blanks. Besides, we may also cluster users/items into groups to get better smoothing results. Accordingly, smoothing methods can also be divided into groups from different perspective. From the dimensional perspective, they are divided into three groups: item-based, user-based and user-item-based smoothing. From the granular perspective, they are divided into matrix-based and cluster-based smoothing.

### 4.1 Dimensional Perspective

**Item-Based Smoothing.** Item-based smoothing method is a function like  $s(\{R_{ij_0}\}, i_0, j_0)$ . It replaces the blanks with other users' rating values to the same item. We may use the mean value of the other users' ratings.

$$s_I(\mathbf{R}^{\text{true}}, i_0, j_0) = s(\{R_{ij}|j = j_0\}, i_0, j_0) = \underset{i, R_{ij_0} \neq \perp}{\text{average}}(R_{ij_0}) \triangleq \overline{R_{ij_0}}$$

**User-Based Smoothing.** User-based smoothing method is a function like  $s(\{R_{i_0j}\}, i_0, j_0)$ . It replaces the blanks with the same user's rating value to the other items. We may use the mean value of the user's ratings to other items.

$$s_U(\mathbf{R}^{\text{true}}, i_0, j_0) = s(\{R_{ij}|i = i_0\}, i_0, j_0) = \underset{j, R_{i_0j} \neq \perp}{\text{average}}(R_{i_0j}) \triangleq \overline{R_{i_0j}}$$

**User-Item-Based Smoothing.** User-item-based smoothing method takes both of the above two scenarios into consideration. We may first calculate each user's average rating values to the movies, and then do item-based smoothing on the deviation rating matrix. Denote  $\Delta R_{i_0j_0} = R_{i_0j_0} - \overline{R_{i_0j}}$  as the deviation rating. The smoothing function can be described as:

$$s_{\text{UI}}(\mathbf{R}^{\text{true}}, i_0, j_0) = \overline{R_{i_0j}} + \underset{i, R_{ij_0} \neq \perp}{\text{average}}(\Delta R_{ij_0}) \triangleq \overline{R_{i_0j}} + \overline{\Delta R_{ij_0}}$$

## 4.2 Granular Perspective

**Matrix-Based Smoothing.** Matrix-based smoothing methods are large granular methods. They reference to all the rating values in the matrix just as  $s_I$ ,  $s_U$  and  $s_{\text{UI}}$  discussed before. We name them  $s_{\text{IM}}$ ,  $s_{\text{UM}}$  and  $s_{\text{UIM}}$ , respectively.

**Cluster-Based Smoothing.** Cluster-based smoothing methods hypothesize that user preferences can be implicitly clustered into groups and smoothing methods can get better effectiveness with similar users. So these methods first cluster users or items into groups by some clustering algorithm. Then, they perform smoothing methods in each cluster.

**IC method.** IC method first clusters users into groups. Thus the whole rating matrix is divided into several small matrixes vertically. For each small matrix, it does smoothing like what IM does to the whole matrix.

$$s_{\text{IC}}(\mathbf{R}^{\text{true}}, i_0, j_0) = \underset{i \in \mathcal{C}(i_0), R_{ij_0} \neq \perp}{\text{average}}(R_{ij_0})$$

Where  $\mathcal{C}(i_0)$  is the user cluster that  $i_0$  belongs to.

**UC method.** UC method first clusters items into groups. Thus the whole rating matrix is divided into several small matrixes horizontally. For each small matrix, it does smoothing like what UM does to the whole matrix.

$$s_{\text{UC}}(\mathbf{R}^{\text{true}}, i_0, j_0) = \underset{j \in \mathcal{C}(j_0), R_{i_0j} \neq \perp}{\text{average}}(R_{i_0j})$$

Where  $\mathcal{C}(j_0)$  is the item cluster that  $j_0$  belongs to.

**UIC method.** UIC method first clusters users into groups. For each user cluster, it does smoothing like what UIM does to the original matrix.

$$s_{\text{UIC}}(\mathbf{R}^{\text{true}}, i_0, j_0) = \underset{R_{i_0j} \neq \perp}{\text{average}}(R_{i_0j}) + \underset{i \in \mathcal{C}(i_0), R_{ij_0} \neq \perp}{\text{average}}(\Delta R_{ij_0}).$$

## 5 Experimental Results

### 5.1 Metrics and Datasets

Mean Absolute Error(MAE) [17] is used here to measure the prediction quality.

$$\text{MAE} = \frac{1}{|T|} \sum_{u \in T} |R_{uj} - \tilde{R}_{uj}|$$

where  $R_{ui}$  is the rating given to item  $i$  by user  $u$ ,  $\tilde{R}_{ui}$  is the predicted value of user  $u$  on item  $i$ ,  $T$  is the test set, and  $|T|$  is the size of the test set.

**Table 1.** Training and Testing Dataset Characteristics

Dataset Name	Data Souce	Rate Number (All)	Movie Number	Matrix Density	Rates Level
EM-Train	EachMovie	293,081	1,628	4.50%	0,1,2,3,4,5
ML-Train	MovieLens	670,346	3,900	4.30%	1,2,3,4,5
EM-Test	EachMovie	107,379	1,628	4.40%	0,1,2,3,4,5
ML-Test	MovieLens	253,918	3,900	4.34%	1,2,3,4,5

To test the matrix filling methods with different data distributions, we have taken EachMovie and MovieLens<sup>3</sup> as two sources of our test-bed. For each data source, we have extracted two subsets with no intersection and a limitation of more than 20 rates per user. One is for training, containing 4,000 users. The other is for testing, containing 1,500 users. Some characteristics of the data sets are listed in Table 1. To get the different density training datasets, we randomly remove some ratings. By predicting each rate as the most appearing rate value 3, we can also get an MAE lower bound of each test set. They are 1.20 on EM-Test and 1.07 on ML-Train. If we got an MAE larger than the lower bound, we may regard the prediction as a failure.

## 5.2 How Does Smoothing Work?

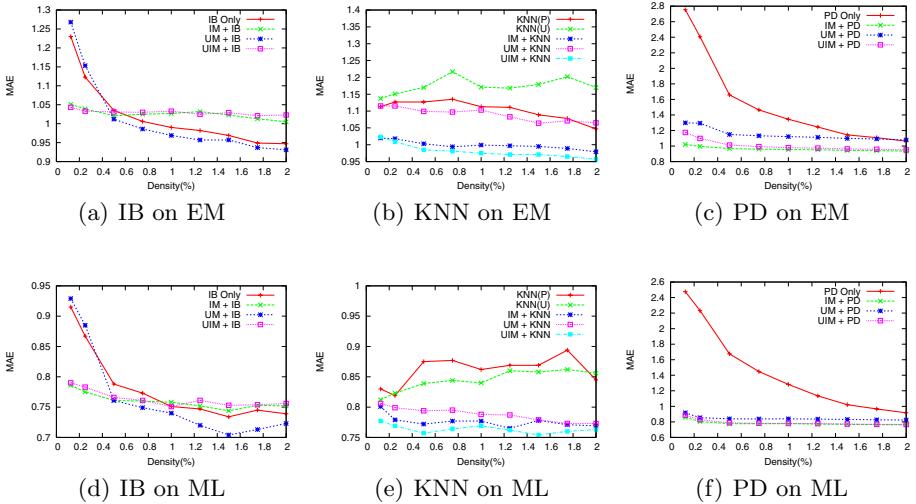
To test how smoothing can work, we performed IB, KNN without penalty (KNN(U)), KNN with penalty (KNN(P))<sup>4</sup>, PD algorithms without smoothing and with UM, IM, UIM smoothing methods on the two datasets. MAE values are given in Figure 1. Typically, MAE decreases as the training dataset density grows except those of KNN's. From the figure, we get the following observations:

1. IB algorithm achieves a higher accuracy when performed on less-than 0.4% density training dataset using IM/UIM or larger-than 0.4% density training dataset using UM. IM and UIM decrease the accuracy of IB algorithm when the training dataset density is larger than 1%.
2. To both KNN(U) and KNN(P), All the three methods can help to lower MAE. Only in some cases on EachMovie data, UM fails.
3. All the three methods can improve PD algorithm's accuracy. The sparser the training data is, the higher the improvement will be.

These observations validate that with proper smoothing methods, memory-based or hybrid CF algorithms can achieve a higher accuracy. Item based CF should use UM smoothing while KNN and PD should use IM or UIM smoothing because such methods can provide some useful information from the other

<sup>3</sup> 1 Million MovieLens Dataset, <http://www.cs.umn.edu/research/GroupLens/>.

<sup>4</sup> After smoothing, KNN algorithm gets the same MAE with and without penalty. We set  $k = 10$  in the experiments.



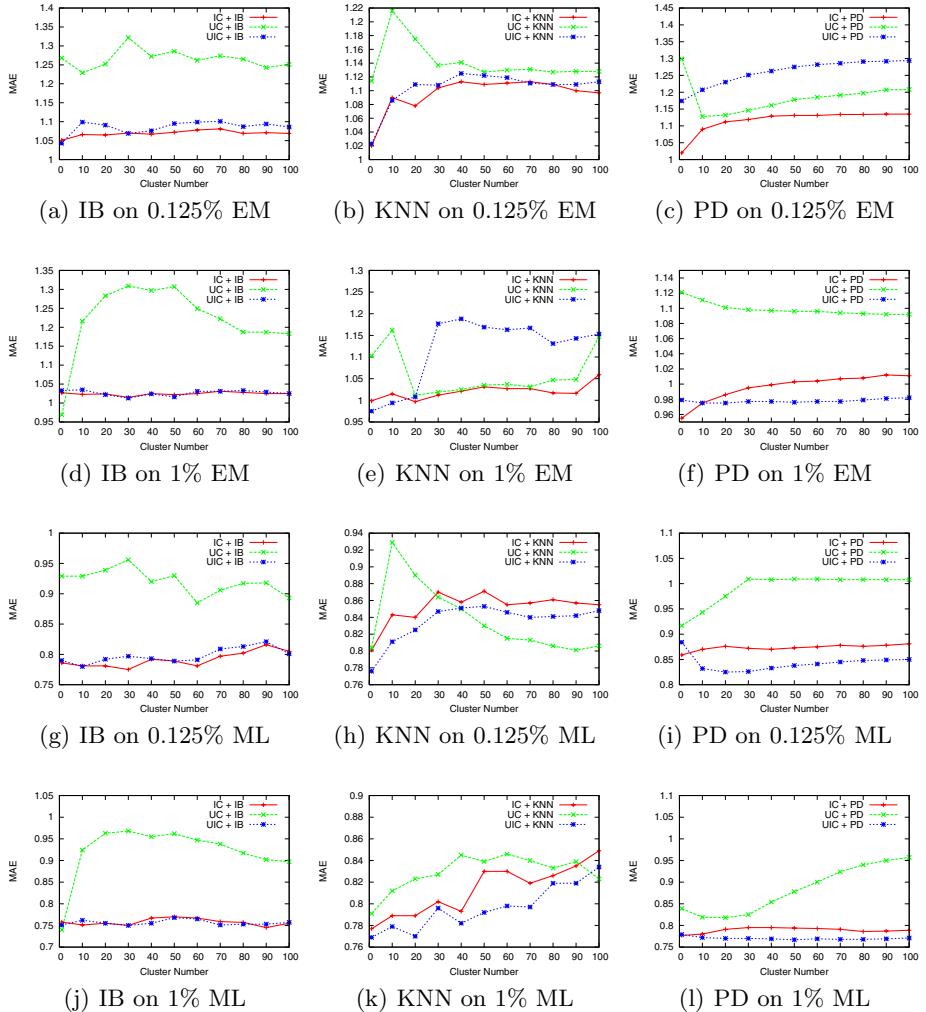
**Fig. 1.** MAEs of IB, KNN, and PD on different sparsity EachMovie and MovieLens datasets with different smoothing methods

perspective of the training matrix than the perspective those methods do predictions from. If we do smoothing from the same perspective of CF algorithms do predictions, the MAEs will be lower on sparse dataset only. They will be higher on dense dataset as these smoothing methods data can weaken the abilities of CF algorithms to identifying unique user preferences. The experimental results of “IM + IB” and “UM + KNN” also show evidences.

### 5.3 Is There Any Difference with Cluster Numbers?

We have also clustered the 0.125%, 1% and 2% density datasets of both EachMovie and MovieLens training data into different number of clusters to see where cluster can help smoothing and CF algorithms. The clustering is performed by the Repeated Bisecting method [18] of cluto - a software package for clustering high-dimensional datasets, using  $\mathcal{I}_2$  criteria [19]. The MAEs are shown in Figure 2. We do not present the figures on the 2% density datasets due to the space limit. They are quite similar to those on the 1% density datasets. In the figures, when cluster number is 1, IC, UC and UIC methods are just the same as IM, UM and UIM. The observations are:

1. Most of the minimum MAEs are obtained by the smoothing methods without clustering. There is no clear advantage of cluster-based smoothing methods.
2. The effects of the cluster-based algorithms are similar on the two datasets.



**Fig. 2.** MAEs of IB, KNN, and PD on different density datasets with different cluster number and different smoothing methods

#### 5.4 A Brief Summary

When CF algorithms are performed in with quite sparse training set, proper smoothing methods can help them to predict user preferences more accurately. Here we list the best smoothing methods for each of the three CF algorithms on the 0.125%, 1% and 2% density datasets in Table 2.

**Table 2.** Best Smoothing(Smt.) Methods on Two Datasets

EachMovie Density	2%			1%			0.125%		
Algorithm	IB	KNN	PD	IB	KNN	PD	IB	KNN	PD
Best Smt. Method	UM	UIM	IM	UIM	UIM	IM	UIM	IM	IM
MAE without Smt.	0.947	1.047	1.063	0.990	1.113	1.345	1.230	1.113	2.754
MAE with Smt.	0.931	0.956	0.935	0.969	0.975	0.955	1.043	1.020	1.020
Improvement	1.69%	8.69%	12.0%	2.12%	12.4%	29.0%	15.2%	8.36%	62.9%
MovieLens Density	2%			1%			0.125%		
Algorithm	IB	KNN	PD	IB	KNN	PD	IB	KNN	PD
Best Smt. Method	UM	UIM	UIC(70)	UM	UIM	UIC(70)	IC(30)	IM	UIC(20)
MAE without Smt.	0.739	0.845	0.917	0.751	0.862	1.282	0.915	0.830	2.476
MAE with Smt.	0.723	0.766	0.759	0.740	0.769	0.768	0.775	0.801	0.825
Improvement	2.17%	9.35%	17.2%	1.46%	10.8%	40.1%	15.3%	3.49%	66.7%

## 6 Conclusion

In this paper, we have compared several smooth methods to deal with the sparsity problem in recommendation systems. When properly smoothing methods are performed, the representative memory-based algorithms (IB, KNN) and the hybrid algorithm (PD) can be improved to a higher accuracy. In our experiments, the average MAE improvements of IB, KNN and PD algorithm are 6.32%, 8.85% and 38.0% respectively. These improvements will lead current recommendation systems to higher accuracies with lower training data requirements.

Our conclusion is that, to deal with the sparse training data problem in collaborative filtering, IB algorithm should use UM smoothing method before predicting. KNN should use UIM smoothing method before predicting. PD algorithm should use IM smoothing method before predicting. Although clustering may sometimes help to improve the prediction accuracy, they do not show a clear advantage. Future work may include a theoretical analysis of the effects of these smoothing methods to the CF algorithms.

## References

1. Breese, J.S., Heckerman, D., Kadie, C.: Empirical analysis of predictive algorithms for collaborative filtering. In: Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence. (1998) 43–52
2. Kohrs, A., Merialdo, B.: Clustering for clooaborative filtering applications, IOS Press (1999)
3. L.H. Ungar, D.P.F.: Clustering methods for collaborative filtering. In: Proceedings of the Workshop on Recommendation Systems, AAAI Press (1998)
4. Hofmann, T.: Latent semantic models for collaborative filtering. ACM Transactions on Information System **22** (2004) 89–115
5. Konstan, J.A., Miller, B.N., Maltz, D., Herlocker, J.L., Gordon, L.R., Riedl, J.: GroupLens: applying collaborative filtering to usenet news. Communications of the ACM **40** (1997) 77–87

6. Soboroff, I., Nicholas, C.: Collaborative filtering and the generalized vector space model (poster session). In: Proceedings of the 23rd annual international conference on Research and development in information retrieval. (2000) 351–353
7. Penmnock, D.M., Horvitz, E., Lawrence, S., Giles, C.L.: Collaborative filtering by personality diagnosis: A hybrid memory-and-model-based approach. In: Proc. of the 16th Conference on Uncertainty in Artificial Intelligence. (2000) 473–480
8. Goldberg, K.Y., Roeder, T., Gupta, D., Perkins, C.: Eigentaste: A constant time collaborative filtering algorithm. *Information Retrieval* **4** (2001) 133–151
9. Fisher, D., Hildrum, K., Hong, J., Newman, M., Thomas, M., Vuduc, R.: Swami: a framework for collaborative filtering algorithm development and evaluation. In: Proceedings of the 23rd annual international conference on Research and development in information retrieval. (2000) 366–368
10. Sarwar, B.M., Karypis, G., Konstan, J.A., Riedl, J.T.: Application of dimensionality reduction in recommender system – a case study. In: ACM WebKDD 2000 Web Mining for E-Commerce Workshop. (2000)
11. Zeng, C., Xing, C.X., Zhou, L.Z.: Similarity measure and instance selection for collaborative filtering. In: Proceedings of the 12th international conference on World Wide Web. (2003) 652–658
12. Balabanovic, M., Shoham, Y.: Fab: content-based, collaborative recommendation. *Communication of the ACM* **40** (1997) 66–72
13. Claypool, M., Gokhale, A., Mirands, T., Murnikov, P., Netes, D., Sartin, M.: Combining content-based and collaborative filters in an online newspaper. In: ACM SIGIR Workshop on Recommender Systems - Implementation and Evaluation. (1999)
14. Popescul, A., Ungar, L.H., Pennock, D.M., Lawrence, S.: Probabilistic models for unified collaborative and content-based recommendation in sparse-data environments. In: Proceedings of the 17th Conference on Uncertainty in Artificial Intelligence. (2001) 437–444
15. Xue, G.R., Lin, C., Yang, Q., Xi, W., Zeng, H.J., Yu, Y., Chen, Z.: Scalable collaborative filtering using cluster-based smoothing. In: Proceedings of the 28th annual international conference on Research and development in information retrieval. (2005) 114–121
16. Sarwar, B., Karypis, G., Konstan, J., Reidl, J.: Item-based collaborative filtering recommendation algorithms. In: Proceedings of the 10th international conference on World Wide Web. (2001) 285–295
17. Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating collaborative filtering recommender systems. *Trans. on Information System* **22** (2004) 5–53
18. Zhao, Y., Karypis, G., Fayyad, U.: Hierarchical clustering algorithms for document datasets. *Data Mining and Knowledge Discovery* **10** (2005) 141 – 168
19. Zhao, Y., Karypis, G.: Soft clustering criterion functions for partitional document clustering: a summary of results. In: Proceedings of the thirteenth ACM conference on Information and knowledge management. (2004) 246–247

# Eliminate Redundancy in Parallel Search: A Multi-agent Coordination Approach

Jiewen Luo<sup>1,2</sup> and Zhongzhi Shi<sup>1</sup>

<sup>1</sup> Institute of Computing Technology, Chinese Academy of Sciences  
100080 Beijing, China  
`{luojw, shizz}@ics.ict.ac.cn`

<sup>2</sup> Graduate University of Chinese Academy of Sciences  
100080 Beijing, China

**Abstract.** Web spider is a widely used approach to obtain information for search engines. As the size of the Web grows, it becomes a natural choice to parallelize the spider's crawling process. However, parallel execution often causes redundant web pages to occupy vast storing space. How to solve this problem becomes a significant issue for the design of next generation web spiders. In this paper, we employ the method from multi-agent coordination to design a parallel spider model and implement it on the multi-agent platform MAGE. Through the control of central facilitator agent, spiders can coordinate each other to avoid redundant pages in the web page search process. Experiment results demonstrate that it is very effective to improve the collection efficiency and can eliminate redundant pages with a tiny efficiency cost.

## 1 Introduction

One of the important components in a search engine is the web spider, a program downloading and storing web pages. It can traverse the World Wide Web information space by following hypertext links and retrieve web documents by standard HTTP protocol. Generally, a spider starts off by placing an initial URL, namely seed URL, in a queue. From this queue, the spider gets a URL, downloads web pages, extracts any URLs in the downloaded pages, and then puts the new URLs in the queue. This process is repeated until the spider decides to stop. Collected pages are mainly used to create a copy of all the visited websites for later processing by a search engine that will index the downloaded pages to provide fast searches.

As the size of the Web grows, it becomes more difficult to retrieve the whole or a significant portion of the Web using a single spider. Therefore, it is a natural choice for many search engines to parallelize the collection process for maximizing the download rate. We refer to this type of spider as a parallel spider. Although this approach can considerably improve the efficiency, it also takes great challenges in how to eliminate the web page redundancy caused by parallelization on the one hand and minimize the cost on the other.

In order to investigate this problem, we design the parallel spider model based on a multi-agent system. A multi-agent system is one in which a number of agents

cooperate and interact with each other to achieve a global objective in a complex and distributed environment. With the cooperation, spider agents can coordinate the information collection actions, which effectively avoid the page redundancy. To test the approach, we realize the model on the multi-agent platform MAGE[3] and conduct a series of experiments. Experiment results demonstrate that it is very effective to solving page redundancy problem with a tiny multi-agent interaction cost.

## 2 Related Work

Web spiders have been studied since the advent of the Web [4-10]. As to parallel spiders, there exists a significant body of literature studying the general problem of parallel and distributed computing [12,13]. Some of these studies focus on the design of efficient parallel algorithms. For example, references [14,15] present various architectures for parallel computing, propose algorithms solving various problems under the architecture, and study the complexity of the proposed algorithms. Besides, some researchers explore to build cooperative information gathering system from distributed problem solving perspective [18]. Another body of literature investigates the parallel spider from multi-agent perspective [1,17,18]. Reference [1] presents the Collaborative Spider, a multi-agent system designed to provide post-retrieval analysis and enable across-user collaboration in web search and mining. This system allows the user to annotate search sessions and share them with other users. References [17,18] introduce the infoSleuth project, which is based on a multi-agent system for heterogeneous information resources in open and dynamic environments.

However, all the parallel models and multi-agent system referred above do not amply consider or present effective approaches for the redundant pages issue. When multiple spiders run in parallel to download pages, it is possible that different processes download the same pages multiple times. One spider may not be aware that another spider has already downloaded the pages. Can we coordinate the parallel process to prevent redundant pages? This paper will investigate this issue in the following sections.

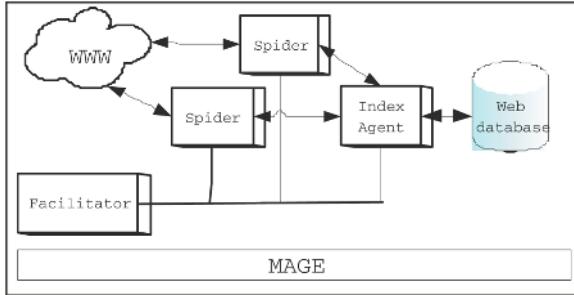
## 3 Parallel Web Spider Model

### 3.1 The Architecture of Parallel Model

Based on the framework of MAGE<sup>1</sup>, we design a parallel spider model for web information collection. Figure 1 shows the model's architecture. It contains three kinds of agents: managing agent (Facilitator), spider agent and index agent. They all execute on the MAGE platform. In our system, MAGE provides the multi-agent runtime environment support. Moreover, MAGE platform has implemented most basic classes, which can be inherited or extended by new classes

---

<sup>1</sup> <http://www.intsci.ac.cn/en/research/mage.html>



**Fig. 1.** Generic Agent Structure in MAGE Platform

conveniently, e.g. Agent class, ACLMessage class. Thus, it makes the model implementation process easier since we need not to consider the complex multi-agent mechanism and can focus our work on the web spider's search function.

Facilitator is the management center in the parallel spider system. It is a special agent that is used to manage the seed URL resource and interaction process among agents. If an agent wants to participate in the system, it should register on the facilitator to get permit message. Moreover, the facilitator controls the life cycle of other agents. For instance, facilitator can kill all the registered agents in the runtime.

Spider is the main part of the system. It starts work after receiving permit message from the facilitator. And then, it begins to search and download web pages from the seed URL and parse their children URL's hyperlinks for useful information. In our parallel model, we adopt the FIPA ACL<sup>2</sup>as the standard communication language among agents. Through coordination of the facilitator, spider agents can effectively avoid redundant pages. After finishing page collection, spiders inform the index Agent .The main job of the index agent is classifying, indexing and storing the web pages, which form the resource library for search engine and future web analysis.

### 3.2 Dynamic Assignment Mechanism

In this section, we explain how to avoid redundant pages caused by parallelization with the dynamic assignment mechanism. When multiple spiders download pages in parallel, different spiders may download the same pages multiple times. The overlap situation is presented by figure 2.

In order to solve this problem, we implement the dynamic assignment mechanism through multi-agent coordination. The central facilitator logically divides the WWW into different domains (e.g. *sina.com*<sup>3</sup> and *yahoo.com* etc) and dynamically assigns each domain to a spider as the seed URL. After MAGE generates a new spider and arbitrarily assigns it a seed URL, the new spider first checks the facilitator's URL domain to check whether the seed URL has been

<sup>2</sup> <http://www.fipa.org/specs/fipa00061/>

<sup>3</sup> <http://www.sina.com/> is the biggest Chinese website.

---

**Algorithm 1.** Facilitator Dynamic Assignment

---

```

1: Facilitator(); // facilitator initialization
2: Message=Facilitator.WaitForMessage();
3: if Message.Type = Registration then
4:   Record(SpiderName,IP,seedURL);
5:   Put(seedURL,DomainLibrary); // put the new seedURL into Domain Library
6: else if Message.Type = checkSeedURL then
7:   URL=Message.getSeedURL(); // get seedURL from Message.
8:   Flag=Check(URL); // check if the URL is existent in domain library
9: end if
10: if Flag = True then
11:   sendMessage(Spider,Refuse,The seed URL is existent);
12: else if Flag = False then
13:   sendMessage(Spider,Allow,The seed URL is vacant);
14:   wakeup(Spider);
15: end if

```

---



---

**Algorithm 2.** Spider Dynamic Assignment

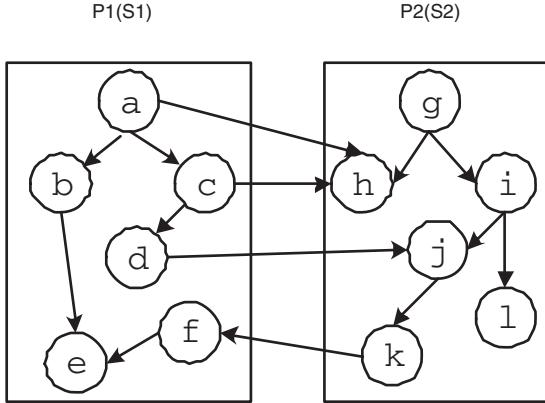
---

```

1: m=maxOfVisiteSite; // assign the max of visit site to m
2: n=maxOfSearchDepth; // assign the max search depth to n
3: Spider(SpiderName,IP); // spider agent initialization
4: seedURL=Spider.getSeedURL();
5: sendMessage(Facilitator,checkSeedURL);
6: Message=Spider.WaitForMessage();
7: if Message.Type = Refuse then
8:   Inform(Message.content);
9: else if Message.Type = Allow then
10:  sendMessage(Facilitator,Registration); // get seedURL from Message.
11: end if
12: loop
13: if numOfSite <= m and searchDepth <= n then
14:   URL=Search(seedURL); // search begin with seedURL
15:   if URL is not consistent with seedURL then
16:     sendMessage(Facilitator,checkSeedURL);
17:     Message=Spider.WaitForMessage();
18:     if Message.Type = Refuse then
19:       seedURL=findNode(URL); //call back and find brother node
20:     else if Message.Type = Allow then
21:       new Spider(SpiderName,IP)
22:       sendMessage(Facilitator,Registration);
23:     end if
24:   end if
25: end if
26: end loop

```

---



**Fig. 2.** Site  $P_1$  is crawled by  $S_1$  and site  $P_2$  is crawled by  $S_2$

assigned to other spiders. If not assigned, the new spider adds the seed URL and successfully registers to the facilitator. Otherwise, the facilitator shows warning message "*The seed URL is existent*" and suggests it to begin with other URLs. The algorithm 1 and 2 present the concrete process of the dynamic assignment.

## 4 Implementation and Experiments

### 4.1 Implementation

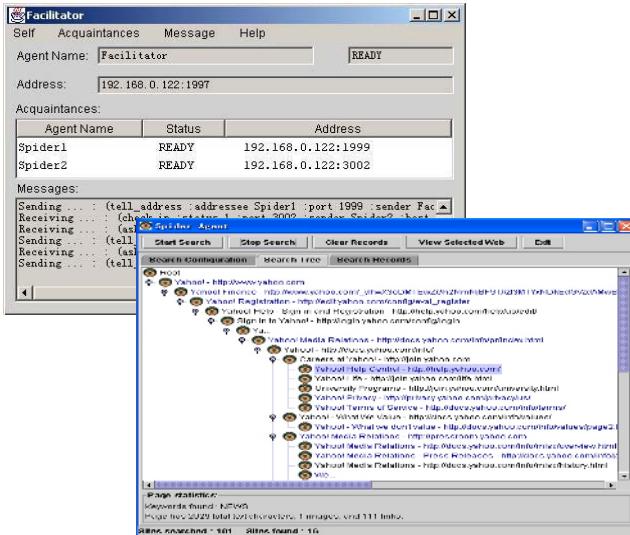
We have implemented the parallel spider system with dynamic assignment function based on MAGE with pure Java language. Every spider inherits *Agent Class* in MAGE. Therefore, they have the basic functions such as communication using FIPA ACL message and life cycle management. Figure 3 illustrates the screenshot of the facilitator and spider agent.

In order to control display, a specific Spider thread runs to conduct the Web search. Separate threads are used so that the main GUI frame can continually update the search tree and process the stop search request. As the Spider runs, it continually adds hyperlink nodes to the display tree. After search finishing, users can view the site's vital statistics, including the keywords present, total text characters, total images, and total links, etc.

### 4.2 Experiments

In this section, we design several experiments to investigate the parallel spider's performance. The experiments are conducted on the same computer hardware environment: 256MB DDR memory, Intel Pentium II processor and Windows XP operating system. At first, we initiate three spiders and assign them seed URLs [www.sina.com](http://www.sina.com), [www.sohu.com](http://www.sohu.com)<sup>4</sup> and [www.yahoo.com](http://www.yahoo.com) respectively. And then, we

<sup>4</sup> <http://www.sohu.com/> is the second biggest Chinese website.



**Fig. 3.** Screenshot of facilitator and spider agent

**Table 1.** Redundant pages and Redundancy Rate when three spiders run independently. (*Time interval =30 seconds*)

Seed URL	Spider	30s	60s	90s	120s	150s	180s	210s	240s
www.sina.com	Spider1/Pages	92	560	574	587	592	604	610	632
www.sohu.com	Spider2/Pages	343	447	466	494	1252	1258	1381	1507
www.yahoo.com	Spider3/pages	286	418	538	592	684	744	780	811
Total pages (Spider1+Spider2+Spider3)		721	1425	1578	1673	2528	2606	2771	2950
Redundant pages		12	27	49	95	127	139	154	162
Redundancy rate (%)		1.67	1.89	3.11	5.68	5.02	5.33	5.56	5.49

**Table 2.** Redundant pages and Redundancy Rate. (*Time interval =5 minutes*)

Seed URL	Spider	5min	10min	15min	20min	25min	30min	35min	40min
www.sina.com	Spider1/Pages	2670	5104	7528	10012	13874	17528	19876	22451
www.sohu.com	Spider2/Pages	2214	4512	6258	7845	12457	16450	18763	22314
www.yahoo.com	Spider3/pages	1907	3857	6421	7774	11053	16272	18354	21906
Total pages (Spider1,2,3)		6791	13473	20207	25631	37384	50250	56993	66671
Redundant pages		356	750	1148	1369	2194	2724	3232	3527
Redundancy rate (%)		5.24	5.57	5.68	5.34	5.87	5.42	5.67	5.29

designate the maximum search depth as 10 and the maximum number of visit sites as 50000, which are invariable in the following experiments.

In the first experiment, we plan to test the independent execution of the three spiders for recording the number of redundant pages caused by parallelization.

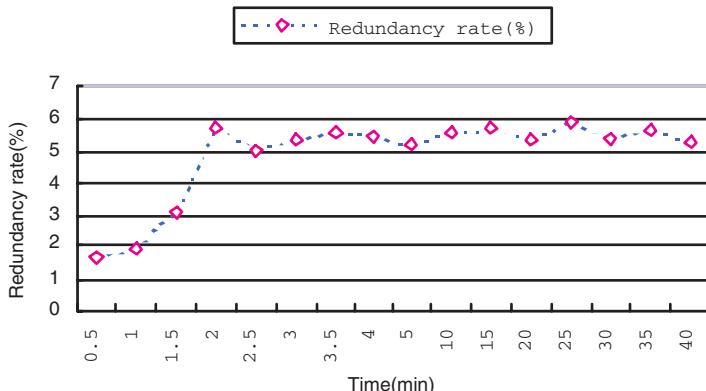
**Table 3.** Cooperative total pages quantity collected by three spiders coordinated by Facilitator Agent and Independent total pages quantity collected by three spiders running independently.(*Time interval =30 seconds*)

Seed URL	Spider	30s	60s	90s	120s	150s	180s	210s	240s
www.sina.com	Spider1/Pages	78	518	543	581	584	598	605	627
www.sohu.com	Spider2/Pages	310	421	432	457	1143	1240	1345	1487
www.yahoo.com	Spider3/pages	252	389	507	543	624	727	765	802
Cooperative total pages		640	1328	1092	1581	2351	2565	2724	2916
Independent total pages		721	1425	1578	1673	2528	2606	2771	2950
Efficiency Cost (%)		1.12	6.8	30.7	5.4	7.0	1.57	1.67	1.15

Thus, we can compute the redundancy rate using formula (1). Independent execution means we do not start the facilitator to work as the coordinator among spider agents. In the experiment, we first respectively record the number of web pages collected by the three spiders and then acquire the number of redundant pages through the database SQL operation. The experiment result is showed in the table 1 and 2. Figure 4 shows the variety of redundant rate in the whole experiment. From figure 4 , we find that the redundant rate is increasing immediately from 1.67% to 5.68% at first 120 seconds and then keeps stable between 5% and 6% in the following time. Because most search engine has to index millions pages, this redundant rate (5% - 6%) is unacceptable along with web pages gradually increasing.

$$\text{Redundancy\_Rate}(t) = \frac{\text{Redundant\_Page}(t)}{\sum_{k=1}^n |\text{Page}(\text{Spider}_k, t)|} \quad (1)$$

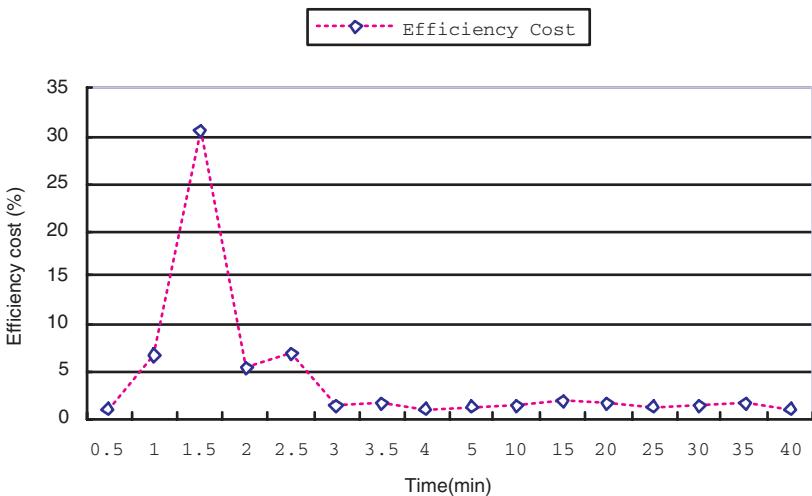
In order to avoid redundancy, we adopt the dynamic assignment mechanism which coordinates multi-spider agent using the facilitator. Here, we all know that interaction and coordination will cause cost. How much does it taken in our



**Fig. 4.** Redundancy rate when three spiders run independently(*Data from Table 1 and Table 2*)

**Table 4.** Cooperative total pages quantity collected by three spiders coordinated by Facilitator Agent and Independent total pages quantity collected by three spiders running independently.(Time interval =5 minutes)

Seed URL	Spider	30s	60s	90s	120s	150s	180s	210s	240s
www.sina.com	Spider1/Pages	2501	5048	7346	9825	12241	16347	19054	22689
www.sohu.com	Spider2/Pages	2195	4435	6704	8765	12029	16472	18997	22035
www.yahoo.com	Spider3/pages	2003	3778	5779	6641	12635	16642	17990	21180
Cooperative total pages		6699	13261	19829	25231	36905	49461	56041	65904
Independent total pages		6791	13473	20207	25631	37384	50250	56993	66671
Efficiency Cost (%)		1.35	1.57	1.87	1.63	1.28	1.57	1.67	1.15



**Fig. 5.** Efficiency Cost caused by parallelization run.(Data from Table 3 and Table 4)

system? Will the cost be reflected by the decreasing number of web pages collected by spiders at a fixed time scope? In order to answer these questions, we design two experiments. The one is independent run and the other is cooperative run which is controlled by the facilitator. We record the experiment data in table 3 and 4. Figure 5 shows the whole efficiency cost in 40-minutes experiment process. From figure 5, we find efficient cost acutely fluctuates at the first 150s and then trend stable (1%-2%) in the following time. This rate(1%-2%) is pervasive for most interaction programs, which, in other words, means our parallel spider system does not take additional cost.

$$\text{Efficiency\_Cost} = \frac{\text{Indepen\_Total\_Pages}(t) - \text{Coop\_Total\_Pages}(t)}{\text{Indepen\_Total\_Pages}(t)} \quad (2)$$

The above experiments investigate the performance of the parallel spider based on multi-agent environment MAGE.Result demonstrates that it is a

feasible and effective approach to apply multi-agent technology constructing parallel spider since it can not only improve the page collection performance but also avoid the redundancy with a tiny efficiency cost.

## 5 Conclusions and Future Work

Experiment shows that it costs about 5%-6% storing space for redundant web pages caused by parallelization if we do not adopt some effective measures to avoid it. In this paper, we present a parallel spider model and implement it on the multi-agent platform MAGE. It can efficiently solve redundant pages problem through multi-agent coordination and dynamic assignment mechanism. With the control of central facilitator, spiders (agents) can always find the inconsistent URLs at first time and initiates new spiders for search, which can effectively avoid redundancy at a very low interaction cost (1%-2%). In fact, it provides a novel and effective method for the design of next generation high performance web spiders. In future work, we plan to integrate the parallel web spider with a generic data-mining tool. Under this situation, various mature data mining technology can be applied to acquire useful information hid in the tremendous web pages and help us for decision support.

## Acknowledgements

The research work in this paper is supported by the National Basic Research Priorities Programme (No.2003CB317004), the 863 Project (2003AA115220) and National Natural Science Foundation of China (No.60435010).

## References

1. J. Michael Chau. Design and evaluation of a multi-agent collaborative Web mining system Decision Support Systems 35 (2003) 167- 183
2. Junghoo Cho, Hector Garcia-Molina. Parallel Crawlers Proc. of the 11th International World-Wide Web Conference
3. Zhongzhi Shi MAGE: An Agent-Oriented Programming Environment Third IEEE International Conference on Cognitive Informatics (ICCI'04) pp. 250-257
4. M. Burner. Crawling towards eterneity: Building an archive of the world wide web. Web Techniques Magazine, 2(5), May 1998.
5. L. Page and S. Brin. The anatomy of a large-scale hyper textual web search engine. In Proceedings of the Seventh Worldwide Web Conference, 1998.
6. A. Heydon and M. Najork. Mercator: A scalable, extensible web crawler. Word Wide Web, 2(4): 219-229, December 1999.
7. J. Cho and H. Garcia-Molina. The evolution of the web and implications for an in-cremental crawler. In Proceedings of the Twenty-sixth International Conference on Very Large Databases, 2000
8. R. C. Miller and K. Bharat. SPHINX: a framework for creating personal, site-specific web crawlers. In Proceedings of the Seventh World-Wide Web Conference, 1998.

9. D. Eichmann. The RBSE spider: Balancing effective search against web load. In Proceedings of the First World-Wide Web Conference, 1994.
10. J. Cho, H. Garcia-Molina, and L. Page. Efficient crawling through URL ordering. Computers networks and ISDN systems, 30:161-172, 1998.
11. S. Chakrabarti, M. van den Berg, and B. Dom. Focused crawling: A new approach to topic-specific web resource discovery. In The 8th International World Wide Web Conference, 1999.
12. D. S. MiloJicic, F. Douglis, Y. Paindaveine, R. Wheeler, and S. Zhou. Process migration. ACM Computing Surveys, 32(3):241-299, September 2000.
13. M. T. Ozs and P. Valduriez. Principles of Distributed Database Systems. Prentice Hall, 1999.
14. M. J. Quinn and N. Deo. Parallel graph algorithms. ACM Computing Surveys, 16(3), September 1984.
15. D. Nassimi and S. Sahni. Parallel permutation and sorting algorithms and a new generalized connection network. Journal of ACM, 29:642-667, July 1982.
16. Tim Oates, M. V. Nagendra Prasad and Victor R. Lesser Cooperative Information Gathering: A Distributed Problem Department of Computer Science University of Massachusetts Amherst, MA 01003, USA
17. Marian Nodine, Jerry Fowler, Tomasz Ksiezyk, Brad Perry, Malcolm Taylor and Amy Unruh. Active Information Gathering in InfoSleuth In International Journal of Cooperative Information Systems 9:1/2, 2000, pp. 3-28.
18. Larry M. Deschaine, Richard S. Brice and Marian H. Nodine. Use of InfoSleuth to Coordinate Information Acquisition, Tracking and Analysis in Complex Applications In Proceedings of Advanced Simulation Technologies Conference, April 2000.
19. M. Najork, J. L. Wiener. Breadth-first search crawling yields high-quality pages, Proceeding of 10th International World Wide Web Conference (WWW10), 2001.

# Intelligent Market Based Learner Modeling

Maryam Ashoori<sup>1</sup>, Chun Yan Miao<sup>1</sup>, Angela Eck Soong Goh<sup>1</sup>, and Wang Qiong<sup>2</sup>

<sup>1</sup> School of Computer Engineering, Nanyang Technological University, Singapore

<sup>2</sup> School of Education, Peking University, China

ashoori@pmail.ntu.edu.sg, {ascymiao, asesgoh}@ntu.edu.sg,  
qwang@gse.pku.edu.cn

**Abstract.** This paper presents an economical inspired intelligent approach for modeling learners in learning systems. Decision making in complex systems like e-learning systems requires processing of large amounts of heterogeneous data and information from disspread sources. Moreover, most of the decision parameters are incomplete and uncertain. Lacking of a complete model of learner is the prominent problem of current learning systems. In this paper, a market based method for describing Learner's preferences to the learning system is provided. The proposed approach strives for applying a Dempster-Shafer decision making over a society of self motivated agents. It tries to present a final learner agent with a high degree of similarity to the user for the purpose that it can act as a model of learner through the system. An implicit learning is also implemented by the idea of Stocks in real markets which can improve decision making efficiently.

**Keywords:** Intelligent agent, personalizing, user modeling, Marker based systems.

## 1 Introduction

The main issue in pedagogy is individualization, i.e., adapting the teaching to the needs of various learners. In many cases, however, current e-learning systems have so far focused most on porting existing courses with traditional teaching methods onto the web, just making non-individualized teaching even more widely available. Agent mediated E-learning systems are proposed to improve the effectiveness of learning achievements through an interactive learning environment. The proposed approach, however, is not limited to e-learning systems. This idea can be applied to a wide variety of application domains where personalizing the content could improve the final performance.

One of the main Problems for personalizing learning objects is how to get a model of user profile. In most of the previous offered algorithms for personalizing learning objects, it is assumed that we have an exact model of user and know user preferences completely. However, the assumption is not true in most cases. The majority of users hardly can express themselves. Therefore, a prominent problem is finding an effective approach to help users to explain their needs. This paper proposes an agent approach for modeling learner preferences where agents try to monitor the user's activity

pattern. Agents automatically adjust the content provided to accommodate such user differences.

Using intelligent agents, the main concern is achieving a satisfactory performance through an effective interaction among selfish agents. System management, therefore, emphasizes to present social strategies for organizing efficient coordination among self-interested individuals.

In other hand, complexity of a dynamic Multi-agent system (MAS) could be effectively governed by a market-based solution inspired of successful humanized societies. Markets can provide effective allocation of resources for a variety of distributed environments and afford related economic analyses as a powerful tool for designing interaction mechanisms [1]. A market-based MAS is any collection of software agents interacting through a price system where prices are a medium of information dissemination and aggregation.

This paper develops a society of self interested agents who negotiate through a market based protocol to present the most similar agent to the learner preferences. This market-based model autonomously apportions complex task of user modeling to multiple cooperating agents giving each agent the motivation of improving performance of the whole system.

By exploring market mechanisms for achieving efficiency in multi-agent domains, essentially we would like to know “what kind of humanoid meta solutions can be utilized to improve the overall performance and what social values may be defined as well”. The presented solution in this paper is an inspiration of actual stock exchange strategies where the social value is presented by a revision of stock values defined on the prominent factors of system. Since total profit is along the lines of individual profits in a market economy, stock updating algorithm should be designed directly base on concrescence in social and individual rewards [2].

Majority of multi-agent systems include non-homogeneous agents are personalized by their preferences. Here, the presented model personifies agents based on the learner characteristics and look for the most available specialized agent to use as the learner model for future decisions.

In this paper, related works are reviewed in the next section. Then, our propose approach is reviewed briefly through section 3. In Section 4 proposed methods are evaluated on a Simple learning environment and simulation results are reported and discussed there. Finally, the conclusions are given in Section 5.

## 2 Related Works

Most of the recent Market based solutions are price-based market where market value is bounded to asset of agents. Producing intelligence in credit assignment is the most critical problem in such systems where several agents collaborate on accomplishing a project.

Holland was the first author to use economic terminology in a multi-agent learning machine [3]. Assets could be created freely in his approach without supporting a sustainable society. To solve this problem, a reinforcement learning approach called

PSALMs (prototypical self-referential learning mechanisms) was introduced [4]. In Holland's systems, agents probably profit from detrimental achievements. Dealing with this problem "The Hayek Machine" was presented by BAUM. Agents of Hayek were learning how to reward their society. Some other methodologies are focuses on distributing risks and rewards in market based systems such as [5] which debates on balancing required to be between risk and rewards.

In learner modeling, while data is uncertain, imprecise or incomplete, fusion is a suitable solution for getting more relevant information. Garvey et al. [6] introduced the use of Dempster-Shafer reasoning for data fusion. The use of evidential reasoning for fusion allows each self interested agent to contribute information at its own level of detail. The Dempster-Shafer theory offers a knowledge model about one or more hypothesis, enables to quantify such concepts as imprecise measurements or uncertainty, and agrees to allocate probability-like weights to a set of events in a way that allows statements of ignorance about likelihood of some of the events. The Dempster-Shafer evidential is an extension to the Bayesian approach that makes explicit any lack of information concerning a proposition's probability by separating firm belief for the proposition from just its plausibility. The other issue in this paper is personalizing learner objects with an agent-based idea which is glanced further.

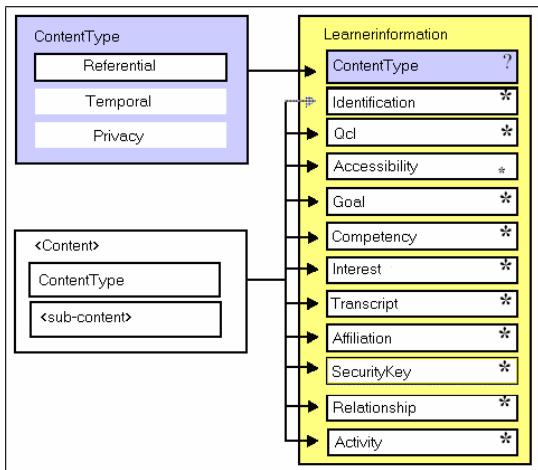
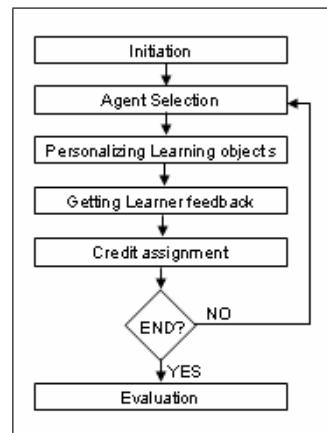
In the LOVE collection [7], learning objects are provided in standardized forms so that intelligent agents can index user profiles and learning objects and match them directly. They are using data mining techniques in the "Semantic Object Manager" and "Emerge" (search engine) for matching them. [8] Focuses on Classification where Software agents are associated with learning objects to support learners. Each kind of learning object has its own specific software agent to integrate the learning object into a learning environment. Such software agents can both interact with other agents and the learner to construct learning spaces.

For modeling learner profiles, one of the most popular standards for modeling learner profile is IMS LIP [9]. IMS consortium maintains an International Conformance Program to provide joint collaboration between communities interested in adapting the IMS specifications to their precise needs. IMS Learner Information package is based on a data model that describes characteristics of a learner needed for the general purposes of:

- Recording and managing learning-related history, goals, and accomplishments;
- Engaging a learner in a learning experience;
- Discovering learning opportunities for learners.

The core structures of the IMS LIP are based upon: accessibilities, activities, affiliations, competencies, goals, identifications, interests, qualifications, certifications and licenses, relationship, security keys and transcripts which are shown in figure 1.

IMS LIP data model seems to be a proper solution for modeling an applicant. However, it is really blurred and also too much complex to analyze. Definition of each term is brought in [9]. This paper employs the IMS LIP parameters and presents an adopted intelligent agent as the model of learner which acts on behalf of learner through the system and provides a personalized collection of learning objects for learner.

**Fig. 1.** IMS Learner Information Package**Fig. 2.** Six different phases of proposed approach

### 3 The Proposed Approach

In the following we propose an agent-based approach for learner modeling to be used in personalizing learning objects. In this approach we have a society of different agents who are self interested with their own specific preferences. Each agent is characterized by some of the parameters defined by IMS LIP. They try to provide different models of sample learners.

The main problem that this paper tries to solve is how to present an effective model of learner in an e-learning system. The proposed approach asks the society to vote and select the most suitable agent as a model of target learner. After defining the most suitable agent, based on the agent preferences the personalization algorithm will be applied and results will be shown to the learner.

A credit assignment method will be then applied to the society. Feedback of learner about the presented collection of learning objects is defined as the main parameter to determine how much rewards and punishments are required to be applied. If the User satisfaction is lower than a threshold, all the society will be punished for choosing a bad agent modeler and on condition that the degree is higher than the threshold, an overall reward will be applied through the society to adjust the coefficients and increase the chance of current learner agent during the next decision making.

According to systematic cognition, Interaction should be accompanied by a special additive manner causing each individual, as a member of a society, to elevate own ability through a preplanned interaction protocol. A well-defined social value is, therefore, required to develop such this shared area thoroughly. This proposed credit assignment in this paper is an inspiration of investment in real stock markets. Investment is partial immolating of status quo for getting an extra profit in the future. It has been divided into two distinct categories consist of tangible and financial investment. In first type, investor directly attains a pecuniary actual asset in return. In contrast, the second is commonly achieving a flow of profits through immolating the present asset

same as stock exchange investment. Our proposed approach covers both types. It means that via presenting an effective combination of different learning objects, not only the modeler agent's pecuniary asset may be increased but also further stocks may be assigned to the doer's fund. Organizing stock value as a dominant shared social value makes it possible to simulate a balanced risk and reward assignment. Since selecting an inappropriate modeling agent in our approach makes all the society punished as well, an implicit cooperative learning is emerged which makes inappropriate agents being removed from the market gradually[10]. Thus, MAS is developed as a stock exchange market where agents become investors and learner parameters remain marketable commodities. Proposed approach can be described over six different phases which are shown in Figure 2.

With the intention of replicating a pseudo market, agents get initiated by a predetermined stock amount and initial asset based on their similarity with learner parameters. Each stock is related directly to the amount of similarity between agent and learner. The higher amount of stock means the higher similarity. After Initiation, the most suitable agent will be selected as the learner model and Personalization approach will be applied over learner agent who is exactly known for system. While in this paper the main focus is learner modeling, the personalization algorithm is assumed to be completely defined.

Agent selection and credit assignment algorithms as two main challenging stages are stated formally in the following.

#### *A. General Assumptions*

There are some assumptions about the agent's knowledge and the system design. First, it is assumed that we have complete information about current learning objects and learning objects are assumed like a common package of data here. This assumption is required while we just want to focus on the incomplete data on the side of learner and we try to provide an effective model of final learner.

Furthermore, each agent has a pseudo-learning characteristic. If the society presents an effective combination of personalized learning objects, a reward assignment is applied on all the society. This simulates a close effect to learning and expertness.

#### *B. Agent Selection*

Agent selection as the main part of our approach can affect the overall performance of learning drastically. So, it's a challenging part to design an effective negotiation protocol over MAS to select the most suitable agent. English auction and a Dempster based voting are two approaches used here as agent selection methods. In auction-based approach, the most suitable agent is determined based on the initial preferences of agents and degree of similarity to the learner parameters. Agents bid based on the similarity degree and the highest bid would be the winner agent.

Our auction-based task allocation algorithm can be presented in a three-step approach:

- 1) Evaluating the price of the pecuniary capital of agent, which is a combination of financial asset, and the price of all agents' stocks based on (1).

$$\begin{aligned} Fund(Agent) = Asset(Agent) \\ + \sum_{i \in Tasks} Portfolio(Agent, i) * PortfolioValue(i) \end{aligned} \quad (1)$$

Where *portfolio* is the number of stocks of each agent and *PortflcioValue* is the stock value of Learner. This parameter emphasizes to this fact that in a similar group of learners in a class as an example, the probability of previous learner agents is higher than the others. It is to simulating the similar factors that shape a group of learners. Here, the stock value is assumed to be 1. The assumption is for ignoring the stock value. This value is determined based on the target learners. While our aim is personalizing learning objects for a unique learner we can ignore stock value else on condition that we want to apply the proposed approach simultaneously for personalizing learning objects for a group of learners, we can adjust these stock values based on the importance of learner as an example.

2) Calculating a fuzzy similarity degree between agent's preferences and learner's parameters:

$$\begin{aligned} \forall i \in Agents, j = Learner, \\ Similarity(i, j) = F(Agent's\_Params(i), Learner's\_Param(j)) \end{aligned} \quad (2)$$

*Learner's\_Param* is the collection of real questions which are answered and determined by user directly. Based on IMS LIP, considered learner parameters are previous activities, goals as the title of the target course, and interests which are presented through selecting just some options through the systems. While this bounded information can not describe user preferences completely, the proposed approach itself tries to complete user preferences based on learner's feedbacks through the system.

*Agent's\_Params* are some especial parameters which are related to *Learner's\_Param* and are adapted to the requirements of personalizing learning object algorithm.

3) Bidding based on the similarity value and pecuniary capital:

$$\begin{aligned} \forall i \in Agents, j = Learner, \\ U(i, j) = \alpha * Fund(i) + \beta * Similarity(i, j) \end{aligned} \quad (3)$$

Where  $U(i, j)$  is the bid value offered by agent  $i$  for modeling learner  $j$ .

Next step is agent selection via participating in an English auction over the Bids' values trough a distributed decision-making where the winner with the highest bid would be selected as the learner agent.

Beside this method, we examined a Dempster based voting as negotiation protocol and compared it with previous approach. As far as Data fusion is an advance method utilized for combining different values got from environment to prepare them for an improved decision making, we made our mind up to use Dempster-Shafer theory as one of the most famous mathematical methods of this approach. Our Dempster based voting is proposed over Dempster-Shafer Theory of Evidence that defines a frame of discernment and tries to assign a basic probability  $m(E)$ , as the degree of belief of evidence where:

$$\begin{aligned} \sum_{A_j \in 2^{\Theta}} m_i(A_j) &= 1 \\ m_i(\emptyset) &= 0 \end{aligned} \quad (4)$$

Frame of discernment in our approach is determined dynamically. During a voting, a set of all available agents constitutes as frame of discernments for each modeling.  $m(E)$  is the current belief for each of the subsets of frame of discernment. According to agents' observation, the probability that "a particular agent is suitable to be the

learner agent" is indicated by a "confidence interval" whose lower bound is a "belief" and whose upper bound is a "plausibility": [Belief, Plausibility]

While we have to combine several beliefs of agents to accomplish a real voting, we can utilize Dempster rule of combination to Combine Evidences. Dempster rule of combination for combining two distinct evidences is:

$$m_1 \oplus m_2(Z) = \frac{\sum_{X \cap Y = Z} m_1(X).m_2(Y)}{1 - \sum_{X \cap Y = \emptyset} m_1(X).m_2(Y)} \quad (5)$$

By associating "belief" with the lower end of a probability range and "plausibility" with its upper end, the Dempster-Shafer approach manages to capture the key features of the human perception-reasoning process. In contrast, the Bayesian approach, which is essentially a subset of the Dempster-Shafer approach, provides no mechanism for dealing quantitatively with the ranges of "belief" and "plausibility" that humans characteristically attach to their estimates of likelihood.

In this approach similar to previous auction, the similarity degree of each agent will be estimated first through the embedded fuzzy system presented in (2). Then, the degree of belief of evidences will be determined based on the market situation as well as agents' assets through (6).

$$\begin{aligned} \forall i \in \text{Agents}, j = \text{Learner}, \delta = \text{Voter}, A = \{\text{a subset of } \Omega\} \\ m_\delta(i, j) = (\alpha * \sum_{i \in A} \text{Fund}(i) + \beta * \sum_{i \in A} \text{Similarity}(i, j)) \\ * \lambda / \sum_{i \in A} (\text{Portfolio}(i, j) - \text{Portfolio}(\delta, j)) \end{aligned} \quad (6)$$

Where  $\lambda$  is the factor used to make the summation of mass functions to be 1,  $\delta$  is belief degree of agent  $i$  for modeling learner  $j$  which is calculated by asset, similarity, and share value.

After estimating the degree of belief of evidences, we ask agents to vote on selected agent base on Dempster rule of combination which is defined in (5).

### C. Credit Assignment

Just after Agent Selection, a personalized learning object algorithm will be run by the learner agent and results will be presented as a collection of different learning objects to the learner. Then, Learner will be asked to define the percentage of efficiency of presented collection. This percentage would be the main factor for applying reward or punishment over the previous operation of agent society. If the Satisfaction factor is lower than the threshold, all the society will be punished for choosing a bad agent modeler and on condition that the feed backed degree is higher than threshold, an overall reward will be applied through the society to adjust the coefficients and increase the chance of current learner agent during the next decision making.

For credit assignment after getting the feedback of learner, In case satisfaction is lower than threshold, asset of Learner agent may be reduced as well as the stock value of learner by (7). This is to prevent the mistake of selecting this especial agent as learner agent in next decision making.

$$\forall i = \text{Learner agent}, j = \text{Learner},$$

$$\begin{cases} PortfolioValue(j) = PortfolioValue(j) + \theta * (Feedback(j) - T) \\ Portfolio(i) = Portfolio(i) + \theta * (Feedback(j) - T) \\ Asset(i) = Asset(i) - \gamma * (Feedback(j) - T) \end{cases} \quad (7)$$

Once a satisfaction degree is higher than threshold, asset and share value of agent will be increased through (7).

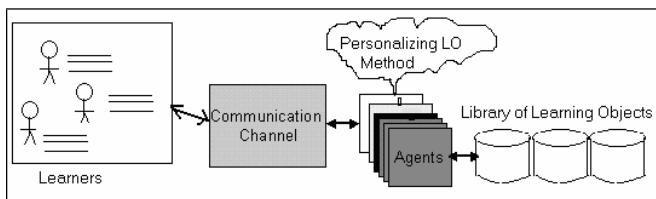
By this formula, getting a high percentage of satisfaction is increasing the amount of learner stocks devoted to learner agent and therefore is raising the chance of being selected during next period of execution.

In our method, stock is assumed as a dominant parameter that implements an implicit unsupervised learning over the market in order to applying both social rewards and total punishments. Revision in stock values is promoting the level of agents' knowledge to get a profitable decision. All shareholders profit indirectly by selecting a fully similar agent and increasing the learner stock value. Consequently, market goes to have an optimal task allocation by reducing the dissimilar agents gradually.

## 4 Experiments and Results

Testing the exact effects of this method in real e-learning environment requires a comprehensive benchmark consists of different variant learning objects where most of their metadata are often incomplete. While currently we are working on providing efficient intelligent personalized learning objects, we don't have any effective algorithm at this time to apply on our user modeling approach and check the results. On the other hand while designing an effective approach of personalizing requires an effective model of learner preferences, we have focused here on learner modeling instead. Our current personalized Learning object approach is a simple algorithm which is able to connect to a repository of 100 learning objects with complete specific metadata adapted by the parameters which determines the preferences of agents through the system. And personalizing Learning objects approach is just a matching method which retrieves required data by running a query.

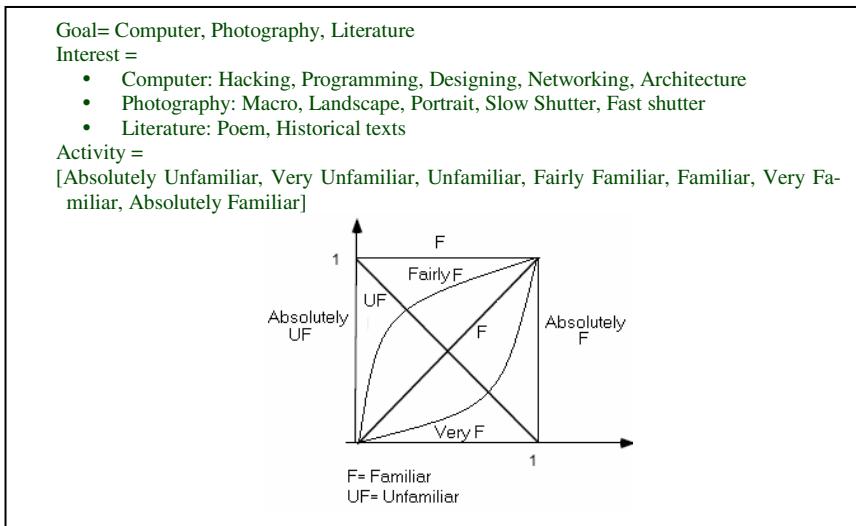
A very simple structural view of the evaluation system is shown below in figure 3.



**Fig. 3.** The MAS Schematic View

We have considered a group of 5 learners through this system. 50 different agents are also designed randomly through the system with their especial preferences. In this evaluation system, we are trying to model these 5 learners periodically and find the most similar agents to their preferences. The way and parameters that we have defined for evaluation approach is presented in figure 4.

These values are initiated by the learner at the start of execution and will be updated during the execution. Among all the IMS LIP items, the related items to preferences are assumed to be activity, goal and interests while the system is basically built to serve as a platform for students as supplement learning environment rather than organization selection of new employees.



**Fig. 4.** Mapping IMS LIP in Our Evaluation System

Two main Evaluation parameters have been considered here. First, Learner Satisfaction factor which is learner's feedback to the system which we are trying to maximize. Second one is the total asset of agents as a parameter that reflects the overall performance of society over decision makings. Total asset of the market after each personalization will be calculated through (1) and (8).

$$\text{TotalAsset} = \sum_{l=1}^{50} \text{Fund}(l)) \quad (8)$$

Table 1 demonstrates constant parameters used in this approach.

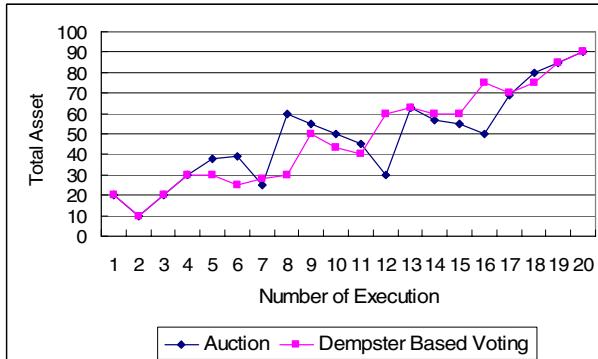
**Table 1.** Constant Parameters

Parameter	$\alpha$				T
Value	0.1	0.5	0.1	0.05	0.5

In our proposed methods, two distinct negotiation protocols are presented. Results of belief combination based on Dempster rule of combination is demonstrated in figure 5 in comparison to the presented auction through 20 times execution while each time 2 personalization has been done for each learner. Vertical axis is Total asset of society calculated by (7) for all 50 agents of society. While based on the learner

satisfaction, the value of assets of agents are modifying through the system, it is possible to consider the raising of the value of total asset as a sign of increasing in performance.

According to this figure, Dempster based voting didn't have the slight fluctuations have been seen in Auction. Therefore, it seems Dempster based voting is an efficient negotiation protocol for this type of agent selection.



**Fig. 5.** Comparison between Dempster based voting and English auction

In figure 6, Dempster based voting is assumed as the negotiation protocol and the average of learner's satisfaction is presented during times of execution. In each time average value is calculated over 5 learners while each of them has taken two different courses.



**Fig. 6.** Learner's satisfaction Average through applying the Dempster based votin.

## 6 Concluding Remarks and Future Works

A cooperative learning strategy based on an inspiration of market solutions is proposed to solve the problem of user modeling through a multi agent system. This approach presents an economical view to achieve learner satisfaction as well as a satisfied social profit through agent personification beside stock management.

In this approach, while some agents have closer similarity with target learners, a high initial pecuniary asset, in proportion of the Learner's feedback will be raised to model the learners better. This means that by selecting a suitable learner agent and incrementing the stock amounts of selected agent of that particular target learner, its chance of being selected during the next periods of execution may go up considerably.

Dempster based voting requires a really heavy computational time. The results illustrates that we can effectively model the target learner while we have incomplete information about learner and while learners couldn't draw a complete view of themselves for the system. Here, the main idea is in fact assigning similar agents with completely known preferences. Besides, by adjusting stock parameters while these stocks are defined on target learners and are distributed to all the agents of system based on their similarity degree, we can simulate an implicit learning among society. While agents try to select learner agent by voting, through the history and also current state of system, it is possible to improve the value of parameters that they use for decision making.

As the final point, it seems the efficiency of our agent selection could be better if we utilize some explicit learning approaches such as reinforcement learning as well as agent clustering and implementing common stocks to simulate collaborative agents. While our credit assignment algorithm is inspired of real stock market, this work can be continued by working on computational economies, getting their mathematical solutions and applying them over society of agents. We can also utilize pedagogy achievements in addition to psychological reaching to present a comprehensive and complete model of learner.

## References

1. Wellman, M., Wurman, P., Market-aware agents for a multi agent world, *Robotics and Autonomous Systems*, vol. 24, pp.115-125, (1998).
2. Tsvetovat, M., Carley, K. and Sycara, K., *Emergence of market segmentation: a multi-agent model*, CASOS and Robotics Institute, (2002).
3. Holland, J. H., *Properties of the bucket brigade*, Proceedings of an International Conf. on Genetic Algorithms, Hillsdale, NJ, (1985).
4. Schmidhuber, J., Evolutionary principles in self-referential learning, or on learning how to learn the meta-meta, Institut f'ur Informatik, Technische Universit'at M'unchen, (1987).
5. Irwin, D.E, Grit, L.E., and Chase, J.S., Balancing Risk and Reward in a Market-based Task Service, Report from Duke University, (2004).
6. Garvey, T.D., Lowrance, J.D., and Fischler, M.A., *An inference Technique for integrating knowledge from disparate sources*, Proceedings of the 7th Int. Joint Conf. Artificial intelligence (pp. 319-325). Vancouver, B.C., (1981).
7. Chen, S., Rodriguez, O., C. Chee-Yoong, Yi Shang and Hongchi Shi , *Personalizing Digital Libraries for Learners*, Proceedings of the 12th Int'l Conf. Database and Expert Systems Applications (DEXA'2001), Munich , Germany ,September (2001).
8. Hawryszkiewycz, I.T., *Agent Support for Personalized Learning Services*, Proceedings of the 3rd IEEE Int'l Conf. Advanced Learning Technologies (ICALT'03), (2003).
9. IMS Learner Information Package Specification, Downloaded: January 2006, <http://www.imsglobal.org/profiles/> , (2006)
10. Ashoori, M., Nili, M., and Moshiri, B., *Toward a model of an improved economy of agents*, Proceedings of the 2nd Int'l Conf. on machine intelligence, November, (2005).

# User Preference Through Bayesian Categorization for Recommendation

Kyung-Yong Jung

School of Computer Information Engineering,  
Sangji University, Korea  
kyjung@sangji.ac.kr

**Abstract.** The personalized recommendation system is required to save efforts in searching the items in ubiquitous commerce, it is very important for a recommendation system to predict accurately by analyzing user's preferences. A recommendation system utilizes in general an information filtering technique called collaborative filtering, which is based on the ratings matrix of other users who have similar preference. This paper proposes the user preference through Bayesian categorization for recommendation to overcome the sparsity problem and the first-rater problem of collaborative filtering. In addition, to determine the similarity between the users belonging to a particular class and new users, we assign different statistical values to the items that the users evaluated using Naive Bayesian classifier. We evaluated the proposed method on the EachMovie datasets of user ratings and it was found to significantly outperform the previously proposed method.

## 1 Introduction

Collaborative filtering, which recommend items to users based on judgment of other users, are the ubiquitous recommendation system used much on the latest recommendations. Usually, the input data is composed of users and a two-dimension matrix in collaborative filtering. The row in matrix represents users, the column represents the list of items, and the value of matrix represents user's preference estimated about items. The recommendation system should predict the most suitable goods for users by retrieving and analyzing their preferences. The recommendation system utilizes in general an information filtering technique called collaborative filtering which is widely used for such recommendation systems as Ringo, Fab [4]. Collaborative filtering calculates the similarity between the test user and each of other users who have rated the items that are already rated by the test user. Collaborative filtering uses the *Pearson correlation coefficient* for calculating the similarity, but it assumes that there must exist at least one item that has already been rated by both the test user and one of the other users [6]. Collaborative filtering works quite well in general, because it is based on the ratings of other users who have similar preference. However, collaborative filtering may not provide high quality recommendations for the user consistently. Collaborative filtering has the first-rater problem because they recommend based on the interest of users in items, not considering the items'

contents. Also, in case users estimate preference on a lot of items, systems have a shortcoming that the accuracy of prediction is fallen due to the sparsity of a rating matrix [1][5]. The other reason of sparsity for a rating matrix in collaborative filtering is the missing value caused from partial rating on items [2].

The sparsity problem and the first-rater problem, and missing value are solved by reconstructing user preferences, which are elements of a rating matrix, based on the user preference through Bayesian categorization. The proposed method was tested in the EachMovie dataset which user rated preference rating on items, and proved its effectiveness compared with existent methods.

## 2 Ratings Matrix in Collaborative Filtering

The aim of collaborative filtering is to recommend new items of interest for a particular user on the basis of users' preferences. Collaborative filtering recommends items through the profiles of the users from their preferences for each item. In collaborative filtering, preferences are represented generally as numeric values that are rated by the users. Predicting the preference for a certain item that is new to the test user is based on the ratings of other users for the target item [6]. A variety of collaborative filtering algorithms have been reported and their performance has been evaluated empirically. The collaborative filtering system based on items recommends an item to users according to a rating matrix [11].

### 2.1 Problem Space of Collaborative Filtering

The users in collaborative filtering don't rate preferences on all items. Therefore, the missing value is occurred in a rating matrix. The missing value causes the sparsity problem and first-rater problem of a rating matrix [3][5][7][15]. Since most users generally do not rate most items, similarities between users calculated from a sparse rating matrix may be inaccurate. As a result, the accuracy of prediction may be poor. To solve the drawbacks, many techniques have been proposed. The methods used in [1][5] solve only the first-rater problem not the sparsity problem. The method used in [12] decreases the dimension of a rating matrix with SVD to capture the similarity in a reduced dimensional matrix. However, it dose not solve the first-rater problem. The method used in [11] use content-based filtering [2] to fill out un-rated entries in a sparse rating matrix. In addition, a method used in [12], it just tried to solve both the sparsity problem and the first-rater problem. In this paper, the user preference generation is mentioned to reduce the sparse rating matrix caused by the missing value.

### 2.2 Rating Matrix

If we define  $m$  items which are composed of  $p$  feature vectors and a group of  $n$  users, user group is expressed as  $U=\{user_i\}(i=1,2,\dots,n)$ , item group is expressed as  $I=\{item_j\}(j=1,2,\dots,m)$ . And  $R=\{r_{ij}\}(i=1,2,\dots,n \ j=1,2,\dots,m)$  is a rating matrix. The element in matrix  $r_{ij}$  means  $user_i$ 's preference to  $item_j$ . Table 1 is an example of a rating matrix, where rows correspond to users, columns correspond to items, and the matrix entries are ratings. Typically, the rating matrix is sparse. Sparse in this context

means that some entries in the matrix are empty, because each user typically rates only a very small subset of all items [11]. The prediction task can be seen as filling the target user's empty elements of the matrix.

**Table 1.** Example of a rating matrix

	<i>item<sub>1</sub></i>	<i>item<sub>2</sub></i>	<i>item<sub>3</sub></i>	<i>item<sub>4</sub></i>	...	<i>item<sub>j</sub></i>	...	<i>item<sub>m</sub></i>
<i>user<sub>1</sub></i>	<i>r<sub>1,1</sub></i>	<i>r<sub>1,2</sub></i>	<i>r<sub>1,3</sub></i>	<i>r<sub>1,4</sub></i>	...	<i>r<sub>1,j</sub></i>	...	<i>r<sub>1,m</sub></i>
<i>user<sub>2</sub></i>	<i>r<sub>2,1</sub></i>	<i>r<sub>2,2</sub></i>	<i>r<sub>2,3</sub></i>	<i>r<sub>2,4</sub></i>	...	<i>r<sub>2,j</sub></i>	...	<i>r<sub>2,m</sub></i>
...	...	...	...	...	...	...	...	...
<i>user<sub>i</sub></i>	<i>r<sub>i,1</sub></i>	<i>r<sub>i,2</sub></i>	<i>r<sub>i,3</sub></i>	<i>r<sub>i,4</sub></i>	...	<i>r<sub>i,j</sub></i>	...	<i>r<sub>i,m</sub></i>
...	...	...	...	...	...	...	...	...
<i>user<sub>n</sub></i>	<i>r<sub>n,1</sub></i>	<i>r<sub>n,2</sub></i>	<i>r<sub>n,3</sub></i>	<i>r<sub>n,4</sub></i>	...	<i>r<sub>n,j</sub></i>	...	<i>r<sub>n,m</sub></i>

Collaborative filtering system uses information that users rate the preference for the items. Preference levels are represented on a scale of 0~1.0 in increments of 0.2, a total of six degrees, only when the value is higher than 0.5 is the user classified as showing interest. The items used are movie-related items gleaned by the EachMovie.  $r_{i,j}$  in Table 1 is defined as  $r_{i,j} \in \{\phi, 0, 0.2, 0.4, 0.6, 0.8, 1\} (i=1,2,\dots,n)(j=1,2,\dots,m)$ . Namely, the element of matrix  $r_{i,j}$  is in one of six degrees or no evaluation case.  $\phi$  means that collaborative filtering  $user_i$  doesn't rate  $item_j$ .

### 3 User Preference Through Bayesian Categorization

#### 3.1 User Similarity Weight

The Naïve Bayesian classifier is one of the most successful algorithms on many classification domains. Despite of its simplicity, it is shown to be competitive with other complex approaches especially in text categorization and content based filtering. The Naïve Bayesian algorithm classifies items in two phases, viz. the learning phase and classification phase [10][14]. The learning phase assigns a probability to an item from the training set, which consists of data collected through the item. Equation (1) is used to accord a probability to  $item_i$  within  $classID$ . The probability of  $item_i$  within  $classID$  is expressed as  $P(item_i|classID)$ . In this case,  $n$  is the total number of items in the training set whose target value is  $classID$ ,  $n_k$  is the frequency of  $item_i$  in  $classID$ , and  $|TotItem|$  is the total number of items.

$$P(item_i|classID) = \frac{n_k + 1}{n + |TotItem|} \quad (1)$$

The learning process is divided into the accumulation and assigning probability phases. In the accumulation phase, the number of items is the accumulated number of items existing within  $classID$ . In the assigning probability phase, the output of the accumulation phase is applied to Equation (1) and the probability is assigned to the items of the training set. Through this process, the probability information is added to the items in  $classID$ .

The user similarity weight differently is applied to the preference of the items that users evaluated as an object to the estimated value. For this process, as the user applies the learning set which the estimated value granted to the item rating the preference, the class item;  $P(item_i|classID)$  is multiplied by  $r_{a,k}$ .  $r_{a,k}$  represents the preference that  $user_a$  accords to  $item_k$ .

$$\beta_{a,k} = P(item_i|classID) \times r_{a,k} \quad (2)$$

By applying the *Pearson correlation coefficient* [3][8][13] based on Equation (2), the user similarity weight between  $user_a$  and  $user_i$  is defined by Equation (3).

$$\beta(a,i) = \frac{\sum_k (\beta_{a,k} - \bar{\beta}_a)(\beta_{i,k} - \bar{\beta}_i)}{\sqrt{\sum_k (\beta_{a,k} - \bar{\beta}_a)^2 \sum_k (\beta_{i,k} - \bar{\beta}_i)^2}} \quad (3)$$

Where  $\beta_{a,k}$  refers to the preference granted to the estimated value between  $user_a$  and  $item_k$ ,  $\bar{\beta}_a$  refers to the average value assigned to the weight value of the item for which  $user_a$  has already input the preference,  $k$  refers to the item for which  $user_a$  and  $user_i$  have input the preference in common.

If  $user_a$  and  $user_i$  have similar ratings for an item,  $\beta(a,i) > 0$ . We denote  $|\beta(a,i)|$  to indicate how much  $user_a$  tends to agree with  $user_i$  on the items that both users have already rated [6]. In this case,  $user_a$  is a positive neighbor with respect to  $user_i$ , and vice versa. If they have opposite ratings for an item, then  $\beta(a,i) < 0$ . Similarity,  $user_a$  is a negative neighbor with respect to  $user_i$ , and vice versa. In this case  $|\beta(a,i)|$  indicates how much they tend to disagree on the item that both again have already rated. Hence, if they don't correlate each other, then  $\beta(a,i) = 0$ . Note that  $\beta(a,i)$  can be in between -1 and 1 inclusive.

## 3.2 Categorizing Users

The Naïve Bayesian classifier [10] classifies  $user_{new}$  according to class by using the learning set with the estimated value, as described in Equation (4).

$$classID = \arg \max_{class \in classTot} p(classID) \prod_{i \in I} r_{a,k} \cdot p(item_i|classID) \quad (4)$$

$user_{new}$  with a preference value for each item is represented by  $user_{new} = \{x \in p | item_1(x), item_2(x), \dots, item_n(x)\}$ .  $item_n(x)$  refers to the item whose preference the user rates. The class in which  $item_{new}$  is to be categorized is represented by  $classID$ , and all of the classes are denoted by  $classTot$ .  $P(item_i|classID)$  is the probability according to Equation (4) and  $P(classID)$  is the probability classified by  $classID$ .

The class selected by  $user_{new}$  is that which has the highest probability value. The preferences are applied to the estimated value differently depending on the class. First, the learning set, which uses estimated values, is applied to those items which the user rates. Second, the user preference is multiplied by the class items. As shown, we can reduce the prediction error caused by the missing values. In addition, we assign the weight to the user preference differently depending on the class, by categorizing

the items. This method provides the weight based on the statistical value, rather than just on the preference [5].

### 3.3 Prediction for New Users

The prediction made for  $user_{new}$  utilizes the similarity weight between  $user_a$  and  $user_i$  by applying the Bayesian estimated value to the *Pearson coefficient correlation*. Through the weighted average of the similarity with the neighbor distance between the preference of the neighbors selected for the specific item and the preference of the mean value for each neighbor, the preference for the  $user_a$ 's  $item_k$  can be predicted. Equation (5) shows the definition.

$$P_{a,k} = \bar{r}_a + \sum_{i=1}^{neighbor} \beta(a,i)(r_{i,k} - \bar{r}_i) / \sum_{i=1}^{neighbor} \beta(a,i) \quad (5)$$

Where  $P_{a,k}$  represents the prediction of the preference that the estimated value is granted for  $user_a$  about  $item_k$ .  $r_a$  means the preference value granted the  $user_a$ 's weight value. *Neighbor* represents the fact that similarity between  $user_a$  and the other users is not 0, but is equal to the number of neighbors. The existing collaborative filtering uses the similarity weight between  $user_a$  and *neighbor*, as defined by the *Pearson correlation coefficient*, but  $\beta(a,i)$  is calculated from the user similarity weight by using Equation (3).

## 4 Evaluations

For the evaluation, we used the preprocessed data from the EachMovie dataset [9]. It was part of a research project at the DEC systems research center. This dataset was processed for a total of 20,864 users, such that each user rated at least 80 movies and 1,612 kinds of movie through the data integrity. User ratings were recorded on a numeric six-point scale, ranging from 0 to 1(0, 0.2, 0.4, 0.6, 0.8, 1). This constituted the training set for the Naïve Bayesian learning phase. 20,862 users were selected for the systematic sample base on the preprocessed EachMovie. In the EachMovie datasets, one of the valuable attributes of an item is the genre of a movie. The 10 genres were classified under the labels of the following classes. There are ten different classes as shown in Table 2.

**Table 2.** The class attributes of a movie

Action	Animation	Art-foreign	Classic	Comedy
Drama	Family	Horror	Romance	Thriller

The training set is constructed from the items for which the preference is determined based on the user's representative attribute [5]: Table 3 shows the classification of the items into 10 classes. The learning set is obtained from the training set by assigning the estimated value. Many users selected the action and drama classes as the representative attributes, because most users like these two types

of movies. The item of the training set is learned, in order to grant the estimated value by means of the Naïve Bayesian classifier.

**Table 3.** Training set

ClassID	Items rated the preference	Users	Items
Action	Golden eye, Clueless, Monkeys, Star gate, Star Wars, Drop Zone	9,637	1,502
Animation	Toy Story, Eden, Heavy Metal, Pocahontas, Space Jam, Robin	115	163
Art/Foreign	Four Rooms, Birdcage, Antonia's Line, Birdcage, Stalker, Diva	353	961
Classic	Jumanji, Balto, Happy Gilmore, Foreign Student, Alien, Annie	245	1,541
Comedy	Ace Ventura, Bronx Tale, Fatal Instinct, Four Rooms, Heather	2,107	1,545
Drama	Sabrina, Nixon, Ventura, Clerks, Get Shorty, Noon, Cape Fear	7,566	1,604
Family	Casper, Apollo13, Bad Boys, Batman Forever, Fly Away Home	158	1,248
Horror	Copycat, Screamers, Mary Reilly, Babe, Clueless, M, Braindead	87	453
Romance	American President, Swiss Family Robinson, Benny & Joon	166	603
Thriller	Die Hard, Taxi Driver, Crimson Tide, The Net, Breakdown, Water	428	916

Table 4 shows information about the missing values, for which the weight value of the user preference for a particular item is assigned differently depending on the classified class in the learning set. If  $user_{new}$  is classified according to the Naïve Bayesian classifier,  $user_{new}$  is differently granted the estimated value to item, in order to include similarity between the users in the classified class and  $user_{new}$ . The shaded boxes represent the predictive preference obtained by granting the Bayesian estimated value to the missing value using Equation (5).

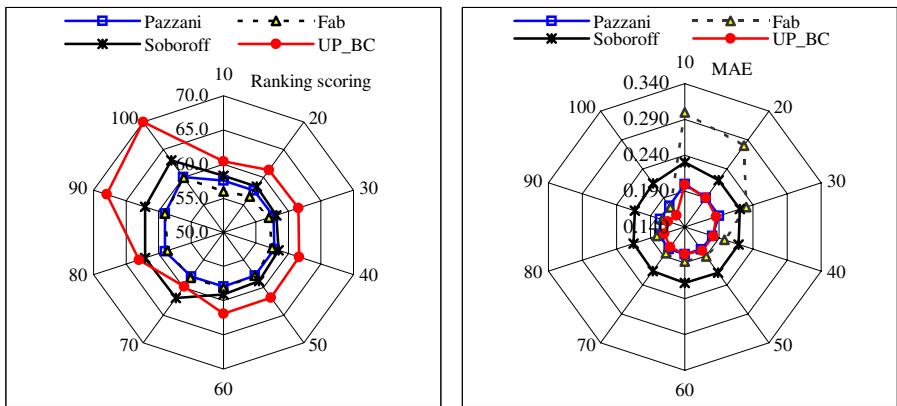
**Table 4.** Matrix given to Bayesain estimated value

	$item_5$	$item_6$	$item_{10}$	$item_{18}$	...	$item_{21}$	...	$item_{127}$
$user_1$	0.0025	0.0038	0.0098	0.0035	...	0.0097	...	0.0084
$user_{19}$	0.0012	0.0015	0.0024	0.0018	...	0.0075	...	0.0027
$user_{21}$	0.0020	0.0037	0.0034	0.0023	...	0.0037	...	0.0037
$user_{35}$	0.0008	0.0013	0.0006	0.0019	...	0.0026	...	0.0008
$user_{45}$	0.0012	0.0020	0.0017	0.0026	...	0.0047	...	0.0007
$user_{48}$	0.0023	0.0028	0.0031	0.0018	...	0.0035	...	0.0061

In this paper, rank score measure (RSM) and mean absolute error (MAE), both suggested by paper [3] are used to gauge performance. MAE is used to evaluate single product recommendation systems. RSM is used to evaluate the performance of systems that recommend products from ranked lists [4][7].

We considered the proposed method using the user preference through Bayesian categorization (UP\_BC) and hypothesized that it would outperform the stand-alone collaborative filtering approach. However, it is also important to compare our approach with that obtained using a combination of content-based filtering and collaborative filtering. By comparing the method proposed and those proposed by Soboroff [13], Pazzani [12] and, Fab [1], we are able to compare with the predictive accuracy values, such as the Rank scoring and MAE. Figure 1 shows the Rank

Scoring and MAE as the frequency with which the user evaluates the  $n$ th rating number is increased. Figure 1 shows that the system proposed by Soboroff [13] that solves the first-rater problem exhibits lower performance when the number of evaluations is lower. The other methods show higher performance than that of Soboroff. As a result, the method developed by Pazzani and the UP\_BC method that solve both the sparsity problem and first-rater problem show the highest accuracy rates.



**Fig. 1.** Rank scoring, MAE of  $n$ th rating

## 6 Conclusions

It is crucial for a recommendation system to have a capability of making accurate prediction by retrieving and analyzing of user's preferences. Collaborative filtering is widely used for recommendation systems. Hence various efforts to overcome its drawbacks have been made to improve prediction quality. In this paper, the user preference through Bayesian categorization is proposed in order to solve the sparsity problem and the first-rater problem. We calculate the weight value of the preference using the Bayesian estimated value differently. So reflect the information of item to the statistical value. The proposed method was compared with the existent methods that combine collaborative filtering and content based filtering. As a result, the proposed method shows higher performance than existent method in both comparisons. Since this mechanism is not limited to the EachMovie dataset, it can be extended to other domain, such as fashion design, music, and books.

## References

1. Balabanovic, M., Shoham, Y.: Fab: Content-based, Collaborative Recommendation. *Communication of the Association of Computing Machinery*. (1997) 66-72

2. Basu, C., Hirsh, H., Cohen, W.: Recommendation as Classification: Using Social and Content-based Information in Recommendation. Proceedings of the 15th National Conference on AI. (1998) 714-720
3. Breese, J. S., Kadie, C.: Empirical Analysis of Predictive Algorithms for Collaborative Filtering. Proceedings of the 14th Conference on Uncertainty in AI. (1998) 43-52
4. Herlocker, J. L., Konstan, J. A., Terveen, L. G., Riedl, J. T.: Evaluating Collaborative Filtering Recommender Systems. ACM Transactions on Information Systems (TOIS) archive, Vol. 22, No. 1. (2004) 5-53
5. Jung, K. Y., Lee, J. H.: User Preference Mining through Hybrid Collaborative Filtering and Content-based Filtering in Recommendation System. IEICE Transaction on Information and Systems, Vol.E87-D, No.12. (2004) 2781-2790
6. Kim, T. H., Yang, S. B.: An Improved Neighbor Selection Algorithm in Collaborative Filtering. IEICE Trans. on Inf. and Systems, Vol. E88-D, No. 5. (2005) 1072-1076
7. Ko, S. J., Lee, J. H.: Feature Selection using Association Word Mining for Classification. Proceedings of the International Conference on DEXA. (2001) 211-220
8. Melville, P., Mooney, R. J., Nagarajan, R.: Content-Boosted Collaborative Filtering. Proceedings of the SIGIR-2001 Workshop on Recommender Systems. (2001)
9. McJones, P., EachMovie collaborative filtering dataset,
10. URL: <http://www.research.digital.com/SRC/eachmovie>. (1997)
11. Michael, T., *Maching Learning*, McGrae-Hill. (1997) 154-200
12. Miyahara, K., Pazzani, M. J.: Collaborative Filtering with the Simple Bayesian Classifier. Proceedings of the 6th Pacific Rim Int. Conference on Artificial Intelligence. (2000) 679-689
13. Pazzani, M. J.: A Framework for Collaborative, Content-based and Demographic Filtering. Artificial Intelligence Review. (1999) 393-408
14. Soboroff, I., Nicholas, C. K.: Related, but not Relevant: Content-Based Collaborative Filtering in TREC-8. Information Retrieval, Vol. 5, No. 2-3. (2002) 189-208
15. Lee, W. S.: Collaborative Learning for Recommender Systems. Proceedings of the 18th International Conference on Machine Learning. (1997) 314-321
16. Wang, J., de Vries, A. P., Reinders, M. J.T.: A User-Item Relevance Model for Log-based Collaborative Filtering. Proceedings of the European Conference on Information Retrieval. (2006) 37-48

# A Stochastic Non-CNF SAT Solver

Rafiq Muhammad and Peter J. Stuckey

NICTA Victoria Laboratory,  
Department of Computer Science and Software Engineering,  
The University of Melbourne, Victoria 3010, Australia  
`{rmmu, pjs}@cs.mu.oz.au`

**Abstract.** Stochastic local search techniques have been successful in solving propositional satisfiability (SAT) problems encoded in conjunctive normal form (CNF). Recently complete solvers have shown that there are advantages to tackling propositional satisfiability problems in a more expressive natural representation, since the conversion to CNF can lose problem structure and introduce significantly more variables to encode the problem. In this work we develop a non-CNF SAT solver based on stochastic local search techniques. Crucially the system must be able to represent how true a proposition is and how false it is, as opposed to the usual stochastic methods which represent simply truth or degree of falsity (penalty). Our preliminary experiments show that on certain benchmarks the non-CNF local search solver can outperform highly optimized CNF local search solvers as well as existing CNF and non-CNF complete solvers.

## 1 Introduction

Modern propositional satisfiability (SAT) solvers are usually designed to solve SAT formula encoded in conjunctive normal form (CNF). CNF solvers can be disadvantageous for problems which are more naturally encoded as arbitrary propositional formula. The conversion to CNF form may increase the size of the formula exponentially, or significantly reduce the strength of the formulation. The translation may introduce many new variables which increases the size of the raw valuation space through which the solver must search.

Recently, interest has arisen in designing non-clausal satisfiability algorithms. In 1993, Armando and Giunchiglia [1] introduced PTAUT, which was a generalization of Davis-Putnam-Logemann-Loveland (DPLL) algorithm [2], a complete CNF SAT algorithm, to work on non-CNF formula. The primary drawback of their implementation was the performance which was far below current implementations of CNF solvers based on the DPLL algorithm, because it failed to exploit clever techniques and efficient data structures used in manipulating CNF formulae, e.g.. [3,4]. This work was improved by Giunchiglia and Sebastiani [5] who devised a non-CNF approach able to exploit all the present and future sophisticated technology of DPLL implementations. They converted the input formula to CNF, but ensured that the DPLL procedure would not backtrack on

new variables introduced in CNF conversion hence maintaining the search space of the original non-CNF formulae.

More recently, Thiffault et al. [6] generalized the DPLL to work directly on non-CNF formulae. They argued that conversion to CNF is unnecessary and results in the drawback of losing structural information and increase in the search space. They implemented a complete non-CNF DPLL like solver that is capable of achieving an efficiency very similar to that of modern highly optimized CNF solvers using techniques very similar to these solvers. They exploited the additional structural information presented in non-CNF propositional formula to achieve significant gains in solving power, to the point where on various benchmarks their non-clausal solver outperforms the CNF solver it was based on.

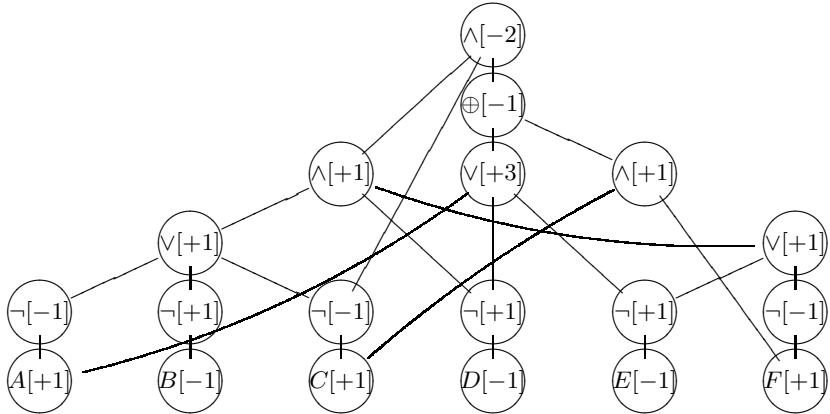
Local search based SAT solvers can typically solve problems an order magnitude larger than those that can be handled by a complete solver. For CNF problems, GSAT [7] demonstrated that an incomplete SAT solver can solve many hard problems much more efficiently than traditional complete solvers based on DPLL. The success of GSAT gave birth to several variants based on stochastic local search techniques, see [8]. Several authors have attempted to generalize these techniques to non-CNF formula. Sebastiani [9] suggested how to modify GSAT to be applied to non-CNF formula but the idea was not implemented. Kautz et al. [10] introduced DAGSat, an improvement of WalkSAT in term of handling of variable with dependencies. DAGSat still required formula in CNF, but allowed handling of non-CNF formula without an increase in size. Later, Stachniak [11] introduced polWSAT an evolution of WalkSAT [12] to handle non-clausal formula, although it was restricted to formula using  $\wedge$ ,  $\vee$  and  $\neg$  in negation normal form (where all negations appear on literals).

In this paper, we present a pure non-clausal solver based on stochastic local search techniques. It works directly on arbitrary non-clausal formulae including psuedo-Boolean constructs and other extended Boolean operators. The contributions of this paper are:

- We present a new way of expressing the scoring function for evaluating a valuation. We are not only able to define the “falsity” (or penalty) of a non solution, but also the “truthfulness” of an assignment, that is, how true it makes the result. This is required since we need to negate “truthfulness” to obtain “falsity”. This complex scoring is necessary since we don’t restrict ourselves to negation normal forms. It provides more accurate heuristic information to assist the local search process.
- We tested our solver on non-CNF benchmarks and demonstrate experimentally that on some benchmarks, our incomplete non-clausal solver can outperform current incomplete clausal solvers as well as current complete non-clausal solvers.

## 2 A Stochastic Non-clausal Solver

Let  $\phi$  be a propositional formula, our aim is to decide if  $\phi$  is satisfiable. The propositional formula is represented as a Boolean DAG where each internal node



**Fig. 1.** The propositional formula  $\phi = (((\neg A \vee \neg B \vee \neg C) \wedge \neg D \wedge (\neg E \vee \neg F)) \wedge \neg C \wedge ((\neg D \vee A \vee \neg E) \oplus (C \wedge F)))$  with scores for the valuation  $\{A, \neg B, C, \neg D, \neg E, F\}$  bracketed

represents an (extended) Boolean operator and its children are sub-trees representing its operands. For example, Figure 1 represents the Boolean formula  $\phi$  as a DAG.

The aim of our algorithm is to find values for the variables such that will result in the satisfiability of the propositional formula. For each assignment, a score is computed for each node in the DAG to represent the state of satisfiability of the corresponding propositional (sub-)formula.

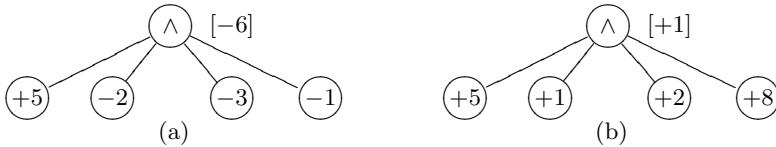
## 2.1 Score

The notion of score plays a key role in determine the “distance” from the current valuation to a satisfying one. We allow our scoring function to take positive and negative values in order to enable our algorithm to express the “truthfulness” as well as the “falsity” of the state of the truth assignment. All Boolean variables (leaf nodes) have a score of either 1 (*true*) or (-1) *false*. For each internal node we calculate the score as defined below. The calculations are such that: if the Boolean DAG were a tree then a score of  $+n$  means that  $n$  variables must change their value for the corresponding (sub-)formula to become *false*, while if the score is  $-n$  then  $n$  variables must change their value for the corresponding (sub-)formula to become *true*. Of course in practice the DAG is never a tree.

We will determine the score  $s_0$  for a node  $\phi_0$ , in terms of the scores  $s_1, \dots, s_n$  of its  $n$  children  $\phi_1, \dots, \phi_n$ .

$$\text{NOT}(\neg) \phi_0 = \neg \phi_1$$

Not is the negation of the truth value of a variable, therefore the score for a negated node,  $s_0$ , is the negation of the score of its child:  $s_0 = -(s_1)$ .



**Fig. 2.** AND score (a) where the node is *false* and (b) where it is *true*

*AND*( $\wedge$ )  $\phi_0 = \phi_1 \wedge \cdots \wedge \phi_n$

An AND node (with  $n$  children) can only be *true* if all its children are *true*, therefore, if the node is currently *false*, the score of the node is the sum of negative children (Fig. 2(a)) and if the node is *true*, the score of the node is the score of the child with minimum value (Fig. 2(b)).

$$s_0 = \begin{cases} \sum\{s_i | 1 \leq i \leq n, s_i < 0\} & \exists 1 \leq i \leq n \ s_i < 0 \\ \min\{s_i | 1 \leq i \leq n\} & \text{otherwise} \end{cases} \quad \begin{matrix} (\text{false}) \\ (\text{true}) \end{matrix}$$

For example, in order to change the node in Fig. 2(a) to be *true*, we have to change the nodes with  $[-2]$ ,  $[-3]$  and  $[-1]$  to have a positive score, hence the score of the parent is  $[-6]$ . On the other hand, the score of the node in Fig. 2(b) is  $[+1]$  because the minimum change to turn the node to *false*, is by turning the truth value of the child with score  $[+1]$  to *false*.

*OR*( $\vee$ )  $\phi_0 = \phi_1 \vee \dots \vee \phi_n$

An OR node becomes *true* when any one of its children is *true*, and *false* if all of its children are *false*, therefore, if the node is *false*, the score of the node is the score of the child with maximum value and if the node is *true*, the score of the node is the sum of positive children.

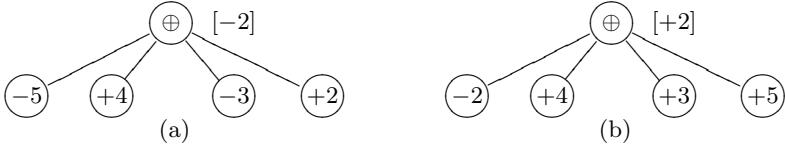
$$s_0 = \begin{cases} \max\{s_i | 1 \leq i \leq n\} & \forall 1 \leq i \leq n \ s_i < 0 \quad (\text{false}) \\ \sum\{s_i | 1 \leq i \leq n, s_i > 0\} & \text{otherwise} \quad (\text{true}) \end{cases}$$

*XOR*( $\oplus$ )  $\phi_0 = \phi_1 \oplus \cdots \oplus \phi_n$

An XOR  $\phi_0$  is *true* if the parity of the total number of *true* children is odd. Simply flipping the truth value of any one of the children of  $\phi_0$  will change the parity from odd to even and vice versa. Hence the XOR node that is *false* has a score equal to the negative of the smallest absolute value of any child score, while if its true, the score is the smallest absolute value of the child score.

$$s_0 = \begin{cases} -\min\{|s_i| \mid 1 \leq i \leq n \mid |s_i| \leq n, s_i > 0\} \bmod 2 = 0 & (\text{false}) \\ +\min\{|s_i| \mid 1 \leq i \leq n \mid \text{otherwise}\} & (\text{true}) \end{cases}$$

For example, the XOR node in Fig. 3(a) is *false* because 2 of the children are *true* (an even number). In order to make the node to be *true*, we have to flip the truth value of one of the children. The least change required is to change the [+2] child to be *false*, hence the score is [-2]. Fig. 3(b) shows the score of an XOR node that is *true*.



**Fig. 3.** XOR score (a) where the node is *false* and (b) where it is *true*

*IFF*( $\Leftrightarrow$ )  $\phi_0 = \phi_1 \Leftrightarrow \dots \Leftrightarrow \phi_n$

Equivalent (IFF) is the opposite of XOR. An IFF node  $\phi_0$  is *true* if the parity of the total number of *true* children is even. The resulting score function is then:

$$s_0 = \begin{cases} -\min\{|s_i| \mid 1 \leq i \leq n\} & |\{s_i \mid 1 \leq i \leq n, s_i > 0\}| \bmod 2 = 1 \\ +\min\{|s_i| \mid 1 \leq i \leq n\} & \text{otherwise} \end{cases} \quad (\text{false}) \quad (\text{true})$$

*IMPLIES*( $\Rightarrow$ )  $\phi_0 = \phi_1 \Rightarrow \phi_2$

Implication is a simple binary operator. An implication formula is *false* if and only if the left operand is *true* and right operand is *false*. If the implication node is *false*, then flipping either operand will turn the implication node to be *true*. If the implication node is *true*, the cost of turning the node to *false* is the sum of the cost of turning the left operand to *true* and the cost of turning the right operand to *false*.

$$s_0 = \begin{cases} \max\{-s_1, +s_2\} & s_1 > 0 \wedge s_2 < 0 \\ \max\{s_2, 0\} - \min\{s_1, 0\} & \text{otherwise} \end{cases} \quad (\text{false}) \quad (\text{true})$$

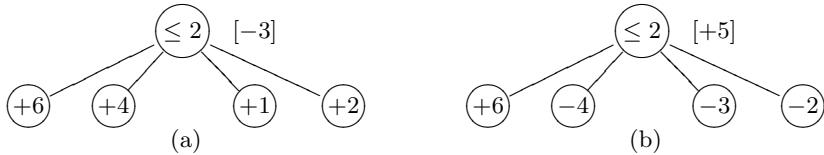
*ATMOST*( $\leq k$ )  $\phi_0 = (\phi_1 + \dots + \phi_n \leq k)$

An ATMOST formula  $\phi_0$  with parameter  $k$  ( $\leq k$ ) is true if at most  $k$  of its  $n$  inputs are *true*. If the node is *false*, and currently  $k' > k$  children are *true*, we need to make *false*  $k' - k$  more variables, so the score is negative of the minimum sum required to do this. Similarly, if the node is *true*, and currently  $k' \leq k$  children are *true*, we need to turn  $k - k' + 1$  children *true* to make it *false*.

$$s_0 = \begin{cases} -\sum \min_{k'-k} T & T = \{s_i \mid 1 \leq i \leq n, s_i > 0\}, \\ & k' = |T|, k' > k \\ -\sum \max_{k-k'+1} \{s_i \mid 1 \leq i \leq n\} - T & T = \{s_i \mid 1 \leq i \leq n, s_i > 0\}, \\ & k' = |T|, k' \leq k \end{cases} \quad (\text{false}) \quad (\text{true})$$

where  $\min_l S$  returns the minimal  $l$  elements of  $S$ , that is a subset  $M$  of  $S$  of cardinality  $l$  such that  $\forall x \in S - M. \forall y \in M. x \geq y$ . Similarly  $\max_l$  returns the maximal  $l$  elements of  $S$ .

For example, in Fig. 4(a), to change the node to be *true*, we need to change two children to be *false*. The cheapest way of doing this is with the  $[+1]$  and  $[+2]$  children.  $T = \{+6, +4, +1, +2\}$ ,  $k' = 4$  and  $\min_2 T = \{+1, +2\}$ . Hence the resulting score is  $[-3]$ . In Fig. 4(b), to change the node to be *false*, we would need to make at least two more children *true*.  $T = \{+6\}$ ,  $k' = 1$  and  $\max_2 \{-4, -3, -2\} = \{-2, -3\}$ . Hence the score is  $[+5]$ .



**Fig. 4.** ATMOST 2 score (a) where the node is *false* and (b) where it is *true*

*ATLEAST*( $\geq k$ )  $\phi_0 = (\phi_1 + \cdots + \phi_n \geq k)$

The ATLEAST operator is similar to the ATMOST operator. The resulting scoring function is thus a mirrored form.

$$s_0 = \begin{cases} \sum \max_{k-k'} \{s_i \mid 1 \leq i \leq n\} - T & T = \{s_i \mid 1 \leq i \leq n, s_i > 0\}, \\ & k' = |T|, k' < k \\ \sum \min_{k'-k+1} T & T = \{s_i \mid 1 \leq i \leq n, s_i > 0\}, \\ & k' = |T|, k' \geq k \end{cases} \quad \begin{matrix} (false) \\ (true) \end{matrix}$$

$$COUNT(\equiv k) \phi_0 = (\phi_1 + \cdots + \phi_n = k)$$

The COUNT operator is simply a combination of the ATMOST and ATLEAST operators, but it ends up with a slightly different form of scoring function, after simplifying. If the node is *false* to make it *true* we need to flip truth values to bring the count either up or down to  $k$ . To make it *false* from *true* we only have to flip one truth value.

$$s_0 = \begin{cases} \sum \max_{k-k'} \{s_i \mid 1 \leq i \leq n\} - T & T = \{s_i \mid 1 \leq i \leq n, s_i > 0\}, \\ & k' = |T|, k' < k \\ - \sum \min_{k'-k} T & T = \{s_i \mid 1 \leq i \leq n, s_i > 0\}, \\ & k' = |T|, k' > k \\ \min \{|s_i| \mid 1 \leq i \leq n\} & \text{otherwise} \end{cases} \quad \begin{matrix} (false) \\ (false) \\ (true) \end{matrix}$$

*Other Operators.* There are also other operators defined in DIMACS non-clausal style [13] such as NAND, NOR, and XNOR. These operators are the negation of operators we have already defined. We can generate their scoring functions by simply negating the corresponding versions: NAND =  $\neg$  AND, NOR =  $\neg$  OR, and XNOR =  $\neg$  IFF.

## 2.2 Searching for a Solution

All local search based solvers work in essentially the same way. A candidate valuation for the variables is determined, and then the search looks for a “neighboring” valuation which is better. In SAT solvers the usual definition of neighboring valuations, is those obtained by flipping the value of one Boolean variable. A valuation is considered better if it satisfies more clauses, or satisfies a greater sum of weighted clauses. In the non-clausal solver the situation is the same. We have a current valuation and its score for the overall formula. We look at neighboring valuations which improve the score, attempting to drive the score to be positive (and hence satisfying the root propositional formula).

We consider a search strategy similar to the approach used by WalkSAT [12] and its extension to non-clausal solvers [11]. WalkSAT works by first selecting an unsatisfied clause, and then selecting a variable in that clause for flipping. A CNF formula is simply a root AND node above many OR nodes relating to literals (variables or NOT variables sub trees). How do we generalize the WalkSAT approach to an arbitrary formula? We consider it in this way. The WalkSAT approach begins at the root and considers which child nodes could improve the root score (a *false* child has a score of  $-1$ , and changing it will improve the score of the AND parent) and randomly selects one. It then does the same for the OR node (although in this case all children can improve the score). Hence the generalization is clear. At node  $n$  we randomly select a child node, for which flipping the truth value would move the nodes score towards the opposite sign it has now. We continue this process until we reach a variable node. This is the variable we then flip in the search process. Note that the children that can improve a node are implicitly readable from the definition of the scoring function. We flip the variables truth value and with probability  $p$  accept the change if it is downhill (improves the score at the root), and with probability  $1 - p$  accept the move whether it is downhill, uphill or flat.

For our experiments we found a value of  $p = 0.9$  was best and use this value throughout our experiments.

Examining the DAG shown in Figure 1. At the root the score of  $[-2]$  is the sum of the negative children of  $[-1]$ , so we randomly select one of them, say the XOR node. Its score is as a result of both children being positive so we randomly select one, say the OR node on the right. Its score is positive because of all three children so we randomly choose one say  $\neg E$ . This is positive because of its child  $E$ . So this is our variable for flipping.

### 3 Preliminary Experimental Results

We implemented our stochastic non-clausal SAT solver in C++ using the technology of one way constraints or invariants [14].

We compare our stochastic non-CNF solver (SNCNFS) with WalkSAT [12] (an incomplete CNF solver), MiniSat [15] (a complete CNF solver), and NoClause [6] (a complete non-CNF solver) on several test suites. Since our solver is incomplete, we only considered formula that is satisfiable. For the incomplete solvers (WalkSAT and SNCNFS) we used a maximum flips of 250,000 and repeated each test 100 times. The initial starting valuations were uniformly randomly generated. We used the Random strategy of WalkSAT most similar to the search strategy defined above for SNCNFS. Since flips in WalkSAT are substantially faster than those in SNCNFS, whenever SNCNFS solved a larger percentage of problems we increased the maximum flips for WalkSAT to 10,000,000. The benchmarks where this occurred are marked as  $\dagger$ . For the complete solvers, we aborted if no solution is found after running for two hours. In the tables, where solutions are found, we give the averages over the successful runs only.

**Table 1.** Comparative results for solvers on hard random formula

R	# Clause	CNF				Non-CNF					
		WALKSAT			MiniSat	SNCNFS			NoClause		
		% Succ	Mean Flips	Mean Time(s)	Mean Time(s)	# Nodes	% Succ	Mean Flips	Mean Time(s)	Mean Time	
1.6	8327	100	153	0.0009	0.0600	8743	100	434	0.2020	2.994	
1.8	9333	100	219	0.0014	0.4100	9801	100	2448	1.0364	22.121	
2.0	10388	100	700	0.0040	29.0000	10837	100	3256	1.7596	71.352	
2.2	11494	100	1975	0.0116	—	11791	100	5282	3.2681	—	
2.4	12508	100	9519	0.0406	—	12749	100	19596	13.4205	—	
2.6	13491	100	56517	0.3773	—	13907	92	64783	51.0984	—	
2.8†	14545	7	4543650	70.3112	—	14942	38	42775	37.3086	—	
3.0†	15577	0	—	—	—	15904	0	—	—	—	

— Max flips or max time exceeded.

### 3.1 Hard Random Non-CNF Formulas

Randomly generated formulae provide a good test bed for evaluating the performance of satisfiability algorithms. We used the random formula generator of Navarro and Voronkov [16] which is based on a fixed shape model. The generator is capable of generating formulae with different level of difficulty in non-CNF as well as CNF format hence making it an ideal choice to test our stochastic non-CNF solver. The difficulty and the satisfiability of the output formulae is controlled by  $r$  the *ratio of formulae-to-variables*. Small  $r$  produces formulae that are satisfiable and under-constrained, while large  $r$  results in formulae that are unsatisfiable and over-constrained. The hardest problems appear in the transition region where there are just enough constraints to make the problem potentially unsatisfiable, but not too many to make it easy for a solver to determine. We generated 200-variable random formulae of shape  $\langle 3, 3, 2 \rangle$  with  $r$  increasing by 0.2 within the range from 1.6 to 3.0.

The comparative results are shown in Table 1. For these problems the size of the non-CNF formula is not substantially small than the CNF form, so there is no advantage to the non-CNF solver in size. Clearly the the execution time of SNCNFS are much longer than WalkSAT for the same number of flips, illustrating the highly optimized implementation of WalkSAT. But SNCNFS is capable of solving harder problems than WalkSAT, illustrating there is an advantage in treating the formula in the non-CNF form. For this class of problems the complete solvers are unable to tackle the more difficult cases.

### 3.2 Minimal Disagreement Parity Problem

The Minimal Disagreement Parity (MDP) Problem [17] is a well-known class of hard satisfiability problem. The advantage of a non-CNF encoding for these problems is that we can maintain the encoding of *XOR* and use the *AT MOST* gate to determine the correctness of the parity function, hence reducing the problem size significantly. We compare the non-CNF encoding versus the standard CNF

**Table 2.** Comparison of results on the CNF and Non-CNF encoding of MDP problem

Problem	CNF						Non-CNF					
	WalkSAT			Minisat			SNCNFS					
	# Vars	# Cls	% Succ	Mean Flips	Mean Time(s)	Time	# Vars	# Nodes	% Succ	Mean Flips	Mean Time(s)	
par8-1†	350	1149	2	298872	0.149	0.02	8	441	100	3227	0.1782	
par16-1†	1015	3310	0	—	—	0.1	16	1649	90	57471	7.0971	
par32-1†	3176	10277	0	—	—	—	32	6369	—	—	—	
par8-1-c	64	254	100	8516	0.0054	0.01	8	36	100	3123	0.0572	
par16-1-c†	317	1264	0	—	—	0.06	16	66	100	32284	1.1499	
par32-1-c†	1315	5254	0	—	—	—	32	131	—	—	—	

The trailing ‘c’ in the problem is the compressed version of the instance.  
— Max flips or max time exceeded.

encoding. Table 2 shows the results of SNCNFS compared to WalkSAT and Minisat (we cannot apply NoClause since it does not support the ATMOST gate). Although we are unable to beat Minisat, SNCNFS can solve 16 bit instances which are beyond the capability of WalkSAT<sup>1</sup>.

## 4 Conclusions and Future Work

We have introduced an incomplete non-clausal solver based on stochastic local search. This is the first work we are aware of which uses both negative and positive scores for evaluating the degree of “truthfulness” or “falsity” of a propositional formula. Our experiments demonstrate that on certain benchmarks, our stochastic local search non-clausal solver can out perform existing incomplete CNF solver as well as complete CNF and non-CNF solvers. The results of our preliminary experiments are very promising.

It would be interesting to find more complex non-clausal benchmarks to experiment on, almost all SAT benchmarks are presently in CNF. The advantage of a non-clausal solver should be more evident on large difficult benchmarks that appear in non-clausal form. There remains a great deal of scope to explore different strategies for selecting a neighborhood move. We could take into account the magnitude of the children scores in weighting the random choice of child to take. We could also extend the approach to generate a candidate set of variables to flip by keeping more than one child that can change the truth value, and then picking the variable whose flipping leads to the best overall score for the root node. We could add penalties to nodes, and learn penalties when we find ourselves in local minima (as in DLM [19]). Finally, it would be worthwhile reimplementing the algorithm in Comet [20] which provides efficient and built in evaluation of invariants.

<sup>1</sup> Systematic methods are known to work well on this class of problem. WalkSAT using the Novelty strategy can solve the compressed version of 16 bits instance, but not the uncompressed version [18].

## References

1. Armando, A., Giunchiglia, E.: Embedding complex decision procedures inside an interactive theorem prover. *Ann. Math. Artif. Intell.* **8**(3-4) (1993) 475–502
2. Davis, M., Logemann, G., Loveland, D.: A machine program for theorem-proving. *Commun. ACM* **5**(7) (1962) 394–397
3. Crawford, J., Auton, L.: Experimental results on the crossover point in random 3-SAT. *Artif. Intell.* **81**(1-2) (1996) 31–57
4. Zhang, H., Stickel, M.: Implementing the Davis-Putnam method. *J. Autom. Reasoning* **24**(1/2) (2000) 277–296
5. Giunchiglia, E., Sebastiani, R.: Applying the Davis-Putnam procedure to non-clausal formulas. In: AI\*IA. (1999) 84–94
6. Thiffault, C., Bacchus, F., Walsh, T.: Solving non-clausal formulas with DPLL search. In: CP. (2004) 663–678
7. Selman, B., Levesque, H., Mitchell, D.: A new method for solving hard satisfiability problems. In: AAAI. (1992) 440–446
8. McAllester, D.A., Selman, B., Kautz, H.A.: Evidence for invariants in local search. In: AAAI/IAAI. (1997) 321–326
9. Sebastiani, R.: Applying GSAT to non-clausal formulas (research note). *J. Artif. Intell. Res. (JAIR)* **1** (1994) 309–314
10. Kautz, H., Selman, B., McAllester, D.: Exploiting variable dependency in local search. In Abstracts of the Poster Session of IJCAI-97 (1997)
11. Stachniak, Z.: Going non-clausal. In 5th International Symposium on Theory and Applications of Satisfiability Testing (2002)
12. Selman, B., Kautz, H., Cohen, B.: Noise strategies for improving local search. In: AAAI. (1994) 337–343
13. Bacchus, F., Walsh, T.: A non-CNF DIMACS style. Available from <http://www.ssatcompetition.org/2005/> (2005)
14. Van Hentenryck, P., Michel, L.: Localizer. *Constraints* **5** (2000) 41–82
15. Eén, N., Sörensson, N.: An extensible SAT-solver. In: SAT. (2003) 502–518
16. Navarro, J.A., Voronkov, A.: Generation of hard non-clausal random satisfiability problems. In: AAAI. (2005) 436–442
17. Crawford, J., Kearns, M., Schapire, R.: The minimal disagreement parity problem as a hard satisfiability problem. unpublished manuscript (1995)
18. Hoos, H.H., Stützle, T.: Systematic vs. local search for sat. In: KI. (1999) 289–293
19. Wah, B.W., Shang, Y.: A discrete lagrangian-based global-search method for solving satisfiability problems. *J. of Global Optimization* **12** (1998)
20. Van Hentenryck, P., Michel, L.: Constraint-Based Local Search. MIT Press (2005)

# Reasoning About Hybrid Probabilistic Knowledge Bases

Kedian Mu<sup>1</sup>, Zuoquan Lin<sup>1</sup>, Zhi Jin<sup>2</sup>, and Ruqian Lu<sup>2</sup>

<sup>1</sup> School of Mathematical Sciences

Peking University, Beijing 100871, P.R. China

<sup>2</sup> Academy of Mathematics and System Sciences

Chinese Academy of Sciences, Beijing 100080, P.R. China

**Abstract.** Most techniques for probabilistic reasoning focus on reasoning about conditional probability constraints. However, human experts are accustomed to representing uncertain knowledge in the form of expectation rather than probability distribution directly in many cases. It is necessary to provide a logic for encoding hybrid probabilistic knowledge bases that contain expectation knowledge as well as the purely probabilistic knowledge in the form of conditional probability. This paper constructs a nonmonotonic logic for reasoning about hybrid probabilistic knowledge bases. We extend the propositional logic for reasoning about expectation to encoding hybrid probabilistic knowledge by introducing the conditional expectation constraint formula. Then we provide an approach to nonmonotonic reasoning about hybrid probabilistic knowledge bases. Finally, we compare this logic with related works.

## 1 Introduction

Probabilistic reasoning is a premier form of uncertain reasoning in Artificial Intelligence (abbr. AI). The semantics of conditional probability constructed a bridge from probability theory to probabilistic reasoning. Informally, given some knowledge  $k$ , the degree of reasoner's belief in a hypothesis  $h$  is represented by the conditional probability  $P(h|k)$  in probabilistic reasoning. Furthermore, given another knowledge  $l$ , the process of belief updating from  $P(h|k)$  to  $P(h|k, l)$  obeys the probability axioms.

During the recent decades, reasoning about interval restrictions for conditional probabilities, called conditional constraints, has been the subject of extensive research efforts. Generally, a conditional constraint is of the form  $(\psi|\phi)[l, u]$ , where  $\psi$  and  $\phi$  are events, and  $[l, u]$  is a subinterval of the unit interval  $[0,1]$ . It signifies that the conditional probability of  $\psi$  given  $\phi$ ,  $P(\psi|\phi)$ , lies in  $[l, u]$ . Current techniques for reasoning about collections of conditional constraints, termed knowledge bases have been surveyed by Lukasiewicz [1]. These knowledge bases focused on both *logical knowledge* and *probabilistic knowledge*.

- At the beginning of reasoning about conditional constraints, the knowledge base only focuses on *strict knowledge*. For example, a simple knowledge base

- may encode the *strict logical knowledge* “all eagles are birds” and “all birds have feathers” as well as the *strict purely probabilistic knowledge* “birds fly with a probability of at least 0.95 (i.e.,  $0.95 \leq P(fly|bird) \leq 1$ )” [2,3].
- Further, the *default knowledge* is also taken into account in conditional knowledge bases. There are several works in the literature on probabilistic foundations for default reasoning from conditional knowledge bases. Lukasiewicz [4] presents weak nonmonotonic probabilistic logics, which focus on knowledge bases that consist of *strict logical knowledge*, *default logical knowledge*, and *strict purely probabilistic knowledge*. For example, consider the knowledge base that encodes the *strict logical knowledge* “all penguins are birds”, the *default logical knowledge* “generally, birds have legs” and “generally, birds fly”, and the *purely probabilistic knowledge* “penguins fly with a probability of at most 0.05 (i.e.,  $P(fly|penguin) \leq 0.05$ )”. In weak nonmonotonic probabilistic logic, the notion of probabilistic lexicographic entailment can produce the desired conclusions such as “generally, penguins have legs” and “penguins fly with a probability of at most 0.05”.
  - On the contrary, the strong nonmonotonic probabilistic logics presented in [5] are quite different from weak nonmonotonic logics in that they allow for handling *default purely probabilistic knowledge*, rather than (*strict*) *purely probabilistic knowledge*, in addition to *strict logical knowledge* and *default logical knowledge*. For example, they allow for expressing sentences “generally, birds fly with a probability of at least 0.95” rather than “birds fly with a probability of at least 0.95”. Intuitively, the former means that being able to fly with a probability at least 0.95 should apply to all birds and all subclasses of birds, as long as this is consistent, while the latter says that being able to fly with a probability of at least 0.95 should only apply to all birds.
  - Finally, nonmonotonic probabilistic logics under variable-strength inheritance with overriding are a general approach to nonmonotonic probabilistic reasoning [1], which subsumes strong nonmonotonic probabilistic reasoning and weak nonmonotonic probabilistic reasoning as special cases. Informally, these formalisms also allow for handling *strict logical knowledge*, *default logical knowledge*, and *default purely probabilistic knowledge*, but the inheritance of *strict purely probabilistic knowledge* is controlled by a strength  $\lambda \in [0, 1]$ .

In all the knowledge bases mentioned above, the *purely probabilistic knowledge* is of the form

(Generally), if  $\phi$  then  $\psi$  with a *probability* between  $l$  and  $u$ ,

since *probability distribution function* is sufficient to represent uncertainty according to the viewpoint of probability theory. However, human experts are not accustomed to representing their uncertain knowledge directly in *probability distribution function* in many cases. In particular, for a random variable with

numerical configuration, the human expert always adopts *its digital features* rather than *probability distribution function* to describe the random variable in routine life. For example, a physician may be accustomed to saying that “generally, the normal value of body temperature is 37 °c” rather than “generally, the normal value of body temperature is 37 °c with the probability of 0.9, 37.2 °c with the probability of 0.1, and 36.8 °c with the probability of 0.1, respectively”.

Expectation is viewed as one of the important digital features to describe a random variable. For a random variable  $X$ , let  $V(X)$  denote its configuration, then its expectation (or its expected value),  $E_P(X)$ , is defined as follows<sup>1</sup>:

$$E_P(X) = \sum_{x \in V(X)} xP(X = x).$$

And given  $Y=y$ , the conditional expectation  $E_P(X|Y=y)$  is defined as follows:

$$E_P(X|Y=y) = \sum_{x \in V(X)} xP(X = x|Y = y).$$

In semantics, the expectation of random variable is the single value that best describes the random variable in the sense of average. In contrast to the *purely probabilistic knowledge* mentioned in non-monotonic probabilistic logics, the *expectation knowledge* is of the form

(Generally), if  $\phi$ , then  $E_P(X|\phi)$  is between  $c_1$  and  $c_2$ ,

where  $\phi$  and  $X$  are an event and a numerical random variable, respectively, and  $c_1, c_2 \in \mathbb{R}$ .

The gist of this paper is to extend the knowledge bases mentioned in non-monotonic probabilistic logics to the *hybrid probabilistic bases*, which may encode *expectation knowledge* as well as *purely probabilistic knowledge in the form of conditional probability*. However, Halpern [6] has presented a propositional logic for reasoning about expectation. As far as the expressive power is concerned, the propositional logic is more expressive than the corresponding logic for reasoning about likelihood [2] in the case of sets of probability measures, but equi-expressive in the case of probability, belief and possibility [6].

In this paper, we extend the propositional logic for reasoning about expectation to a nonmonotonic probabilistic logic that can encode hybrid probabilistic knowledge, and discuss an approach to nonmonotonic reasoning about hybrid knowledge bases. The paper is organized as follows: section 2 gives a brief introduction to the propositional logic for reasoning about expectation proposed by Halpern [6]; Section 3 extends the logic for encoding hybrid probabilistic knowledge by introducing the conditional expectation constraint formula; Section 4 introduces an approach to nonmonotonic reasoning about hybrid knowledge bases;

---

<sup>1</sup> For simplicity, all random variables mentioned in this paper are finite discrete variables.

Section 5 compares the extended logic to the related works and discusses the expressive power of this logic. Section 6 concludes the main results of the paper.

## 2 Preliminaries: Reasoning About Expectation

We first give a brief introduction to the propositional logic for reasoning about expectation that proposed by Halpern [6]. It is similar to the **FHM** logic [2] in syntax.

We assume a finite set of primitive propositions  $\Phi_0$ . *True* and *false* are two special propositional constants of  $\Phi_0$ , denoted by  $\top$  and  $\perp$ , respectively. The set  $\Phi$  of propositional formulas is the closure of  $\Phi_0$  under  $\wedge$  and  $\neg$ . In propositional logic for reasoning about expectation, a linear propositional gamble is a linear combination of propositional formulas, of the form  $\gamma = a_1\varphi_1 + \dots + a_n\varphi_n$ , where  $a_1, \dots, a_n \in \mathbb{R}$ ,  $\varphi_1, \dots, \varphi_n \in \Phi$ . A basic expectation formula has the form  $b_1e(\gamma_1) + \dots + b_ke(\gamma_k) \geq c$ , where  $k \geq 1$ ,  $b_1, \dots, b_n, c \in \mathbb{R}$ , and  $\gamma_1, \dots, \gamma_k$  are linear propositional gambles. An expectation formula is a Boolean combination of basic expectation formulas. The expectation formula involves  $\leq, =, <$ , and  $>$  can be easily defined from  $\geq$  and negation. For example,  $b_1e(\gamma_1) + \dots + b_ke(\gamma_k) \leq c$  is an abbreviation for  $-b_1e(\gamma_1) - \dots - b_ke(\gamma_k) \geq -c$ . Furthermore, we may use obvious abbreviation where needed, such as  $e(\gamma_1) - e(\gamma_1) \geq a$  for  $e(\gamma_1) + (-1)e(\gamma_1) \geq a$ . Let  $L^E$  be the language consisting of expectation formulas. Halpern [6] mentioned that the semantics of  $L^E$  depends on how  $e$  is interpreted. For simplicity, we present a universal semantics for  $L^E$  in the form of uncertainty structure.

Let a tuple  $M = (W, \tilde{\mu}, \pi)$ <sup>2</sup> be an uncertainty structure, where  $\tilde{\mu}$  is a kind of uncertainty measures over the set of possible worlds  $W$  and  $\pi$  is an interpretation, which associates with each world in  $W$  a truth assignment on the primitive propositions in  $\Phi_0$ . That is,  $\pi(w)(p) \in \{\text{true}, \text{false}\}$  for  $w \in W$  and  $p \in \Phi_0$ . Extend  $\pi(w)$  to a truth assignment on all propositional formulas in the standard way, and associate with each propositional formula the set  $[\lvert\varphi\rvert]_M = \{w \in W : \pi(w)(\varphi) = \text{true}\}$ . For every  $\varphi \in \Phi$ , we define a random indictor function  $X_{[\lvert\varphi\rvert]_M}$  as follows:  $X_{[\lvert\varphi\rvert]_M}(w) = 1$ , if  $\pi(w)(\varphi) = \text{true}$ ,  $X_{[\lvert\varphi\rvert]_M}(w) = 0$ , if  $\pi(w)(\varphi) = \text{false}$ . Thus,  $\tilde{\mu}(X_{[\lvert\varphi\rvert]_M}(w) = 1) = \tilde{\mu}([\lvert\varphi\rvert]_M)$ . By the indicator function, we can associate with a gamble  $\gamma$  the random variable  $\{\lvert\gamma\rvert\}_M$ , where

$$\{b_1\varphi_1 + \dots + b_n\varphi_n\}_M = b_1X_{[\lvert\varphi_1\rvert]_M} + \dots + b_nX_{[\lvert\varphi_n\rvert]_M}.$$

Then  $M \models b_1e(\gamma_1) + \dots + b_ke(\gamma_k) \geq c$  iff  $b_1E_{\tilde{\mu}}(\{\lvert\gamma_1\rvert\}_M) + \dots + b_kE_{\tilde{\mu}}(\{\lvert\gamma_k\rvert\}_M) \geq c$ , where  $E_{\tilde{\mu}}$  is the expectation with respect to  $\tilde{\mu}$ . For uncertainty structure  $M$ , if  $\tilde{\mu}$  is probability  $P$ , then  $E_{\tilde{\mu}}$  is the expectation  $E_P$ . On the other hand, if  $\tilde{\mu}$  is based on other measures, such as belief functions, the sets of probabilities, and possible measures, then  $E_{\tilde{\mu}}$  is the expectation in the case of belief functions, the sets of probabilities, and possible measures, respectively [6]. From now on, we are only interested in the case of probability.

<sup>2</sup> In particular, in the case of probability measure, probability structure is denoted by  $M = (W, P, \pi)$ , and corresponding expectation is denoted by  $E_P$ .

### 3 What Are Hybrid Probabilistic Knowledge Bases?

#### 3.1 Conditional Expectation Constraints

We now extend the propositional logic for reasoning about expectation by introducing conditional expectation formulas<sup>3</sup>. A basic conditional expectation constraint has the form  $b_1e(\gamma_1|\phi) + \cdots + b_ke(\gamma_k|\phi) \geq c$ , where  $k \geq 1$ ,  $b_1, \dots, b_n, c \in \mathbb{R}$ ,  $\gamma_1, \dots, \gamma_k$  are linear propositional gambles, and  $\phi$  is a proposition. In general case, a basic conditional expectation constraint has the form  $b_1e(\gamma_1|\phi_1) + \cdots + b_ke(\gamma_k|\phi_k) \geq c$ , where  $k \geq 1$ ,  $\phi_1, \dots, \phi_k$  are propositions.

Similar to that of basic expectation formula, it is easy to define basic conditional expectation constraint with  $\leq$ ,  $=$ ,  $<$ , and  $>$ , using the logic operators. For example,  $b_1e(\gamma_1|\phi_1) + \cdots + b_ke(\gamma_k|\phi_k) \leq c$  is an abbreviation for  $-b_1e(\gamma_1|\phi_1) - \cdots - b_ke(\gamma_k|\phi_k) \geq -c$ . In particular, for a random variable  $X$ , let  $V(X) = \{x_1, \dots, x_n\}$  and  $\psi_i$  be the event  $X = x_i$ , then  $E_P(X|\phi) = \sum_{i=1}^n x_i P(\psi_i|\phi)$ . Thus, we view  $e(X|\phi) \geq c$  as an abbreviation for  $x_1e(\psi_1|\phi) + \cdots + x_ne(\psi_n|\phi) \geq c$ . For example, we may use  $5000 \leq e(\text{Vital\_capability}|\text{Swimmer}) \leq 6000$  to encode expectation knowledge “*Generally, the swimmer’s vital capability is between 5000 and 6000 milliliter*”.

A conditional expectation constraint is a Boolean combination of basic conditional expectation constraints. The set of expectation formulas is the closure of all conditional expectation constraints under  $\neg$  and  $\wedge$ .

Just for convenience, in this paper, a conditional expectation constraint has the form  $c \leq b_1e(\gamma_1|\phi_1) + \cdots + b_ke(\gamma_k|\phi_k) \leq d$ , where  $k \geq 1$ ,  $\phi_1, \dots, \phi_k$  are propositions.

Now we present semantics for conditional expectation constraints in the form of probability structure  $M = (W, P, \pi)$  as follows:  $M \models b_1e(\gamma_1|\phi) + \cdots + b_ke(\gamma_k|\phi) \geq c$  iff  $P([\phi]_M) = 0$  or  $b_1E_P(\{|\gamma_1|\}_M | X_{[\phi]_M} = 1) + \cdots + b_kE_P(\{|\gamma_k|\}_M | X_{[\phi]_M} = 1) \geq c$ .

Generally,  $M \models b_1e(\gamma_1|\phi_1) + \cdots + b_ke(\gamma_k|\phi_k) \geq c$  iff  $\exists i \in \{1, \dots, k\}$ ,  $P([\phi_i]_M) = 0$  or  $b_1E_P(\{|\gamma_1|\}_M | X_{[\phi_1]_M} = 1) + \cdots + b_kE_P(\{|\gamma_k|\}_M | X_{[\phi_k]_M} = 1) \geq c$ .

For the truth of conditional expectation constraints and logical constraints  $F, G$  is defined by follows:  $M \models \neg F$  iff not  $M \models F$ ;  $M \models (F \wedge G)$  iff  $M \models F$  and  $M \models G$ ;  $M$  is a model of  $F$  iff  $M \models F$ . For a set of logical constraints and expectation constraints  $\mathcal{F}$ ,  $M$  is a model of  $\mathcal{F}$ , or  $M$  satisfies  $\mathcal{F}$ , denoted  $M \models \mathcal{F}$ , iff  $M$  is a model of all  $F \in \mathcal{F}$ .  $\mathcal{F}$  is satisfiable iff a model of  $\mathcal{F}$  exists. A conditional expectation constraint  $C$ :  $c \leq b_1e(\gamma_1|\phi_1) + \cdots + b_ke(\gamma_k|\phi_k) \leq d$  is a logical consequence of  $\mathcal{F}$ , denoted  $\mathcal{F} \models C$ , iff each model of  $\mathcal{F}$  is also a model of the constraint.  $C$  is a tight logical consequence of  $\mathcal{F}$ , denoted  $\mathcal{F} \models_{tight} C$ , iff  $c = \inf b_1e(\gamma_1|\phi_1) + \cdots + b_ke(\gamma_k|\phi_k)$  (resp.  $d = \sup b_1e(\gamma_1|\phi_1) + \cdots + b_ke(\gamma_k|\phi_k)$ ) subject to all models of  $\mathcal{F}$  with  $P(\phi_1) \cdots P(\phi_k) \geq 0$ .

In particular, if  $\phi_1 = \cdots = \phi_k = \top$ , then  $b_1e(\gamma_1|\phi_1) + \cdots + b_ke(\gamma_k|\phi_k) \geq c$  is equal to  $b_1e(\gamma_1) + \cdots + b_ke(\gamma_k) \geq c$  in syntax as well as semantics, so we can

<sup>3</sup> For consistency in terms with probabilistic default theory [5], the conditional expectation formula is denoted by conditional expectation constraint.

view the expectation formula proposed by Halpern [6] as a special conditional expectation constraint.

### 3.2 Conditional Probability Constraints

Generally, in nonmonotonic probabilistic logics [1,5], a conditional probability constraint has the form  $(\psi|\phi)[l, u]$ , where  $\psi$  and  $\phi$  are events and  $l, u \in [0, 1]$  are reals. In this paper, a general conditional probability constraint has the form  $b_1p(\psi_1|\phi_1) + \dots + b_kp(\psi_k|\phi_k) \geq c$ , where  $k \geq 1$ ,  $b_1, \dots, b_n, c \in \mathbb{R}$ ,  $\psi_1, \phi_1, \dots, \psi_k, \phi_k$  are events. We also present semantics for conditional probability constraints in the form of probability structure  $M = (W, P, \pi)$  as follows:  $M \models b_1p(\psi_1|\phi_1) + \dots + b_kp(\psi_k|\phi_k) \geq c$  iff  $\exists i \in \{1, \dots, k\}, P(\phi_i) = 0$  or  $b_1P(\psi_1|\phi_1) + \dots + b_kP(\psi_k|\phi_k) \geq c$ . The other details are similar to that of conditional expectation constraints.

Just for consistency in terms, in this paper, we assume that the conditional probability constraint has the form  $l \leq p(\psi|\phi) \leq u$ , where  $l, u \in [0, 1]$ . The techniques for reasoning about conditional probability constraints have been discussed in [1,5]. This paper focuses on the technique for reasoning about conditional expectation constraints in the subsequent sections.

### 3.3 Hybrid Probabilistic Knowledge Bases

A hybrid probabilistic knowledge base  $HKB = (L, E, P)$  consists of a finite set of logical constraints  $L$ , a finite set of conditional expectation constraints  $E$  and a finite set of conditional probability constraints  $P$ . Just for convenience, we use  $H_C$  to denote  $E \cup P$ , then the hybrid probabilistic knowledge base  $HKB = (L, E, P)$  can be abbreviated as  $HKB = (L, H_C)$ , where  $H_C$  is a finite set of conditional constraints. We say  $HKB$  is satisfiable iff  $L \cup H_C$  is satisfiable. A conditional constraint  $C$  is a logical consequence of  $HKB$ , denoted  $HKB \models C$ , iff  $L \cup H_C \models C$ . It is a tight logical consequence of  $HKB$ , denoted  $L \cup H_C \models_{tight} C$ , iff  $L \cup H_C \models_{tight} C$ . Now we give an example to illustrate hybrid probabilistic knowledge bases.

*Example 1.* Consider the following scenario. We have the following knowledge: *All swimmers are healthy persons; A swimmer is also an athlete with a probability of at most 0.05; (Generally), the healthy person's vital capability is between 2500 and 4000 milliliters.*

We use  $HKB = (L, H_C)$  to express the knowledge, where  $L = \{\text{swimmer} \rightarrow \text{healthy} (\text{or } p(\text{healthy}|\text{swimmer}) = 1)\}$  and  $H_C = \{0 \leq p(\text{athlete}|\text{swimmer}) \leq 0.05, 2500 \leq e(V\_C|\text{healthy}) \leq 4000\}$ .

A particular thing is that we translate the defeasible conditional constraints into the same formulas as strict conditional constraints in the above. However, defeasible constraints have exceptions, which should be handled in a special way. For a hybrid probabilistic knowledge base  $HKB = (L, H_C)$ , we use  $L^S$ ,  $L^D$ ,  $H_C^S$  and  $H_C^D$  to denote the set of strict logical constraints, the set of default logical constraints, the set of strict conditional constraints, and the set of default

conditional constraints, respectively. Then  $L^S \cup L^D = L$  and  $H_C^S \cup H_C^D = H_C$ . Further, let  $L^E = L^S \cup H_C^S$  and  $D^E = L^D \cup H_C^D$ , then we use  $(L^E, D^E)$  to denote a hybrid knowledge base that consists of a set of strict constraints and a set of default constraint.

## 4 Nonmonotonic Reasoning About Hybrid Probabilistic Knowledge Bases

We first introduce some basic conceptions about default reasoning. A probability structure  $M = (W, P, \pi)$  verifies a default  $b_1e(\gamma_1|\phi) + \cdots + b_k e(\gamma_k|\phi) \geq c$  iff  $P(\phi) = 1$  and  $M \models b_1e(\gamma_1|\phi) + \cdots + b_k e(\gamma_k|\phi) \geq c$ .  $M$  falsifies a default  $b_1e(\gamma_1|\phi) + \cdots + b_k e(\gamma_k|\phi) \geq c$  iff  $P(\phi) = 1$  and  $M \not\models b_1e(\gamma_1|\phi) + \cdots + b_k e(\gamma_k|\phi) \geq c$ . A set of defaults  $D^E$  tolerates a default  $d$  under  $L^E$  iff  $L^E \cup D^E$  has a model that verifies  $d$ .  $D^E$  is under  $L^E$  in conflict with  $d$  iff  $L^E \cup D^E$  has no model that verifies  $d$ .

A default ranking  $\sigma$  on  $D^E$  maps each  $d \in D^E$  to a nonnegative integer. It is admissible with a hybrid probabilistic knowledge base  $HKB = (L^E, D^E)$  iff each set of default  $D'^E \subseteq D^E$  that is under  $L^E$  in conflict with some default  $d \in D^E$  contains a default  $d'$  such that  $\sigma(d') < \sigma(d)$ .  $HKB$  is  $\sigma$ -consistent iff there is a default ranking  $\sigma$  on  $D^E$  is admissible with  $HKB$ , otherwise, it is  $\sigma$ -inconsistent.

Let  $HKB$  be  $\sigma$ -consistent, the  $z$ -partition of  $D^E$  is the unique ordered partition  $(D_0^E, \dots, D_n^E)$  of  $D^E$  such that each  $D_i^E$  is the set of all  $d \in \bigcup_{j=i}^n D_j^E$  tolerated under  $L^E$  by  $\bigcup_{j=i}^n D_j^E$ . By the  $z$ -partition  $(D_0^E, \dots, D_n^E)$  of  $D^E$ , we define the preferability of structures. For uncertainty structures  $M$  and  $M'$ , we say  $M$  is *lexicographically preferable* (or *lex-preferable*) to  $M'$  iff  $\exists i \in \{0, \dots, n\}$  such that  $|\{d \in D_i^E : M \models d\}| > |\{d \in D_i^E : M' \models d\}|$  and  $|\{d \in D_j^E : M \models d\}| = |\{d \in D_j^E : M' \models d\}|$  for all  $i < j \leq n$ . A model  $M$  of a set of expectation formulas  $\mathcal{F}^E$  is a *lex-preferable minimal* model of  $\mathcal{F}^E$  iff no model of  $\mathcal{F}^E$  is *lex-preferable* to  $M$ .

Lukasiewicz [5] generalized the notion of *lex-consequence* in probabilistic reasoning. However, it can be extended to domain of reasoning about hybrid probabilistic knowledge bases as follows. A strict expectation formula  $F$  is a *lex-consequence* of  $HKB$ , denoted by  $HKB \parallel \sim^{lex} F$ , iff each *lex-preferable minimal* model of  $L^E$  satisfies  $F$ . A strict conditional formula  $b_1e(\gamma_1|\phi) + \cdots + b_k e(\gamma_k|\phi) \geq c$  is a *tight lex-consequence* of  $HKB$ , denoted by

$$HKB \parallel \sim_{tight}^{lex} b_1e(\gamma_1|\phi) + \cdots + b_k e(\gamma_k|\phi) \geq c$$

iff  $c$  is the infimum of

$$b_1 E_P(\{|\gamma_1|\}_M | X_{[|\phi|]M} = 1) + \cdots + b_k E_P(\{|\gamma_k|\}_M | X_{[|\phi|]M} = 1)$$

subject to all *lex-preferable minimal* models of  $L^E$  with  $P([|\phi|]_M) > 0$ . We extend the notion of *lex-consequence* to  $\sigma$ -inconsistent hybrid knowledge bases  $HKB$  by defining  $HKB \parallel \sim^{lex} F$  and

$$HKB \|\sim_{tight}^{lex} -\infty \geq b_1 e(\gamma_1|\phi) + \dots + b_k e(\gamma_k|\phi) \geq +\infty$$

for all knowledge base  $HKB$ , strict expectation formulas  $F$ , and propositional gambles.

Now we illustrate the default reasoning about hybrid probabilistic knowledge bases by the following examples.

*Example 2.* Consider the following scenario. We have the following knowledge:

- All athletes are healthy persons;
- Generally, an athlete is an actor with a probability of at most 0.1;
- Generally, the healthy person's vital capability is between 2500 and 4000 milliliters;
- Generally, the athlete's vital capability is between 5000 and 6000 milliliters;
- Generally, the healthy man's vital capability is between 3500 and 4000 milliliters;
- **Jordan** is a male athlete.

Now we express the knowledge by following propositions. let  $\phi = \text{"Athlete"}$ ,  $\psi = \text{"Actor"}$ ,  $v = \text{"Male"}$ ,  $\varphi = \text{"Healthy"}$  and  $V\_C$  be a random variable to denote vital capability, then  $HKB = (L^E, D^E)$ , where

$$\begin{aligned} L^E &= \{p(\varphi|\phi) = 1, p(\phi \wedge v|\top) = 1\} \\ D^E &= \{5000 \leq e(V\_C|\phi) \leq 6000, 0 \leq p(\psi|\phi) \leq 0.1, \\ &\quad 3500 \leq e(V\_C|v) \leq 4000, 2500 \leq e(V\_C|\varphi) \leq 4000\} \end{aligned}$$

Consider the  $z$ -partition of  $D^E = D_0^E \cup D_1^E$ , where

$$\begin{aligned} D_0^E &= \{3500 \leq e(V\_C|v) \leq 4000, 2500 \leq e(V\_C|\varphi) \leq 4000\}, \\ D_1^E &= \{5000 \leq e(V\_C|\phi) \leq 6000, 0 \leq p(\psi|\phi) \leq 0.1\} \end{aligned}$$

However, the set of all  $lex-minimal$  models of  $L^E$  is given by the set of all models of  $D_1^E \cup L^E$ . Thus

$$\begin{aligned} HKB \|\sim_{tight}^{lex} 5000 \leq e(V\_C|\top) \leq 6000, \\ HKB \|\sim_{tight}^{lex} 0 \leq p(\psi|\top) \leq 0.1. \end{aligned}$$

that is, we can get the result as follows:

**Jordan**'s vital capability is between 5000 and 6000 milliliters. **Jordan** is also an actor with a probability of at most 0.1.

This approach is similar to the strong nonmonotonic probabilistic reasoning in [5], we term this the approach to strong nonmonotonic reasoning about hybrid knowledge bases. However, it is not difficult to verify that *lex-entailment* satisfies the postulates proposed by Kraus [7] including right weakening, reflexive, left logical equivalence, Cut, Or, and Cautious Monotonicity, which are commonly considered as being particular desirable for any reasonable notion of non-monotonic entailment.

**Proposition 1.**  $\|\sim^{\text{lex}}\|$  satisfy the following properties for all hybrid knowledge bases  $HKB$  and  $HKB'$ , and all expectation formulas  $F$  and  $G$ :

- Right Weakening. If  $F \rightarrow G$  is logically valid and  $HKB \|\sim^{\text{lex}} F$ , then  $HKB \|\sim^{\text{lex}} G$ .
- Reflexive.  $HKB \|\sim^{\text{lex}} HKB$ .
- Left Logical Equivalence. If  $HKB \longleftrightarrow HKB'$  is logically valid, then  $HKB \|\sim^{\text{lex}} F$  iff  $HKB' \|\sim^{\text{lex}} F$ .
- Cut. If  $HKB \|\sim^{\text{lex}} F$  and  $HKB \wedge F \|\sim^{\text{lex}} G$ , then  $HKB \|\sim^{\text{lex}} G$ .
- Cautious Monotonicity. If  $HKB \|\sim^{\text{lex}} F$  and  $HKB \|\sim^{\text{lex}} G$ , then  $HKB \wedge F \|\sim^{\text{lex}} G$ .
- Or. If  $HKB \|\sim^{\text{lex}} F$  and  $HKB' \|\sim^{\text{lex}} F$ , then  $HKB \vee HKB' \|\sim^{\text{lex}} F$ .

It also satisfies the desirable properties of rational monotonicity, irrelevance [8] and direct inference [9].

**Proposition 2.**  $\|\sim^{\text{lex}}\|$  satisfy the following properties for all hybrid knowledge bases  $HKB$  and all expectation formulas  $F$  and  $G$ :

- Rational Monotonicity. If  $HKB \|\sim^{\text{lex}} F$  and  $HKB \|\not\sim^{\text{lex}} \neg G$ , then  $HKB \wedge G \|\sim^{\text{lex}} F$ .
- Irrelevance. If  $HKB \|\sim^{\text{lex}} F$ , and no atom of  $HKB$  and  $F$  occurs in  $G$ , then  $HKB \wedge G \|\sim^{\text{lex}} F$ .
- Direct Inference. If  $b_1e(\gamma_1|\phi_1) + \dots + b_k e(\gamma_k|\phi_k) \geq c \in D^E$  and

$$\{p(\phi_1|\top) = 1 \wedge \dots \wedge p(\phi_k|\top) = 1\} \longleftrightarrow L^E$$

is logically valid, then  $HKB \|\sim^{\text{lex}} b_1e(\gamma_1|\top) + \dots + b_k e(\gamma_k|\top) \geq c$ .

## 5 Discussion

Actually, the defeasible conditional constraints can be handled in different ways, such as weak nonmonotonic probabilistic logics [4] and nonmonotonic probabilistic logics under variable-strength inheritance with overriding [1]. Most of these techniques for defeasible conditional probability constraints may be generalized to handle defeasible conditional expectation constraints.

Compared to the similar nonmonotonic probabilistic logics in literature [1,4,10,11,12,13], the extended logic for reasoning about expectation presented above provides a tool to express conditional expectation constraints as well as conditional probability constraints. Thus, it is appropriate for encoding hybrid probabilistic knowledge. Moreover, in this sense, it is easy to see that the hybrid probabilistic knowledge base is at least as expressive as purely probabilistic knowledge base.

However, similar in [6], it is easy to get that the logic reasoning for hybrid conditional constraints presented in this paper and the corresponding logic for reasoning about purely probabilistic constraints are equivalent in expressive power with respect to the set of all probability structures  $\mathcal{M}$ .

## 6 Conclusion

We have presented a logic for reasoning about hybrid probabilistic knowledge bases in the paper. By introducing the conditional expectation constraint formula, we encode both the conditional expectation constraint and the conditional probability constraint in a uniform logical framework based on the propositional logic for reasoning about expectation, which was proposed by Halpern [6]. Then we provide an approach to nonmonotonic reasoning about hybrid probabilistic knowledge bases. As far as the expressive power is concerned, the logic for reasoning about hybrid conditional constraints and the corresponding logic for reasoning about purely probabilistic constraints are equivalent.

## Acknowledgements

This work was partly supported by the National Natural Science Foundation of China (No.60233010, No.60496322, and No.60496324), the National Key Research and Development Program (Grant No. 2002CB312004) of China, the Knowledge Innovation Program of the Chinese Academy of Sciences, and the NSFC and the British Royal Society China-UK Joint Project.

## References

1. Lukasiewicz, T.: Nonmonotonic probabilistic logics under variable-strength inheritance with overriding:algorithms and implementation in nmproblog. In: 4th International Symposium on Imprecise Probabilities and Their Applications. (2005) 230–239
2. Fagin, R., Halpern, J., Megiddo, N.: A logic for reasoning about probabilities. *Information and Computation* **87** (1990) 78–128
3. Dubois, D., Prade, H.: Inference with imprecise numerical quantifiers. (1990)
4. Lukasiewicz, T.: Weak nonmonotonic probabilistic logics. In: Proceedings KR-2004. (2004) 22–33
5. Lukasiewicz, T.: Probabilistic default reasoning with conditional constraints. *Annals of Mathematics and Artificial Intelligence* **34** (2002) 35–88
6. Halpern, J., Pucella, R.: Reasoning about expectation. In: Proceedings of the Eighteenth Conference on Uncertainty in AI. (2002) 207–215
7. Kraus, S., Lehmann, D., Magidor, M.: Non-monotonic reasoning, preferential models and cumulative logics. *Artif. Intell.* **14** (1990) 167–207
8. Benferhat, S., Saffiotti, A., Smets, P.: Belief functions and default reasoning. *Artif. Intell.* **122** (2000) 1–69
9. Bacchus, F., Grove, A., Halpern, J., Koller, D.: From statical knowledge bases to degrees of beliefs. *Artif.Intell.* **87** (1996) 75–143
10. Pearl, J.: System z: A natural ordering of default with tractable applications in default reasoning. In: proceedings of TARK-90. (1990) 121–135
11. Lehmann, D.: Another perspective on default reasoning. *Ann. Math. Artif. Intell.* **15** (1995) 61–82
12. Geffner, H.: Causal and conditional theories. MIT Press, Combridge, MA (1992)
13. Geffner, H., Pearl, J.: Conditional entailment: Bridging two approach to default reasoning. *Artif.Intell.* **53** (1992) 209–244

# Update Rules for Parameter Estimation in Continuous Time Bayesian Network

Dongyu Shi and Jinyuan You

Department of Computer Science and Engineering Shanghai Jiao Tong University,  
Shanghai 200030, P.R. China  
[{shi-dy, you-jy}@cs.sjtu.edu.cn](mailto:{shi-dy, you-jy}@cs.sjtu.edu.cn)

**Abstract.** Continuous time Bayesian network is a new kind of dynamic graphical models developed in recent year, which describe structured stochastic processes with finitely many states that evolve over continuous time. The parameters for each variable in the model represent a finite state continuous time Markov process, whose transition model is a function of its parents. This paper presents an algorithm for updating parameters from an existing CTBN model with a set of data samples. It is a unified framework for online parameter estimation and batch parameter updating where a pre-accumulated set of samples is used. We analyze different conditions of the algorithm, and show its performance in experiments.

## 1 Introduction

Many applications involve systems that have causal or probabilistic relationships in temporal processes. Learning about these complex dynamic systems is an important task. Different processes include analyzing web server logs, studying trends in demographic data, and learning dynamic models for information processing in sensor networks. There are some standard frameworks, such as state-space models [1] and dynamic Bayesian networks (DBNs) [5], discretize time at fixed intervals, known as time slices, and model the system as evolving discretely from one time slice to the next. However, many systems do not have natural time granularity. Some variables evolve quickly, whereas others change more slowly. Sometimes even the same variable can change more quickly or slowly in some conditions. In such systems, attempts to model the system as evolving over uniform discrete time intervals either lead to very coarse approximations, or require that the entire trajectory be modeled at a very fine granularity, which usually result in a high computational cost. When estimating the parameters in such a system, this problem is more acute if there is little prior knowledge about the evolution of different variables.

To solve this problem, some models are presented as evolving over continuous time. In [3], Blossfeld and Rohwer present event history analysis for discrete state systems. In [8], Lando presents a Markov process model over continuous time. In [9], Nodelman et al. extend the former representations, and present the framework of Continuous Time Bayesian Networks (CTBNs). This framework

allows for modeling stochastic processes over a structured state space evolving in continuous time. The model has gained great success in dynamic systems modeling. For learning the parameters and structures in CTBNs, they present a Maximum-Likelihood Estimation algorithm in [10] and an EM algorithm in [11]. The learning algorithms they provide are offline batch learning methods based on a given training set of data cases. In many real-world applications, we may not get sufficient training data at the beginning, but keep getting data cases during the time when the system is working. Even when the model is finely learned, the parameters are often required to be adjusted by new evidences since it works in a dynamic environment. Some works have been done in related areas. For online parameter estimation in DBNs, the parameters are simply added to the state space and then do online inference [7] (i.e., filtering). Online learning of traditional BN parameters is in the work of [2] and [4].

In this paper, we present a learning algorithm for update parameters in a CTBN. We provide the detailed process on how to update the parameters on a set of data cases from existing parameter values (either learned or assigned) in the model. The applicability of the algorithm to the problems of adapting the CTBN parameters is shown in analysis and experiments. The rest of the paper is organized as follows: we begin in Section 2 with introduction of the CTBNs. In Section 3, we derive the parameter update rules based on a basic framework for the system. And section 4 is the experimental results and conclusion.

## 2 Continuous Time Bayesian Networks

**A Continuous Time Bayesian Network** (CTBN) [9] represents a stochastic process over a structured state space, consisting of assignments to some set of local variables  $X = \{X_1, \dots, X_n\}$ , where each  $X_i$  has a finite domain of values  $Val(X_i)$ . First consider a Markov process over a single variable. A finite state, continuous time, homogeneous Markov process  $X(t)$  with state space  $Val(X) = \{x_1, \dots, x_N\}$  is described by an initial distribution  $P_X^0$  and an  $N \times N$  matrix of transition intensities:

$$\mathbf{Q}_X = \begin{pmatrix} -q_{x_1} & q_{x_1 x_2} & \cdots & q_{x_1 x_N} \\ q_{x_2 x_1} & -q_{x_2} & \cdots & q_{x_2 x_N} \\ \vdots & \vdots & \ddots & \vdots \\ q_{x_N x_1} & q_{x_N x_2} & \cdots & -q_{x_N} \end{pmatrix} \quad (1)$$

where  $q_{x_i x_j} > 0$  is the intensity of transition from state  $x_i$  to state  $x_j$ ,  $j \neq i$  and  $q_{x_i} = \sum_j q_{x_i x_j}$ . Given  $\mathbf{Q}_X$ , the transient behavior of  $X(t)$  can be described as follows. If  $X(0) = x$  then it stays in state  $x$  for an amount of time exponentially distributed with parameter  $q_x$ . Thus, the probability density function  $f$  for  $X(t)$  remaining at  $x$  is  $f(q_x, t) = q_x \exp(-q_x t)$  for  $t \geq 0$ , and the corresponding probability distribution function  $F$  for  $X(t)$  remaining at  $x$  for an amount of time  $\geq t$  is given by  $F(q_x, t) = 1 - \exp(-q_x t)$ . The expected time of transition is  $1/q_x$ . Upon the transition,  $X$  shifts to state  $x'$  with probability  $\theta_{xx'} = q_{xx'}/q_x$ .

In a CTBN, the joint dynamics of the local variables are modeled by allowing the transition model of each local variable  $X$  to be a Markov process. And its parameterization depends on some subset of other variables  $\mathbf{U}$ . The key building block is a conditional Markov process. A **conditional Markov process**  $X$  is an inhomogeneous Markov process whose intensity matrix varies with time, but only as a function of the current values of a set of discrete conditioning variables  $\mathbf{U}$ . Its intensity matrix, called **conditional intensity matrix** (CIM), is written as  $\mathbf{Q}_{X|\mathbf{U}}$ , and can be viewed as a set of homogeneous intensity matrices  $\mathbf{Q}_{X|\mathbf{u}}$  - one for each instantiation of values  $\mathbf{Q}_{X|\mathbf{U}}$  to  $\mathbf{U}$ . The parameters of  $\mathbf{Q}_{X|\mathbf{U}}$  are  $q_{X|\mathbf{u}} = \{q_{x|\mathbf{u}} : x \in Val(X)\}$  and  $\theta_{X|\mathbf{u}} = \{\theta_{xx'|\mathbf{u}} : x, x' \in Val(X), x' \neq x\}$ . Combining the set of conditional Markov processes of all the variables, a CTBN is formed as following:

**Definition 1.** A continuous time Bayesian network  $\mathcal{N}$  over  $\mathbf{X}$  consists of two components: an initial distribution  $\mathbf{P}_{\mathbf{X}}^0$ , specified as a Bayesian network  $\mathcal{B}$  over  $\mathbf{X}$ , and a continuous transition model, specified as

- A directed (possibly cyclic) graph  $\mathcal{G}$  whose nodes are  $X_1, \dots, X_n$ ;  $Pa_{\mathcal{G}}(X_i)$ , often abbreviated  $\mathbf{U}_i$ , denotes the parents of  $X_i$  in  $\mathcal{G}$ .
- A conditional intensity matrix,  $\mathbf{Q}_{X_i|\mathbf{U}_i}$ , for each variable  $X_i \in \mathbf{X}$ .

The transitions of each local variable in a CTBN are controlled by the values of its parents.

### 3 Update Rules for CTBN Parameter Estimation

The general task is to update the parameters of the CTBN model from a set of data. This implementation assumes a fixed structure  $\mathcal{G}$  of the network, and that the variables are discrete valued. In a CTBN  $\mathcal{N}$ , each variable  $X \in \mathbf{X}$  is conditioned on its parent set  $\mathbf{U}$ , and each transition of  $X$  is considered in the context of the instantiation to  $\mathbf{U}$ . The transition is described by the conditional intensity matrix (CIM):

$$\mathbf{Q}_{X_i|\mathbf{U}_i=\mathbf{u}} = \begin{pmatrix} -q_{x_1|\mathbf{u}} & q_{x_1|\mathbf{u}} \cdot \theta_{x_1x_2|\mathbf{u}} & \dots & q_{x_1|\mathbf{u}} \cdot \theta_{x_1x_N|\mathbf{u}} \\ q_{x_2|\mathbf{u}} \cdot \theta_{x_2x_1|\mathbf{u}} & -q_{x_2|\mathbf{u}} & \dots & q_{x_2|\mathbf{u}} \cdot \theta_{x_2x_N|\mathbf{u}} \\ \vdots & \vdots & \ddots & \vdots \\ q_{x_N|\mathbf{u}} \cdot \theta_{x_Nx_1|\mathbf{u}} & q_{x_N|\mathbf{u}} \cdot \theta_{x_Nx_2|\mathbf{u}} & \dots & -q_{x_N|\mathbf{u}} \end{pmatrix} \quad (2)$$

The distribution over transitions of  $X$  under  $U = \mathbf{u}$  factors into two pieces: an exponential distribution over when the next transition will occur and a multinomial distribution over where the state transitions—the next state of the system. The natural parameter for the exponential distribution is  $q_{x|\mathbf{u}}$  and the natural parameters for the multinomial distribution are  $\theta_{xx'|\mathbf{u}}, x' \neq x$ . (As staying in the state  $x$  is not a transition of  $X$ , there is no multinomial parameter  $\theta_{xx|\mathbf{u}}$ .) The learning is then the estimation of the CIM parameters –  $\{q_{x|\mathbf{u}}, \theta_{xx'|\mathbf{u}}\}$  for each  $X_i$  and  $U_i$  of the network.

### 3.1 The Basic Equations

The data we use to learn the parameters are a set of trajectories  $\mathbf{D} = \{\sigma_1, \dots, \sigma_H\}$ , where each  $\sigma_i$  is a set of state transitions and the times at which they occurred. Consider a conditional Markov process  $P_{X|\mathbf{u}}(t)$ . The likelihood of  $\mathbf{D}$  can be decomposed as a product of the likelihoods for individual transition  $d$ . Let  $d = < x_d, t_d, x'_d > \in \sigma_i$  be the transition where  $X$  transits to state  $x'_d$  after spending the amount of time  $t_d$  in state  $x_d$ . The likelihood for the single transition  $d$  can be written as

$$\begin{aligned} L_{X|\mathbf{u}}(\mathbf{q}, \boldsymbol{\theta} : d) &= L_{X|\mathbf{u}}(\mathbf{q} : d) \cdot L_{X|\mathbf{u}}(\boldsymbol{\theta} : d) \\ &= q_{x_d} \exp(-q_{x_d} t_d) \cdot \theta_{x_d x'_d}. \end{aligned} \quad (3)$$

The data  $\mathbf{D}$  can be summarized in terms of the sufficient statistics:  $T[x|\mathbf{u}]$ , the amount of time  $X$  spends in state  $x$ ;  $M[x, x'|\mathbf{u}]$ , the number of times  $X$  transitions from  $x$  to  $x'$  for  $x' \neq x$ ; and  $M[x|\mathbf{u}] = \sum_{x'} M[x, x'|\mathbf{u}]$ , the total number of transitions leaving the state  $X = x$ .

Our task is as follows: we have a current model with assignment of parameters  $\bar{\rho} = (\bar{\mathbf{q}}, \bar{\boldsymbol{\theta}})$ , and a set of data  $\mathbf{D}$ . We want to construct a new model  $\tilde{\rho} = (\tilde{\mathbf{q}}, \tilde{\boldsymbol{\theta}})$  based on  $(\bar{\mathbf{q}}, \bar{\boldsymbol{\theta}})$  and  $\mathbf{D}$ . Similar to the framework in [2], the choice of  $\tilde{\rho}$  should balance two factors: the potential increases to the log-likelihood of the data sets in the current parameters, and the extent to which it moves away from the current model. Therefore the updating of the CTBN  $\mathcal{N}$ 's parameters is achieved by the following maximization:

$$\begin{aligned} \tilde{\rho} &= \arg \max_{\rho} [F(\rho)] \\ &= \arg \max_{\rho} [\eta l_{\mathcal{N}}(\rho : \mathbf{D}) - d(\rho, \bar{\rho})]. \end{aligned} \quad (4)$$

$\eta > 0$  is the learning rate. The term  $l_{\mathcal{N}}(\rho : \mathbf{D})$  is the normalized log-likelihood of  $\mathbf{D}$ . It's computed in the following way. The likelihood of  $\mathcal{N}$  can be decomposed by variable as

$$\begin{aligned} L_{\mathcal{N}}(\mathbf{q}, \boldsymbol{\theta} : \mathbf{D}) &= \prod_{X_i \in \mathbf{X}} L_{X_i}(\mathbf{q}_{X_i|\mathbf{U}_i}, \boldsymbol{\theta}_{X_i|\mathbf{U}_i} : \mathbf{D}) \\ &= \prod_{X_i \in \mathbf{X}} L_{X_i}(\mathbf{q}_{X_i|\mathbf{U}_i} : \mathbf{D}) \cdot L_{X_i}(\boldsymbol{\theta}_{X_i|\mathbf{U}_i} : \mathbf{D}). \end{aligned} \quad (5)$$

According to [10], the log-likelihood of a local variable  $X_i$  can be written as a sum of the log-likelihoods in the following form:

$$\begin{aligned} l_{X_i}(\mathbf{q}, \boldsymbol{\theta} : \mathbf{D}) &= l_{X_i}(\mathbf{q} : \mathbf{D}) + l_{X_i}(\boldsymbol{\theta} : \mathbf{D}) \\ &= \sum_{\mathbf{u}} \sum_x (M[x|\mathbf{u}] \ln(q_{x|\mathbf{u}}) - q_{x|\mathbf{u}} \cdot T[x|\mathbf{u}]) \\ &\quad + \sum_{\mathbf{u}} \sum_x \sum_{x' \neq x} M[x, x'|\mathbf{u}] \ln(\theta_{xx'|\mathbf{u}}). \end{aligned} \quad (6)$$

And the total log-likelihood of the model is

$$l_{\mathcal{N}}(\boldsymbol{\rho} : \mathbf{D}) = \sum_i l_{X_i}(\mathbf{q}, \boldsymbol{\theta} : \mathbf{D}). \quad (7)$$

For a local variable  $X_i$ , when its parents  $U_i = \mathbf{u}_j$ , the corresponding sufficient statistics from  $\mathbf{D}$  are:  $T_{ijk}$ , the amount of time that  $X_i$  stays in  $x_k$ ;  $M_{ijkn}$ , the number of times that  $X_i$  transitions from  $x_k$  to  $x_n$ ; and  $M_{ijk} = \sum_n M_{ijkn}$ . Let  $l(\mathbf{q} : \mathbf{D}) = \sum_{ijk} (M_{ijk} \ln q_{ijk} - q_{ijk} T_{ijk})$ , and  $l(\boldsymbol{\theta} : \mathbf{D}) = \sum_{ijkn} M_{ijkn} \ln \theta_{ijkn}$ , we can have

$$l_{\mathcal{N}}(\boldsymbol{\rho} : \mathbf{D}) = l(\mathbf{q} : \mathbf{D}) + l(\boldsymbol{\theta} : \mathbf{D}). \quad (8)$$

We maximize (4) in the following way. There are two kinds of parameters in the model, “ $q$ ” and “ $\theta$ ”. For every parameter  $q_{ijk}$ , we conclude that the solution should satisfy

$$\frac{\partial F(\tilde{\boldsymbol{\rho}})}{\partial \tilde{q}_{ijk}} = \frac{\partial}{\partial \tilde{q}_{ijk}} [\eta l(\tilde{\mathbf{q}} : \mathbf{D}) - d(\tilde{\boldsymbol{\rho}}, \bar{\boldsymbol{\rho}})] = 0. \quad (9)$$

for every  $i, j, k$ . That is

$$\eta \frac{\partial (M_{ijk} \ln \tilde{q}_{ijk} - \tilde{q}_{ijk} T_{ijk})}{\partial \tilde{q}_{ijk}} - \frac{\partial d(\tilde{\boldsymbol{\rho}}, \bar{\boldsymbol{\rho}})}{\partial \tilde{q}_{ijk}} = 0. \quad (10)$$

$$\eta \left( \frac{M_{ijk}}{\tilde{q}_{ijk}} - T_{ijk} \right) - \frac{\partial d(\tilde{\boldsymbol{\rho}}, \bar{\boldsymbol{\rho}})}{\partial \tilde{q}_{ijk}} = 0. \quad (11)$$

For every parameter  $\theta_{ijkn}$ , the maximization problem is a constrained optimization problem. That is, for every  $i, j, k$ , we must have that  $\sum_n \theta_{ijkn} = 1$ . Introducing Lagrange multipliers for these constraints we conclude that the solution should satisfy

$$\frac{\partial}{\partial \tilde{\theta}_{ijkn}} [F(\tilde{\boldsymbol{\rho}}) + \sum_{i'j'k'} \beta_{i'j'k'} \left( \sum_{n', n' \neq k'} \tilde{\theta}_{i'j'k'n'} - 1 \right)] = 0. \quad (12)$$

for every  $i, j, k$ . That is

$$\frac{\partial}{\partial \tilde{\theta}_{ijkn}} [\eta l(\tilde{\boldsymbol{\theta}} : \mathbf{D}) - d(\tilde{\boldsymbol{\rho}}, \bar{\boldsymbol{\rho}})] + \beta_{ijk} = 0. \quad (13)$$

It is computationally difficult to directly use  $l(\tilde{\boldsymbol{\theta}} : \mathbf{D})$  for the maximization. Thus, in the vicinity of  $\bar{\boldsymbol{\theta}}$ , the log-likelihood is approximated by its first order Taylor approximation:

$$l(\tilde{\boldsymbol{\theta}} : \mathbf{D}) \approx l(\bar{\boldsymbol{\theta}} : \mathbf{D}) + \nabla l(\bar{\boldsymbol{\theta}} : \mathbf{D}) \cdot (\tilde{\boldsymbol{\theta}} - \bar{\boldsymbol{\theta}}). \quad (14)$$

The gradient of  $l(\boldsymbol{\theta} : \mathbf{D})$  is computed as following:

$$\nabla l(\boldsymbol{\theta} : \mathbf{D}) = \sum_{ijkn} \nabla_{ijkn} l(\boldsymbol{\theta} : \mathbf{D}). \quad (15)$$

$$\begin{aligned} \nabla_{ijkn} l(\boldsymbol{\theta} : \mathbf{D}) &= \frac{\partial}{\partial \theta_{ijkn}} \sum_{ijkn} M_{ijkn} \ln \theta_{ijkn} \\ &= M_{ijkn} / \theta_{ijkn}. \end{aligned} \quad (16)$$

So the equation (13) is simplified to

$$\eta \frac{M_{ijkn}}{\tilde{\theta}_{ijkn}} - \frac{\partial d(\tilde{\rho}, \bar{\rho})}{\partial \tilde{\theta}_{ijkn}} + \beta_{ijk} = 0. \quad (17)$$

The equation (11) and (17) represent two sets of basic equations for the maximization problem.

### 3.2 The Update Rules

To resolve the maximization problem we must choose an appropriate distance functions for the above basic equations. The term  $d(\tilde{\rho}, \bar{\rho})$  is an estimate of the distance between the new and old models. The two parameter vectors  $\tilde{\rho}$  and  $\bar{\rho}$ , with the associated network structure and transition process model, define two probability distributions over the same space of the variables in the CTBN. So we can use a distance function to compare the two parameter vectors over this space. Its effect is to keep  $\tilde{\rho}$  close to  $\bar{\rho}$ .

Here we use the widely-applied KL distance (also known as relative entropy) for the two parameter vectors. The joint distributions over all the factors in a CTBN are the multiplications of all the exponential distributions over when the next transition will occur, and all the multinomial distributions over which is the next state. By using KL distance, we get

$$\begin{aligned} d(\tilde{\rho}, \bar{\rho}) &= KL(\tilde{\rho} || \bar{\rho}) \\ &= KL(\tilde{q} || \bar{q}) + KL(\tilde{\theta} || \bar{\theta}) \\ &= \sum_{ijk} KL(\tilde{q}_{ijk} || \bar{q}_{ijk}) + \sum_{ijk} KL(\tilde{\theta}_{ijk} || \bar{\theta}_{ijk}). \end{aligned} \quad (18)$$

Where the  $\tilde{\theta}_{ijk}$  is the vector  $(\tilde{\theta}_{ijk1}, \dots, \tilde{\theta}_{ijk,k-1}, \tilde{\theta}_{ijk,k+1}, \dots, \tilde{\theta}_{ijkN})$ , or (equivalently) the distribution over different transition results  $x_i^n$ , from state  $x_i^k$  of  $X_i$ . Note that the two kinds of parameters are from the conditional intensity matrix of the CTBN model, represent intensity and discrete distribution respectively.

For the part of the exponential distributions, we have

$$\begin{aligned} KL(\tilde{q} || \bar{q}) &= \int_0^{+\infty} \tilde{q} e^{\tilde{q}t} \ln \frac{\tilde{q} e^{\tilde{q}t}}{\bar{q} e^{\bar{q}t}} dt \\ &= \ln \frac{\tilde{q}}{\bar{q}} + \frac{\bar{q}}{\tilde{q}} - 1. \end{aligned} \quad (19)$$

Put the result into equation (11), and we have

$$\eta \left( \frac{M_{ijk}}{\tilde{q}_{ijk}} - T_{ijk} \right) - \frac{\partial}{\partial \tilde{q}_{ijk}} \sum_{ijk} \left( \ln \frac{\tilde{q}_{ijk}}{\bar{q}_{ijk}} + \frac{\bar{q}_{ijk}}{\tilde{q}_{ijk}} - 1 \right) = 0. \quad (20)$$

$$\eta \left( \frac{M_{ijk}}{\tilde{q}_{ijk}} - T_{ijk} \right) + \frac{\bar{q}_{ijk}}{\tilde{q}_{ijk}^2} - \frac{1}{\tilde{q}_{ijk}} = 0. \quad (21)$$

Solving this equation we get

$$\tilde{q}_{ijk} = \frac{\eta M_{ijk} - 1 + \sqrt{(\eta M_{ijk} - 1)^2 + 4\eta T_{ijk}\bar{q}_{ijk}}}{2\eta T_{ijk}}. \quad (22)$$

For the part of multinomial distributions, we have

$$KL(\tilde{\boldsymbol{\theta}}_{ijk} || \bar{\boldsymbol{\theta}}_{ijk}) = \sum_n \tilde{\theta}_{ijkn} \ln \frac{\tilde{\theta}_{ijkn}}{\bar{\theta}_{ijkn}}. \quad (23)$$

Put it into equation (17), and we have

$$\eta \frac{M_{ijkn}}{\bar{\theta}_{ijkn}} - (\ln \frac{\tilde{\theta}_{ijkn}}{\bar{\theta}_{ijkn}} + 1) + \beta_{ijk} = 0. \quad (24)$$

$$\tilde{\theta}_{ijkn} = e^{\eta \frac{M_{ijkn}}{\bar{\theta}_{ijkn}} (\beta_{ijk} - 1)} \cdot \bar{\theta}_{ijkn}. \quad (25)$$

Summing over  $n$  to get  $\beta_{ijk}$ , we get

$$\beta_{ijk} = -\ln \sum_n \bar{\theta}_{ijkn} e^{\eta M_{ijkn} / \bar{\theta}_{ijkn}} + 1. \quad (26)$$

So the parameter  $\theta_{ijkn}$  is updated by

$$\tilde{\theta}_{ijkn} = \bar{\theta}_{ijkn} \cdot \frac{e^{\eta M_{ijkn} / \bar{\theta}_{ijkn}}}{\sum_n \bar{\theta}_{ijkn} e^{\eta M_{ijkn} / \bar{\theta}_{ijkn}}}. \quad (27)$$

### 3.3 Analysis of the Rules

Equation (22) and (27) can be seen as update rules for a model's parameters based on a current parameter vector and a set of data cases. We take some analysis of the rules and exceptions. In the case that  $T_{ijk}$  is extremely small, that means in the data sample,  $(U_i = \mathbf{u}_j, X_i = x_k)$  only exists a very short time, it has

$$\begin{aligned} & \lim_{T_{ijk} \rightarrow 0} \frac{\eta M_{ijk} - 1 + \sqrt{(\eta M_{ijk} - 1)^2 + 4\eta T_{ijk}\bar{q}_{ijk}}}{2\eta T_{ijk}} \\ &= \frac{\eta M_{ijk} - 1}{2\eta T_{ijk}} + \frac{|\eta M_{ijk} - 1|}{2\eta T_{ijk}} + \bar{q}_{ijk}. \end{aligned} \quad (28)$$

If  $\eta M_{ijk} > 1$ , the result is  $(\eta M_{ijk} - 1) / (\eta T_{ijk}) + \bar{q}_{ijk}$ ; else if  $\eta M_{ijk} < 1$ , the result is  $\bar{q}_{ijk}$ . That means when  $T_{ijk}$  is very small, whether the existence of the data case  $(U_i = \mathbf{u}_j, X_i = x_k)$  will be ignored or not, is depend on the choice of  $\eta$  and how many times the case appears in the sample ( $M_{ijk}$ ). And if the case never appears in the sample, the parameter  $q_{ijk}$  remains unchanged.

For the parameters of multinomial distributions, if the case  $(U_i = \mathbf{u}_j, X_i = x_k)$  does not exist in the data sample, the parameter  $\theta_{ijkn}$  remains unchanged.

In fact if the state does not transit to  $x_n$ , by computing (27), it keeps the old value. So the whole update rules are:

$$\tilde{q}_{ijk} = \begin{cases} \frac{\eta M_{ijk} - 1 + \sqrt{(\eta M_{ijk} - 1)^2 + 4\eta T_{ijk} \bar{q}_{ijk}}}{2\eta T_{ijk}} & M_{ijk} \neq 0, \\ \bar{q}_{ijk} & M_{ijk} = 0. \end{cases} \quad (29)$$

$$\tilde{\theta}_{ijkn} = \begin{cases} \bar{\theta}_{ijkn} \cdot \frac{e^{\eta M_{ijkn}/\bar{\theta}_{ijkn}}}{\sum_n \bar{\theta}_{ijkn} e^{\eta M_{ijkn}/\bar{\theta}_{ijkn}}} & M_{ijkn} \neq 0, \\ \bar{\theta}_{ijkn} & M_{ijkn} = 0. \end{cases} \quad (30)$$

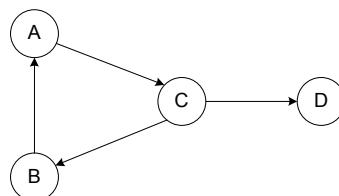
Since the update rules are derived from the relative entropy distance, following [6], we call it  $EG(\eta)$  algorithm for CTBN.

The learning rate  $\eta$  is a very important parameter to the update rules. It controls how much we rely on the old values of variables. In (4) we only set  $\eta > 0$ . As  $\eta$  approaches zero, the parameters of the model change slowly from the previous model. In fact we'll have  $\lim_{\eta \rightarrow 0} \tilde{q}_{ijk} = \bar{q}_{ijk}$ , and  $\lim_{\eta \rightarrow 0} \tilde{\theta}_{ijkn} = \bar{\theta}_{ijkn}$  according to the rules. As  $\eta$  approaches 1, the old values are weighted less, and the update of the parameters is based more on the present data. If  $\eta > 1$ , it uses the sample data to extrapolate the direction of update. Although this may works well in some problems, generally we do not take it into account. For the online learning problems,  $\eta$  will not be a fixed constant. At the beginning, the update should based more on sample data. As the model becomes well learned, it should not be changed too much. So  $\eta$  will change from 1 to 0+, as more and more data are used for the online learning.

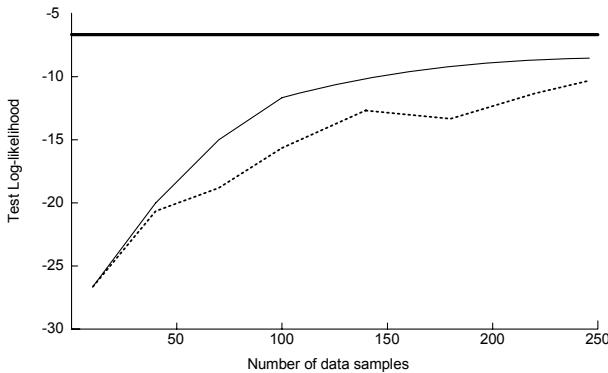
## 4 Experiments and Conclusion

We test the performance of our  $EG(\eta)$  algorithm on an example CTBN model. This model contains 4 nodes and 4 arcs with a directed cycle. Fig. 1 shows the structure of the example model.

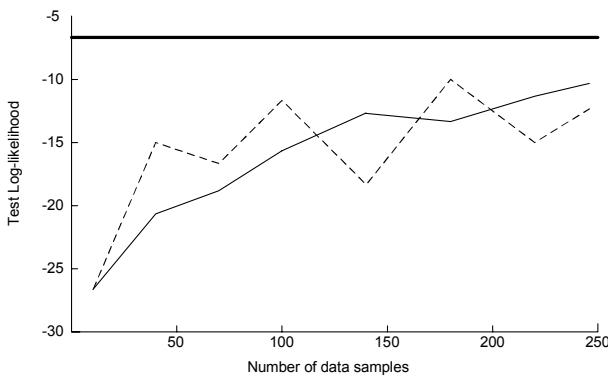
We first compare the algorithm with the MLE parameter learning algorithm in [10], with increasing amount of sample data. The difference between the 2 algorithms is that the MLE algorithm combines all the current and history data samples together to get the parameters, while the  $EG(\eta)$  algorithm learns them based on the learned model and current data. So  $EG(\eta)$  has much less information. The result is shown in Fig. 2. The thick line is the true parameters.



**Fig. 1.** The example CTBN structure



**Fig. 2.** Performance of  $EG(\eta)$  algorithm, in comparison with MLE



**Fig. 3.** Performance of  $EG(\eta)$  algorithm with different values of  $\eta$

The solid line is the parameters learned from MLE, and the thick dashed line is from  $EG(\eta)$ . We can see that  $EG(\eta)$  works well, and does not degrade much in performance comparing to MLE. For computation time,  $EG(\eta)$  needs more time to compute than MLE with the same amount of data, but it's updating from the old values other than learning from beginning. It is used when data can't be collected at one time.

Another test is comparison between different values of  $\eta$ . If  $\eta$  is small, it updates slowly, but changes smoother, and yields to the true parameters eventually. If  $\eta$  is near 1, it updates fast at first, but is very sensitive to the current sample, and oscillates much. Fig. 3 shows the result. The solid and long-dashed lines are  $\eta = 0.05$  and  $\eta = 0.2$  respectively.

As a conclusion, we present a method to update parameters in the continuous time Bayesian network model in this paper. This is a recently developed dynamic graphical model that promised to be widely used. We focus on the updating process, since many applications require tuning the existing parameters of a model

rather than constructing a new one. We derive the algorithm of update rules from a basic framework which takes a balance between an existing model and new data samples. This algorithm can be used to adjust the CTBN parameters, or for online parameter learning. Our future work is using the method in applications of practical dynamic systems.

## Acknowledgements

This paper is supported by the Shanghai Science and Technology Development Foundation under Grant No. 03DZ15027 and the National Natural Science Foundation of China under Grant No. 60173033.

## References

1. Aoki, M.: State space modeling of time series. Springer, 1987
2. Bauer, E., Koller, D., Singer, Y.: Update rules for parameter estimation in bayesian networks. In Proceedings of the 13th Conference on Uncertainty in Artificial Intelligence (UAI-97): 3-13, 1997
3. Blossfeld, H.-P., Rohwer, G.: Tech. of event history modeling. Mahwah, New Jersey, Lawrence Erlbaum Associates, 1995
4. Cohen, I., Bronstein, A.: Online Learning of Bayesian Networks. HP-Labs Tech report, HPL-2001-55TR1, 2001
5. Dean, T., Kanazawa, K.: A model for reasoning about persistence and causation. Computational Intelligence, 5, 142-150, 1989
6. Helmbold, D.P., Schapire, R.E., Singer, Y., Warmuth, M.K.: A comparison of new and old algorithms for a mixture estimation problem. Machine Learning, 27:97-119, 1997
7. Jordan, M.I., Weiss, Y.: Graphical models: Probabilistic inference. In M. Arbib (Ed.), The Handbook of Brain Theory and Neural Networks, 2nd edition. Cambridge, MA: MIT Press, 2002
8. Lando, D.: On Cox processes and credit risky securities. Review of Derivatives Research, 2, 99-120, 1998
9. Nodelman, U., Shelton, C. R., Koller, D.: Continuous time Bayesian networks. in Proceedings of the 18th Conference on Uncertainty in Artificial Intelligence (UAI-02):378-387, 2002
10. Nodelman, U., Shelton, C. R., Koller, D.: Learning continuous time Bayesian networks. in Proceedings of the 19th Conference on Uncertainty in Artificial Intelligence (UAI-03):451-458, 2003
11. Nodelman, U., Shelton, C. R., Koller, D.: Expectation maximization and complex duration distributions for continuous time Bayesian networks. in Proceedings of the 21st Conference on Uncertainty in Artificial Intelligence (UAI-05), 2005

# On Constructing Fibred Tableaux for BDI Logics

Vineet Padmanabhan and Guido Governatori

School of ITEE, The University of Queensland, Australia  
{vnair, guido}@itee.uq.edu.au

**Abstract.** In [11,13] we showed how to combine propositional BDI logics using Gabbay's *fibring* methodology. In this paper we extend the above mentioned works by providing a tableau-based decision procedure for the combined/fibred logics. We show how to uniformly construct a tableau calculus for the combined logic using Governatori's labelled tableau system **KEM**.

## 1 Introduction

The BDI model [15] is a rich and powerful logical framework developed in the early 90's focusing on three components of an agent: beliefs, desires and intentions. Its language is a propositional modal language with three families of modal operators  $B_i, D_i, I_i, i \in Agents$ . The logic **KD45** of modal logic is used for the belief operator  $B$  and the logic **KD** for  $D$  and  $I$  respectively. The semantics is given through standard Kripke frames by defining three families of accessibility relations,  $(b_i, d_i, i_i)$  for belief, desire and intention. The binary relations  $b_i$  are Euclidean, transitive and serial whereas the relations for intention  $i_i$  and desires  $d_i$  are serial. In addition to the above representation, the BDI framework impose constraints on beliefs, desires and intentions in the form of *interaction axioms* like,  $I(\varphi) \rightarrow D(\varphi), D(\varphi) \rightarrow B(\varphi)$ : intentions are stronger than desires and desires than beliefs. Hence the basic BDI logic  $\mathbb{L}$  is the combination of different component logics plus the two interaction axioms as given below

$$\mathbb{L} \equiv (\otimes_{i=1}^n \mathbf{KD45}_{B_i}) \otimes (\otimes_{i=1}^n \mathbf{KD}_{D_i}) \otimes (\otimes_{i=1}^n \mathbf{KD}_{I_i}) + \{I_i\varphi \rightarrow D_i\varphi\} + \{D_i\varphi \rightarrow B_i\varphi\} \quad (1)$$

Any BDI theory, or for that matter any fully-fledged Multi-Agent-System (MAS) theory, modelling rational agents consists of a combined system of logic of beliefs, desires, goals and intentions as mentioned above. They are basically well understood standard modal logics *combined together* to model different facets of the agents. A number of researchers have provided such combined systems for different reasons and different applications. However, investigations into a general methodology for combining the different logics involved has been mainly neglected to a large extent. Recently [11,13] *fibring/dovetailing* [7] has been advanced as a semantic methodology to characterise BDI logics. Here we extend our previous work to provide a tableau decision procedure for the fibred logic based on the labelled tableau system **KEM** [9,8,1].

The key feature of our tableau system is that it is neither based on resolution nor on standard sequent/tableau techniques. It combines linear tableau expansion rules with natural deduction rules and an analytic version of the cut rule. The tableau rules are supplemented with a powerful and flexible label algebra that allows the system to deal with a large class of intensional logics admitting possible world semantics (non-normal

modal logic [10], multi-modal logics [9] and conditional logics [2]). The algebra simulates the possible world semantics and it is very strongly related with fibring [8].

As far as the field of *combining logics* is concerned, it has been an active research area for some time now and powerful results about the preservation of important properties of the logics being combined has been obtained [12,4,16,17]. Also, investigations related to using fibring as a combining technique in various domains has produced a wealth of results [7,18,5]. The novelty of combining logics is the aim to develop *general techniques* that allow us to produce combinations of *existing* and well understood logics. Such general techniques are needed for formalising complex systems in a systematic way. Such a methodology can help decompose the problem of designing a complex system into developing components (logics) and combining them.

The next section provides a brief introduction to the technique of fibring. Section 3 is divided into several sections describing the label formalism, unification, inference rules and proof search of the **KEM** tableau system, all within the context of a Multi-Agent-Scenario. The paper concludes with some final remarks.

## 2 Fibring BDI Logics

The basic BDI logic  $\mathbb{L}$  given in (1) is defined from three component logics, **KD45**<sub>n</sub> for belief, and **KD**<sub>n</sub> for desires and intentions. For sake of clarity, consider just two of the component logics,  $\blacktriangledown_1(\mathbf{KD45})$  and  $\blacktriangledown_2(\mathbf{KD})$  and their corresponding languages  $\mathcal{L}_{\blacktriangledown_1}, \mathcal{L}_{\blacktriangledown_2}$  built from the respective sets  $\mathbb{Q}_1$  and  $\mathbb{Q}_2$  of atoms having classes of models  $\mathfrak{M}_{\blacktriangledown_1}, \mathfrak{M}_{\blacktriangledown_2}$  and satisfaction relations  $\models_1$  and  $\models_2$ . Hence we are dealing with two different systems  $S_1$  and  $S_2$  characterised, respectively, by the class of Kripke models  $\mathcal{K}_1$  and  $\mathcal{K}_2$ . For instance, we know how to evaluate  $\Box_1\varphi(B(\varphi))$  in  $\mathcal{K}_1(\mathbf{KD45})$  and  $\Box_2\varphi(D(\varphi))$  in  $\mathcal{K}_2(\mathbf{KD})$ . We need a method for evaluating  $\Box_1$  (resp.  $\Box_2$ ) with respect to  $\mathcal{K}_2$  (resp.  $\mathcal{K}_1$ ). To do so, we link (fibre), via a *fibring* function the model for  $\blacktriangledown_1$  with a model for  $\blacktriangledown_2$  and build a fibred model of the combination. The fibring function evaluates (give yes/no) answers with respect to a modality in  $S_2$ , being in  $S_1$  and vice versa. The interpretation of a formula  $\varphi$  of the combined language in the fibred model at a state  $w$  can be given as  $w \models \varphi$  if and only if  $\mathbf{f}(w) \models^* \varphi$ , where  $\mathbf{f}$  is a fibring function mapping a world to a model *suitable for interpreting*  $\varphi$  and  $\models^*$  is the corresponding satisfaction relation ( $\models_1$  for  $\blacktriangledown_1$  or  $\models_2$  for  $\blacktriangledown_2$ ). For example, let  $\blacktriangledown_1, \blacktriangledown_2$  be two modal logics as given above and  $\varphi = \Box_1 \Diamond_2 p_0$  be a formula on a world  $w_0$  of the fibred semantics.  $\varphi$  belongs to the language  $\mathcal{L}_{(1,2)}$  as the outer connective ( $\Box_1$ ) belongs to the language  $\mathcal{L}_1$  and the inner connective ( $\Diamond_2$ ) belongs to the language  $\mathcal{L}_2$ . By the standard definition we start evaluating  $\Box_1$  of  $\Box_1 \Diamond_2$  at  $w_0$ . According to the standard definition we have to check whether  $\Diamond_2 p_0$  is true at every  $w_1$  accessible from  $w_0$  since from the point of view of  $\mathcal{L}_1$  this formula has the form  $\Box_1 p$  (where  $p = \Diamond_2 p_0$  is atomic). But at  $w_1$  we cannot interpret the operator  $\Diamond_2$ , because we are in a model of  $\blacktriangledown_1$ , not of  $\blacktriangledown_2$ . To evaluate this we need the fibring function  $\mathbf{f}$  which at  $w_1$  points to a world  $v_0$ , a world in a model suitable to interpret formulae from  $\blacktriangledown_2$ . Now all we have to check is whether  $\Diamond_2 p_0$ , is true at  $v_0$  in this last model and this can be done in the usual way. Hence the fibred semantics for the combined language  $\mathcal{L}_{(1,2)}$  has models of the form  $(\mathcal{F}_1, w_1, v_1, \mathbf{f}_1)$ ,

where  $\mathcal{F}_1 = (\mathcal{W}_1, \mathcal{R}_1)$  is a frame, and  $\mathbf{f}_1$  is the fibring function which associates a model  $\mathfrak{M}_w^2$  from  $\mathcal{L}_2$  with  $w$  in  $\mathcal{L}_1$  i.e.  $\mathbf{f}_1(w) = \mathfrak{M}_w^2$ .

Let  $\mathbf{I}$  be a set of labels representing the modal operators for the intentional states (belief, desire, intention) for a set of agents, and  $\nabla_i, i \in \mathbf{I}$  be modal logics whose respective modalities are  $\square_i, i \in \mathbf{I}$ .

**Definition 1** [7] A fibred model is a structure  $(\mathcal{W}, \mathcal{S}, \mathcal{R}, \mathbf{a}, \mathbf{v}, \tau, \mathbf{F})$  where

- $\mathcal{W}$  is a set of possible worlds;
- $\mathcal{S}$  is a function giving for each  $w$  a set of possible worlds,  $\mathcal{S}^w \subseteq \mathcal{W}$ ;
- $\mathcal{R}$  is a function giving for each  $w$ , a relation  $\mathcal{R}^w \subseteq \mathcal{S}^w \times \mathcal{S}^w$ ;
- $\mathbf{a}$  is a function giving the actual world  $\mathbf{a}^w$  of the model labelled by  $w$ ;
- $\mathbf{v}$  is an assignment function  $\mathbf{v}^w(\mathbf{q}_0) \subseteq \mathcal{S}^w$ , for each atomic  $\mathbf{q}_0$ ;
- $\tau$  is the semantical identifying function  $\tau : \mathcal{W} \rightarrow \mathbf{I}$ .  $\tau(w) = i$  means that the model  $(\mathcal{S}^w, \mathcal{R}^w, \mathbf{a}^w, \mathbf{v}^w)$  is a model in  $\mathcal{K}_i$ , we use  $\mathcal{W}_i$  to denote the set of worlds of type  $i$ ;
- $\mathbf{F}$ , is the set of fibring functions  $\mathbf{f} : \mathbf{I} \times \mathcal{W} \mapsto \mathcal{W}$ . A fibring function  $\mathbf{f}$  is a function giving for each  $i$  and each  $w \in \mathcal{W}$  another point (actual world) in  $\mathcal{W}$  as follows:

$$\mathbf{f}_i(w) = \begin{cases} w & \text{if } w \in \mathcal{S}^{\mathfrak{M}} \text{ and } \mathfrak{M} \in \mathcal{K}_i \\ \text{a value in } \mathcal{W}_i, & \text{otherwise} \end{cases}$$

such that if  $w \neq w'$  then  $\mathbf{f}_i(w) \neq \mathbf{f}_i(w')$ . It should be noted that fibring happens when  $\tau(w) \neq i$ . Satisfaction is defined as follows with the usual boolean connections:

$$\begin{aligned} w \models \mathbf{q}_0 &\text{ iff } \mathbf{v}(w, \mathbf{q}_0) = 1, \text{ where } \mathbf{q}_0 \text{ is an atom} \\ w \models \square_i \varphi &\text{ iff } \begin{cases} w \in \mathfrak{M} \text{ and } \mathfrak{M} \in \mathcal{K}_i \text{ and } \forall w' (w \mathcal{R} w' \rightarrow w' \models \varphi), \text{ or} \\ w \in \mathfrak{M}, \text{ and } \mathfrak{M} \notin \mathcal{K}_i \text{ and } \forall \mathbf{f} \in \mathbf{F}, \mathbf{f}_i(w) \models \square_i \varphi. \end{cases} \end{aligned}$$

We say the model satisfies  $\varphi$  iff  $w_0 \models \varphi$ .

A fibred model for  $\nabla_{\mathbf{I}}$  can be generated from fibring the semantics for the modal logics  $\nabla_i, i \in \mathbf{I}$ . The detailed construction is given in [13]. Also, to accommodate the interaction axioms specific constraints need to be given on the fibring function. In [11] we outline the specific conditions required on the fibring function to accommodate axiom schemas of the type  $G^{a,b,c,d}$  ( $\diamond_a \square_b \varphi \rightarrow \square_c \diamond_d \varphi$ ). We do not want to get into the details here as the main theme of this paper is with regard to tableau decision procedures for fibred logics. Notice, however, that the fibring construction given in [11,13] works for normal (multi-)modal logics as well as non-normal modal logics.

### 3 Labelled Tableau for Fibred BDI Logic

In this section we show how to adapt **KEM**, a labelled modal tableau system, to deal with the fibred combination of BDI logics. A tableau system is a semantic based refutation method that systematically tries to build a (counter-) model for a set of formulas. A failed attempt to refute (invalidate) a set of formulas generates a model where the

set of formulas is true. To show that a property  $P$  follows from a theory (set of formulas/axioms)  $A_1, \dots, A_n$  we verify whether a model for  $\{A_1, \dots, A_n, \neg P\}$  exists. If it does not then  $P$  is a consequence of the theory.

In labelled tableau systems, the object language is supplemented by labels meant to represent semantic structures (possible worlds in the case of modal logics). Thus the formulas of a labelled tableau system are expressions of the form  $A : i$ , where  $A$  is a formula of the logic and  $i$  is a label. The intuitive interpretation of  $A : i$  is that  $A$  is true at (the possible world(s) denoted by)  $i$ . **KEM**'s inferential engine is based on a combination of standard tableau linear expansion rules and natural deduction rules supplemented by an analytic version of the cut rule. In addition it utilises a sophisticated but powerful label formalism that enables the logic to deal with a large class of modal and non-classical logics. Furthermore the label mechanism corresponds to fibring and thus it is possible to define tableau systems for multi-modal logic by a seamless combination of the (sub)tableaux systems for the component logics of the combination. It is not possible in this paper to give a full presentation of **KEM** for fully fledged BDI logic supplemented with interaction axioms as given in (1) (for a comprehensive presentation see [8]). Accordingly we will limit ourselves to a single modal operator for each agent and show how to characterise the axioms corresponding to each individual agent as well as the interactions between different agents with the help of an example.

### 3.1 Label Formalism

**KEM** uses *Labelled Formulas* ( $L$ -formulas for short), where an  $L$ -formula is an expression of the form  $A : i$ , where  $A$  is a wff of the logic, and  $i$  is a label. For fibred BDI logic we need to have labels for various modalities (belief, desire, intention) for each agent. However, as we have just explained we will consider only one modality and thus will have only labels for the agents. The set of atomic labels,  $\mathfrak{I}_1$ , is then given as  $\mathfrak{I}_1 = \bigcup_{i \in \text{Agt}} \Phi^i$ , where  $\text{Agt}$  is the set of agents. Every  $\Phi^i$  is partitioned into (non-empty) sets of variables and constants:  $\Phi^i = \Phi_V^i \cup \Phi_C^i$  were  $\Phi_V^i = \{W_1^i, W_2^i, \dots\}$  and  $\Phi_C^i = \{w_1^i, w_2^i, \dots\}$ .  $\Phi_C$  and  $\Phi_V$  denote the set of constants and the set of variables. We also add a set of auxiliary un-indexed atomic labels,  $\Phi^A = \Phi_V^A \cup \Phi_C^A$  where  $\Phi_V^A = \{W_1, W_2, \dots\}$  and  $\Phi_C^A = \{w_1, w_2, \dots\}$ , that will be used in unifications and proofs. A label  $u \in \mathfrak{I}_1$  is either (i) an element of the set  $\Phi_C$ , or (ii) an element of the set  $\Phi_V$ , or (iii) a path term  $(u', u)$  where (iiiia)  $u' \in \Phi_C \cup \Phi_V$  and (iiib)  $u \in \Phi_C$  or  $u = (v', v)$  where  $(v', v)$  is a label. As an intuitive explanation, we may think of a label  $u \in \Phi_C$  as denoting a world (a *given* one), and a label  $u \in \Phi_V$  as denoting a set of worlds (*any* world) in some Kripke model. A label  $u = (v', v)$  may be viewed as representing a path from  $v$  to a (set of) world(s)  $v'$  accessible from  $v$  (the world(s) denoted by  $v$ ). For any label  $u = (v', v)$  we shall call  $v'$  the *head* of  $u$ ,  $v$  the *body* of  $u$ , and denote them by  $h(u)$  and  $b(u)$  respectively. If  $b(u)$  denotes the body of  $u$ , then  $b(b(u))$  will denote the body of  $b(u)$ , and so on. We call each of  $b(u)$ ,  $b(b(u))$ , etc., a *segment* of  $u$ . The length of a label  $u$ ,  $\ell(u)$ , is the number of atomic labels in it.  $s^n(u)$  will denote the segment of  $u$  of length  $n$  and we shall use  $h^n(u)$  as an abbreviation for  $h(s^n(u))$ . Notice that  $h(u) = h^{\ell(u)}(u)$ . Let  $u$  be a label and  $u'$  an atomic label. For any label  $u$ ,  $\ell(u) > n$ , we define the *counter-segment-n* of  $u$ , as follows (for  $n < k < \ell(u)$ ):

$$c^n(u) = h(u) \times (\cdots \times (h^k(u) \times (\cdots \times (h^{n+1}(u), w_0))))$$

where  $w_0$  is a dummy label, i.e., a label not appearing in  $u$  (the context in which such a notion occurs will tell us what  $w_0$  stands for). The counter-segment- $n$  defines what remains of a given label after having identified the segment of length  $n$  with a ‘dummy’ label  $w_0$ . In the context of fibring  $w_0$  can be thought of as denoting the actual world obtained via the fibring function from the world denoted by  $s^{n(u)}$ .

So far we have provided definitions about the structure of the labels without regard to the elements they are made of. The following definitions will be concerned with the type of world symbols occurring in a label. We say that a label  $u$  is *i-preferred* iff  $h(u) \in \Phi^i$ ; a label  $u$  is *i-pure* iff each segment of  $u$  of length  $n > 1$  is *i*-preferred.

### 3.2 Label Unifications

The basic mechanism of **KEM** is its logic dependent label unification. In the same way as each modal logic is characterised by a combination of modal axioms (or semantic conditions on the model), **KEM** defines a unification for each modality and axiom/semantic condition and then combines them in a recursive and modular way. In particular we use what we call unification to determine whether two labels can be mapped to the same possible world in the possible worlds semantics. The second key issue is the ability to split labels and to work with parts of labels. The mechanism permits the encapsulation of operations on sub-labels. This is an important feature that, in the present context, allows us to correlate unifications and fibring functions. Given the modularity of the approach the first step of the construction is to define unifications (pattern matching for labels) corresponding to the single modality in the logic we want to study. Every unification is built from a basic unification defined in terms of a substitution  $\rho : \mathfrak{I}_1 \mapsto \mathfrak{I}$  such that:  $\Phi_V^i \mapsto \mathfrak{I}^i$  for every  $i \in \text{Agt}$ ,  $\Phi_V^A \mapsto \mathfrak{I}$ . Accordingly we have that two atomic labels  $u$  and  $v$   $\sigma$ -unify iff there is a substitution  $\rho$  such that  $\rho(u) = \rho(v)$ . We shall use  $[u; v]\sigma$  both to indicate that there is a substitution  $\rho$  for  $u$  and  $v$ , and the result of the substitution. The  $\sigma$ -unification is then extended to composite labels:

$$[i; j]\sigma = k \text{ iff } \exists \rho : h(k) = \rho(h(i)) = \rho(h(j)) \text{ and } b(k) = [b(i); b(j)]\sigma$$

Clearly  $\sigma$  is symmetric, i.e.,  $[u; v]\sigma$  iff  $[v; u]\sigma$ . Moreover this definition offers a flexible and powerful mechanism: it allows for an independent computation of the elements of the result of the unification.

### 3.3 An Example (Friends Puzzle [3])

Consider the agents Peter, John and Wendy with modalities  $\square_p, \square_j$ , and  $\square_w$ . John and Peter have an appointment. Suppose that Peter knows the time of appointment. Peter knows that John knows the place of their appointment. Wendy knows that if Peter knows the time of appointment, then John knows that too (since John and Peter are friends). Peter knows that if John knows the place and the time of their appointment, then John knows that he has an appointment. Peter and John satisfy the axioms T and 4. Also, if Wendy knows something then Peter knows the same thing (suppose Wendy is Peter’s wife) and if Peter knows that John knows something then John knows

1. $\square_p \text{ptime}$	$A_1 T_p : \square_p \varphi \rightarrow \varphi$
2. $\square_p \square_j \text{place}$	$A_2 4_p : \square_p \varphi \rightarrow \square_p \square_p \varphi$
3. $\square_w (\square_p \text{ptime} \rightarrow \square_j \text{time})$	$A_3 T_j : \square_j \varphi \rightarrow \varphi$
4. $\square_p \square_j (\text{place} \wedge \text{time} \rightarrow \text{appt})$	$A_4 4_j : \square_j \varphi \rightarrow \square_j \square_j \varphi$
	$A_5 I_{wp} : \square_w \varphi \rightarrow \square_p \varphi$
	$A_6 S_{pj} : \square_p \square_j \varphi \rightarrow \square_j \square_p \varphi$

**Fig. 1.** Knowledge base related to the Friend's puzzle

that Peter knows the same thing. The Knowledge/belief base of the example is given in Fig.1. So we have a modal language consisting of three modalities  $\square_p$ ,  $\square_j$  and  $\square_w$  denoting respectively the agents Peter, John and Wendy and characterised by the set  $A = \{A_i \mid i = 1, \dots, 6\}$  of interaction axioms. Suppose now that one wants to show that each of the friends knows that the other one knows that he has an appointment, i.e,

$$\square_j \square_p \text{appointment} \wedge \square_p \square_j \text{appointment} \quad (2)$$

In other words one want to show that (2) is a theorem of the knowledge-base. The tableau proof of (2) using the **KEM** tableau procedure is given in Fig.3. But before getting into the proof details we should understand how the label unification introduced in the previous section works for the modal operators,  $\square_w$ ,  $\square_j$  and  $\square_p$ . We can capture the relationship between  $\square_w$  and  $\square_p$  by extending the substitution  $\rho$  by allowing a variable of type  $w$  to be mapped to labels of the same type and of type  $p$ .

$$\rho^w(W^w) \in \mathfrak{S}^w \cup \mathfrak{S}^p$$

Then the unification  $\sigma^w$  is obtained from the basic unification  $\sigma$  by replacing  $\rho$  with the extended substitution  $\rho^w$ . This procedure must be applied to all pairs of modalities  $\square_1, \square_2$  related by the interaction axiom  $\square_1 \varphi \rightarrow \square_2 \varphi$ . For the unifications for  $\square_p$  and  $\square_j$  ( $\sigma^p$  and  $\sigma^j$ ) we assume that the labels involved are  $i$ -pure. First we notice that these two modal operators are **S4** modalities and thus have to use the unification for this logic.

$$[u; v] \sigma^{S4} = \begin{cases} [u; v] \sigma^D & \text{if } \ell(u) = \ell(v) \\ [u; v] \sigma^T & \text{if } \ell(u) < \ell(v), h(u) \in \Phi_C \\ [u; v] \sigma^4 & \text{if } \ell(u) < \ell(v), h(u) \in \Phi_V \end{cases} \quad (3)$$

It is worth noting that the conditions on axiom unifications are needed in order to provide a deterministic unification procedure. The  $\sigma^T$  and  $\sigma^4$  are defined as follows: (It should be noted that for the rest of the unifications, given two labels  $u$  and  $v$  we will assume that  $\ell(u) > \ell(v)$ . The conditions for  $\ell(v) > \ell(u)$  are symmetric). Thus,

$$\begin{aligned} [u; v] \sigma^T &= [s^{\ell(v)}(u); v] \sigma \text{ if } \forall n \geq \ell(v), [h^n(u); h(v))] \sigma = [h(u); h(v)] \sigma \\ [u; v] \sigma^4 &= c^{\ell(v)}(u) \text{ if } h(v) \in \Phi_V \text{ and } w_0 = [s^{\ell(v)}(u); v] \sigma \end{aligned}$$

$\sigma^T$  allows us to unify two labels such that the segment of the longest with the length of the other label and the other label unify, provided that all remaining elements of the longest have a common unification with the head of the shortest. For example let

$u = (w_3, (W_2, (w_2, w_1)))$  and  $v = (w_3, (W_1, w_1))$ . Here  $[W_2; w_3]\sigma = [w_3; w_3]\sigma$  and the two labels  $\sigma^T$ -unify to  $(w_3, (w_2, w_1))$ . This means that after a given point the head of the shortest is always included in its extension, and thus it is accessible from itself, and consequently we have reflexivity. For  $\sigma^4$  we have that the shortest label unifies with the segment with the same length of the longest and that the head of the shortest is variable. A variable stands for all worlds accessible from its predecessor. Thus, given transitivity, every element extending the segment with length of the shortest is accessible from this point. Then a unification corresponding to axiom  $A_6$  from Figure 1 is

$$[u; v]\sigma^{S_{p,j}} = \begin{cases} c^{m+n}(v) \text{ if } h(u) \in \Phi_V^j \text{ and } c^n(v) \text{ is } p\text{-pure, and} \\ \quad h^{\ell(u)-1}(u) \in \Phi_V^p \text{ and } c^n(v) \text{ is } j\text{-pure, and} \\ \quad w_0 = [s^{\ell(u)-2}(u); s^m(v)]\sigma \end{cases}$$

This unification allows us to unify two labels when in one label we have a sequence of a variable of type  $p$  followed by a variable of type  $j$  and a label where we have a sequence of labels of type  $j$  followed by a sequence of labels of type  $p$ . The unification for  $\square_p$  and  $\square_j$  are just the combination of the three unifications given above. Finally the unification for the logic  $\mathbf{L}$  defined by the axioms  $A_1 - A_6$  is obtained from the following recursive unification

$$[u; v]\sigma_L = \begin{cases} [u; v]\sigma^{w,p,j} \\ [c^m(u); c^n(v)]\sigma^{w,p,j} \text{ where } w_0 = [s^m(u); s^n(v)]\sigma_L \end{cases}$$

$\sigma^{w,p,j}$  is the simple combination of the unifications for the three modal operators. Having accounted for the label unification we now give the inference rules used in **KEM** proofs.

### 3.4 Inference Rules

For the inference rules we use the Smullyan-Fitting unifying notation [6] (Figure.2). The  $\alpha$ -rules are just the familiar linear branch-expansion rules of the tableau method. The  $\beta$ -rules are nothing but natural inference patterns such as Modus Ponens, Modus Tollens and Disjunctive syllogism generalised to the modal case. To apply such rules the labels of the premises must unify and the label of the conclusion is the result of the

$\frac{\alpha : u}{\alpha_1 : u} (\alpha)$ $\alpha_2 : u$ $A : u$	$\frac{\beta : u}{\beta_i^c : v} (i = 1, 2)$ $\frac{\beta_{3-i}^c : [u; v]\sigma}{\beta_{3-i}^c : [u; v]\sigma} (\beta)$	$\frac{v^i : u}{v_0^i : (W_n^i, u)} \clubsuit (v)$ $\frac{\pi^i : u}{\pi_0^i : (w_n^i, u)} \clubsuit (\pi)$
$\frac{}{} A : u \quad   \quad \neg A : u$	$\frac{}{} \frac{\neg A : v}{\times} [\text{ if } [u; v]\sigma] \text{ (PNC)}$	$(\clubsuit) W_n^i, w_n^i \text{ are new labels.}$

**Fig. 2.** Inference Rules of **KEM** using the Smullyan-Fitting Notation

unification. The  $v$  and  $\pi$  rules are the normal expansion rules for modal operators of labelled tableaux with free variable. The intuition for the  $v$  rule is that if  $\Box_i A$  is true at  $u$ , then  $A$  is true at all worlds accessible via  $R_i$  from  $u$ , and this is the interpretation of the label  $(W_n^i, u)$ ; similarly if  $\Box_i A$  is false at  $u$  (i.e.,  $\neg\Box_i A$  is true), then there must be a world, let us say  $w_n^i$  accessible from  $u$ , where  $\neg A$  is true. A similar intuition holds when  $u$  is not  $i$ -preferred, but the only difference is that we have to make use of the fibring function instead of the accessibility relation. PB represents the semantic counterpart of the cut rule of the sequent calculus (intuitive meaning: a formula  $A$  is either true or false in any given world). Accordingly it is a zero-premise inference rule, so in its unrestricted version can be applied whenever we like. However, we impose a restriction on its application. PB can be only applied w.r.t. immediate sub-formulas of unanalysed  $\beta$ -formulas, that is  $\beta$  formulas for which we have no immediate sub-formulas with the appropriate labels in the tree. PNC states that two labelled formulas are  $\sigma_L$ -complementary when the two formulas are complementary and their labels  $\sigma_L$ -unify.

### 3.5 Proof Search

Let  $\Gamma = \{X_1, \dots, X_m\}$  be a set of formulas. Then  $\mathcal{T}$  is a **KEM-tree** for  $\Gamma$  if there exists a finite sequence  $(\mathcal{T}_1, \mathcal{T}_2, \dots, \mathcal{T}_n)$  such that (i)  $\mathcal{T}_1$  is a 1-branch tree consisting of  $\{X_1 : t_1, \dots, X_m : t_m\}$ ; (ii)  $\mathcal{T}_n = \mathcal{T}$ , and (iii) for each  $i < n$ ,  $\mathcal{T}_{i+1}$  results from  $\mathcal{T}_i$  by an application of a rule of **KEM**. A branch  $\theta$  of a **KEM-tree**  $\mathcal{T}$  of  $L$ -formulas is said to be  $\sigma_L$ -closed if it ends with an application of *PNC*, open otherwise. As usual with tableau methods, a set  $\Gamma$  of formulas is checked for consistency by constructing a **KEM-tree** for  $\Gamma$ . Moreover we say that a formula  $A$  is a **KEM-consequence** of  $\Gamma$  ( $\Gamma \vdash_{\mathbf{KEM}(L)} A$ ) if a **KEM-tree** for  $\{X_1 : u_1, \dots, X_n : u_n, \neg A : v\}$  is closed using the unification for the logic  $L$ , where  $v \in \Phi_C^A$ , and  $u_i \in \Phi_V^A$ . The intuition behind this definition is that  $A$  is a consequence of  $\Gamma$  when we take  $\Gamma$  as a set of global assumptions [6], i.e., true in every world in a Kripke model.

We now describe a systematic procedure for **KEM**. First we define the following notions. Given a branch  $\theta$  of a **KEM-tree**, we shall call an  $L$ -formula  $X : u$  *E-analysed* in  $\theta$  if either (i)  $X$  is of type  $\alpha$  and both  $\alpha_1 : t$  and  $\alpha_2 : u$  occur in  $\theta$ ; or (ii)  $X$  is of type  $\beta$  and one of the following conditions is satisfied: (a) if  $\beta_1^C : v$  occurs in  $\theta$  and  $[u; v]\sigma$ , then also  $\beta_2 : [u; v]\sigma$  occurs in  $\theta$ , (b) if  $\beta_2^C : v$  occurs in  $\theta$  and  $[u; v]\sigma$ , then also  $\beta_1 : [u; v]\sigma$  occurs in  $\theta$ ; or (iii)  $X$  is of type  $\mu$  and  $\mu_0 : (u', u)$  occurs in  $\theta$  for some appropriate  $u'$  of the right type, not previously occurring in  $\theta$ . We shall call a branch  $\theta$  of a **KEM-tree** *E-completed* if every  $L$ -formula in it is *E-analysed* and it contains no complementary formulas which are not  $\sigma_L$ -complementary. We shall say a branch  $\theta$  of a **KEM-tree** *completed* if it is *E-completed* and all the  $L$ -formulas of type  $\beta$  in it are either analysed or cannot be analysed. We shall call a **KEM-tree** *completed* if every branch is completed.

The following procedure starts from the 1-branch, 1-node tree consisting of  $\{X_1 : u, \dots, X_m : v\}$  and applies the inference rules until the resulting **KEM-tree** is either closed or completed. At each stage of proof search (i) we choose an open non completed branch  $\theta$ . If  $\theta$  is not *E-completed*, then (ii) we apply the 1-premise rules until  $\theta$  becomes *E-completed*. If the resulting branch  $\theta'$  is neither closed nor completed, then (iii) we apply the 2-premise rules until  $\theta$  becomes *E-completed*. If the resulting branch

1. $\mathbf{F} \Box_j \Box_p appt$	$w_0$	9. $\mathbf{T}(place \wedge time \rightarrow appt)$ ( $W_1^j, W_1^p, w_0$ )
2. $\mathbf{T} \Box_p \Box_j (place \wedge time \rightarrow appt)$	$W_0$	10. $\mathbf{F} place \wedge time$ ( $W_1^p, w_1^j, w_0$ )
3. $\mathbf{T} \Box_w (\Box_p time \rightarrow \Box_j time)$	$W_0$	11. $\mathbf{T} \Box_p time \rightarrow \Box_j time$ ( $W_1^w, w_0$ )
4. $\mathbf{T} \Box_p \Box_j place$	$W_0$	12. $\mathbf{T} \Box_j place$ ( $W_2^p, w_0$ )
5. $\mathbf{T} \Box_p time$	$W_0$	13. $\mathbf{T} place$ ( $W_2^j, W_2^p, w_0$ )
6. $\mathbf{F} \Box_p appt$	( $w_1^j, w_0$ )	14. $\mathbf{F} time$ ( $w_1^p, w_1^j, w_0$ )
7. $\mathbf{F} appt$	( $w_1^p, w_1^j, w_0$ )	15. $\mathbf{T} \Box_p time$ ( $w_1^j, w_0$ )
8. $\mathbf{T} \Box_j (place \wedge time \rightarrow appt)$	( $W_1^p, w_0$ )	16. $\mathbf{T} time$ ( $W_3^p, w_1^j, w_0$ )
		×

**Fig. 3.** Proof of  $\Box_j \Box_p$  using **KEM** representation

$\theta'$  is neither closed nor completed, then (iv) we choose an  $L$ -formula of type  $\beta$  which is not yet analysed in the branch and apply  $PB$  so that the resulting  $LS$ -formulas are  $\beta_1 : u'$  and  $\beta_1^C : u'$  (or, equivalently  $\beta_2 : u'$  and  $\beta_2^C : u'$ ), where  $u = u'$  if  $u$  is restricted (and already occurring when  $h(u) \in \Phi_C$ ), otherwise  $u'$  is obtained from  $u$  by instantiating  $h(u)$  to a constant not occurring in  $u$ ; (v) (“Modal  $PB$ ”) if the branch is not  $E$ -completed nor closed, because of complementary formulas which are not  $\sigma_L$ -complementary, then we have to see whether a restricted label unifying with both the labels of the complementary formulas occurs previously in the branch; if such a label exists, or can be built using already existing labels and the unification rules, then the branch is closed, (vi) we repeat the procedure in each branch generated by  $PB$ . Fig.3. shows a **KEM** tableaux proof using the inference rules in Fig.2. and following the proof search mentioned above to solve the first conjunct of (2).

The proof goes as follows; 1. is the negation of the formula to be proved. The formulas in 2–5 are the global assumptions of the scenario and accordingly they must hold in every world of every model for it. Hence we label them with a variable  $W_0$  that can unify with every other label. This is used to derive 12. from 11. and 5. using a  $\beta$ -rule, and for introducing 15.; 6. is from 1., and 7. from 6. by applying  $\pi$  rule. Similarly we get 8. from 2., 9. from 8. using  $\nu$  rule. 10. comes from 9. and 7. through the use of modus tollens. Applying  $\nu$  rule twice we can derive 11. from 3. as well as 13. from 12. Through propositional reasoning we get 14. from 10. and by a further use of  $\nu$  rule on 15. we get 16. (14. and 16.) are complementary formulas indicating a contradiction and this results in a closed tableaux because the labels in 14. and 16. unify, denoting that the contradiction holds *in the same world*.

## 4 Conclusions and Related Work

In this paper we have argued that BDI logics can be explained in terms of fibring as a combination of simpler modal logics. We then outlined a decision procedure based on the **KEM** tableaux system showing the correlation between **KEM** unification and fibring. To evaluate the features of our tableau system we demonstrated how it can deal with a multi-agent scenario like the Friend’s puzzle.

Elsewhere [14] we have shown why other labelled tableaux approaches (both *path* & *graph*) are not suited for fibring. The path approach (where prefixes are sequences of

integers representing a world as a path in the model connecting the initial world to the one at hand) requires the definition of new inference rules for each logic with a simple labelling mechanism. It is not clear how the path-based approach can be extended to more complex cases of fibring, for example when we consider non-normal modal operators for the mental attitudes of the agents. The graph approach (where accessibility relations are given explicitly) on the other hand does not require, in general, any new rule, since it uses the semantic structure to propagate formulas to the appropriate labels. It is then suitable for an approach based on fibring, since the relationships between two labels can be given in terms of fibring. However, when the structure of the model is more complicated (for example when the models for the logics are given in terms of neighbourhood models) then the approach might not be applicable since it assumes relationships between labels/worlds in a model and not more complex structures. In addition, the system does not give a decision procedure unless the relationships among labels are restricted to decidable fragments of first-order logic. Thus it is not possible to represent logic that are not first-order definable and the designer of an agent logic has to verify that she is operating within a decidable fragment of first order logic.

**KEM**, similar to the graph approach, does not need logic dependent rules, however, similar to the path approach, it needs logic dependent label unifications. The label algebra can be seen as a form of fibring [8], thus simple fibring does not require special attention; therefore **KEM** allows for a seamless composition of (sub)tableaux for modal logics. The label algebra contrary to the graph reasoning mechanism is not based on first order logic and thus can deal with complex structure and is not limited to particular fragments. Indeed **KEM** has been used with complex label schema for non-normal modal logics in a uniform way [10] as well as other intensional logics such as conditional logics [2]. For these reasons we believe that **KEM** offers a suitable framework for decision procedure for multi-modal logic for MAS. As we only described the static fragment of BDI logics (no temporal evolution was considered) the future work is to extend the tableaux framework so as to accommodate temporal modalities.

**Acknowledgements.** This work was supported by the Australian Research Council (ARC) under Discovery Project No. DP0452628 on "Combining Modal Logics for Dynamic and Multi-Agent Systems".

## References

1. A. Artosi, P. Benassi, G. Governatori, and A. Rotolo. Shakespearian modal logic: A labelled treatment of modal identity. In *Advances in Modal Logic*, volume 1. CSLI, 1998.
2. A. Artosi, G. Governatori, and A. Rotolo. Labelled tableaux for non-monotonic reasoning: Cumulative consequence relations. *Journal of Logic and Computation*, 12(6):1027–1060, 2002.
3. M. Baldoni. *Normal Multimodal Logics: Automatic Deduction and Logic Programming Extension*. PhD thesis, Universita degli Studi di, Italy, 1998.
4. P. Blackburn and M. Rijke. Zooming in, zooming out. *Journal of Logic, Language and Information*, 1996.
5. A. S. d' Avila Garcez and D. M. Gabbay. Fibring neural networks. In *American National Conference on Artificial Intelligence (AAAI-2004)*, pages 342–347. AAAI/MIT Press, 2004.

6. M. Fitting. *Proof Methods for Modal and Intuitionistic Logics*. Reidel, Dordrecht, 1983.
7. D. M. Gabbay. *Fibring Logics*. Oxford University Press, Oxford, 1999.
8. Dov M. Gabbay and Guido Governatori. Fibred modal tableaux. Kluwer Publishers, 2000.
9. G. Governatori. Labelled tableau for multi-modal logics. In *TABLEAUX*, volume 918, pages 79–94. Springer, 1995.
10. G. Governatori and A. Luppi. Labelled tableaux for non-normal modal logics. In *AI\*IA 99*, number 1792 in LNAI, pages 119–130. Springer-Verlag, 2000.
11. G. Governatori, V. Padmanabhan, and A. Sattar. On Fibring Semantics for BDI Logics. In *Logics in Artificial Intelligence (JELIA-02)*, Italy, LNAI-2424. Springer, 2002.
12. M. Kracht and F. Wolter. Properties of independently axiomatizable bimodal logics. *The Journal of Symbolic Logic*, 56(4):1469–1485, 1991.
13. V. Padmanabhan. *On Extending BDI Logics*. PhD thesis, School of Information Technology, Griffith University, Brisbane, Australia, 2003.
14. V. Padmanabhan and G. Governatori. A fibred tableaux calculi for BDI logics. To be published in the proceedings of DALT-06 (Declarative Agent Languages & Technologies).
15. A. S. Rao and M. P. Georgeff. Formal models and decision procedures for multi-agent systems. Technical note 61, Australian Artificial Intelligence Institute, 1995.
16. A. Sernadas, C. Sernadas, and M. Ryan. Combining logics: Synchronising and fibring. Technical report, Department of Mathematics, Instituto superior Tecnico, Lisbon, Portugal, 1996.
17. F. Wolter. The decision problem for combined (modal) logics. Technical report, Universitat Leipzig, Germany, [www.informatik.uni-leipzig.de/~wolter/](http://www.informatik.uni-leipzig.de/~wolter/), September 9, 1999.
18. A. Zanardo, A. Sernadas, and C. Sernadas. Fibring: Completeness preservation. *Journal of Symbolic Logic*, 66(1):414–439, 2001.

# The Representation of Multiplication Operation on Fuzzy Numbers and Application to Solving Fuzzy Multiple Criteria Decision Making Problems

Chien-Chang Chou

Department of Transportation and Navigation Science, National Taiwan Ocean University  
2, Pei-Ning Road, Keelung 202, Taiwan, Republic of China

&

Department of Shipping Technology, National Kaohsiung Marine University  
482, Chung-Chou 3 Road, Chi-Chin 805, Kaohsiung, Taiwan, Republic of China  
[chiensch1@ms58.hinet.net](mailto:chiensch1@ms58.hinet.net)

**Abstract.** This paper proposes the canonical representation of multiplication operation on trapezoidal fuzzy numbers using the  $L^{-1}-R^{-1}$  Inverse Function Arithmetic Representation method. Finally, the canonical representation proposed in this paper is applied to solve a fuzzy multiple criteria decision making problem of selection of plant location.

## 1 Introduction

In the past, most papers discussed the representation of one fuzzy number [1, 2, 3, 5, 7, 8, 9, 10, 11, 12, 14, 15, 16]. Few papers [4, 6, 13] discussed the representation of multiplication operation on two fuzzy numbers. In fact, the representation of multiplication operation on two fuzzy numbers is useful for decision makers to solve fuzzy multiple criteria decision making problems. This paper proposes the canonical representation of multiplication operation on two trapezoidal fuzzy numbers using the  $L^{-1}-R^{-1}$  Inverse Function Arithmetic Representation method. In Section 2, the representation of one trapezoidal fuzzy number is introduced. Section 3 presents the canonical representation of multiplication operation on two trapezoidal fuzzy numbers using the  $L^{-1}-R^{-1}$  Inverse Function Arithmetic Representation method and the representation proposed in this paper is applied to solve a fuzzy multiple criteria decision making problem of selection of plant location in Section 4. Finally, the conclusion is given in Section 5.

## 2 The Representation of One Fuzzy Number

The representation of fuzzy number is often used to ranking fuzzy numbers or fuzzy rating of alternatives. For example, the representation of fuzzy number is useful for the decision maker to select the best alternative from several candidate alternatives under fuzzy multiple criteria decision making environment. The decision maker can rank quickly the rank ordering of each candidate alternative using the representation of fuzzy number. In 1998, Chen and Hsieh [3] proposed the graded mean integration representation method for representing fuzzy numbers. Now we introduce briefly the graded mean integration representation method.

Suppose  $A=(c, a, b, d)$  is a fuzzy number as Figure 1. The membership function of A is as follows.

$$f_A(x) = \begin{cases} \frac{(x - c)}{(a - c)}, & c \leq x \leq a \\ 1, & a \leq x \leq b \\ \frac{(x - d)}{(b - d)}, & b \leq x \leq d \\ 0, & \text{otherwise} \end{cases}$$

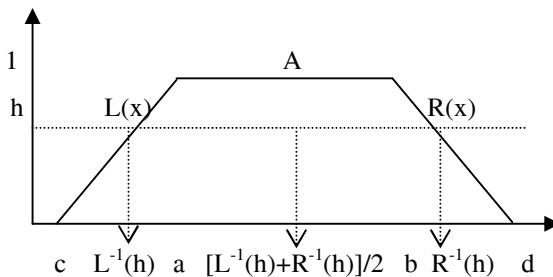
$$\text{Since } L_A(x) = \frac{(x - c)}{(a - c)}, \quad c \leq x \leq a, \quad R_A(x) = \frac{(x - d)}{(b - d)}, \quad b \leq x \leq d,$$

$$\text{and } L^{-1}_A(h) = c + (a - c)h, \quad 0 \leq h \leq 1, \quad R^{-1}_A(h) = d + (b - d)h, \quad 0 \leq h \leq 1.$$

$L_A(x)$  and  $R_A(x)$  are the function L and the function R of fuzzy number A, respectively.  $L^{-1}_A(h)$  and  $R^{-1}_A(h)$  are the inverse functions of function  $L_A(x)$  and function  $R_A(x)$  at h-level, respectively. Then the graded mean h-level value of the fuzzy number A is  $[L^{-1}_A(h) + R^{-1}_A(h)]/2$  as Figure 1. The graded mean integration representation of A is

$$\begin{aligned} P(A) &= \int_0^1 \frac{h[L^{-1}_A(h) + R^{-1}_A(h)]}{2} dh / \int_0^1 h dh \\ &= \int_0^1 \frac{h[c + (a - c)h + d + (b - d)h]}{2} dh / \int_0^1 h dh \\ &= \frac{1}{6}(c + 2a + 2b + d) \end{aligned} \quad (1)$$

Formula (1) is equal to the formula proposed by Delgado, Vila, and Voxman et al. [5].



**Fig. 1.** The graded mean h-level value of fuzzy number  $A=(c, a, b, d)$

### 3 The Inverse Function Arithmetic Representation for Multiplication Operation

Based on the above graded mean integration representation method, this paper further proposes the representation of multiplication operation on two trapezoidal

fuzzy numbers using the  $L^{-1}$ - $R^{-1}$  Inverse Function Arithmetic Presentation method as follows.

Suppose the membership function of fuzzy number  $A_1=(c_1, a_1, b_1, d_1)$  is

$$f_{A_1}(x) = \begin{cases} \frac{(x - c_1)}{(a_1 - c_1)}, & c_1 \leq x \leq a_1 \\ 1, & a_1 \leq x \leq b_1 \\ \frac{(x - d_1)}{(b_1 - d_1)}, & b_1 \leq x \leq d_1 \\ 0, & \text{otherwise} \end{cases}$$

$$\text{Since } L_{A_1}(x) = \frac{(x - c_1)}{(a_1 - c_1)}, \quad c_1 \leq x \leq a_1, \quad R_{A_1}(x) = \frac{(x - d_1)}{(b_1 - d_1)}, \quad b_1 \leq x \leq d_1,$$

$$\text{and } L^{-1}_{A_1}(h) = c_1 + (a_1 - c_1)h, \quad 0 \leq h \leq 1, \quad R^{-1}_{A_1}(h) = d_1 + (b_1 - d_1)h, \quad 0 \leq h \leq 1.$$

$L_{A_1}(x)$  and  $R_{A_1}(x)$  are the function L and the function R of fuzzy number  $A_1$  respectively.  $L^{-1}_{A_1}(h)$  and  $R^{-1}_{A_1}(h)$  are the inverse functions of function  $L_{A_1}(x)$  and function  $R_{A_1}(x)$  at h-level, respectively.

Similarly, suppose the membership function of fuzzy number  $A_2=(c_2, a_2, b_2, d_2)$  is

$$f_{A_2}(x) = \begin{cases} \frac{(x - c_2)}{(a_2 - c_2)}, & c_2 \leq x \leq a_2 \\ 1, & a_2 \leq x \leq b_2 \\ \frac{(x - d_2)}{(b_2 - d_2)}, & b_2 \leq x \leq d_2 \\ 0, & \text{otherwise} \end{cases}$$

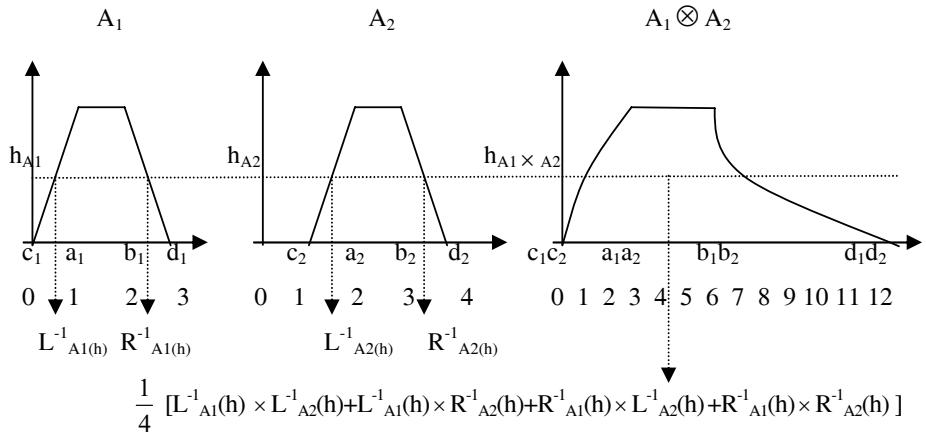
$$\text{Since } L_{A_2}(x) = \frac{(x - c_2)}{(a_2 - c_2)}, \quad c_2 \leq x \leq a_2, \quad R_{A_2}(x) = \frac{(x - d_2)}{(b_2 - d_2)}, \quad b_2 \leq x \leq d_2,$$

$$\text{and } L^{-1}_{A_2}(h) = c_2 + (a_2 - c_2)h, \quad 0 \leq h \leq 1, \quad R^{-1}_{A_2}(h) = d_2 + (b_2 - d_2)h, \quad 0 \leq h \leq 1.$$

$L_{A_2}(x)$  and  $R_{A_2}(x)$  are the function L and the function R of fuzzy number  $A_2$  respectively.  $L^{-1}_{A_2}(h)$  and  $R^{-1}_{A_2}(h)$  are the inverse functions of function  $L_{A_2}(x)$  and  $R_{A_2}(x)$  at h-level, respectively.

*Definition 1.* Let  $A_1=(c_1, a_1, b_1, d_1)$  and  $A_2=(c_2, a_2, b_2, d_2)$  be two trapezoidal fuzzy numbers as figure 2. The representation of multiplication operation on the two fuzzy numbers at h-level is

$$A_{1(h)} \otimes A_{2(h)} = \frac{1}{4} [L^{-1}_{A_1}(h) \times L^{-1}_{A_2}(h) + L^{-1}_{A_1}(h) \times R^{-1}_{A_2}(h) + R^{-1}_{A_1}(h) \times L^{-1}_{A_2}(h) + R^{-1}_{A_1}(h) \times R^{-1}_{A_2}(h)]$$



**Fig. 2.** The Inverse Function Arithmetic Representation for multiplication operation

*Definition 2.* Let  $P(A_1 \otimes A_2)$  be the representation of  $A_1 \otimes A_2$ ,

$$\begin{aligned} P(A_1 \otimes A_2) = & \int_0^1 \int_0^1 \frac{1}{4} \{ [h_{A1} L^{-1}_{A1}(h) \times h_{A2} L^{-1}_{A2}(h)] + [h_{A1} L^{-1}_{A1}(h) \times h_{A2} R^{-1}_{A2}(h)] \\ & + [h_{A1} R^{-1}_{A1}(h) \times h_{A2} L^{-1}_{A2}(h)] + [h_{A1} R^{-1}_{A1}(h) \times h_{A2} R^{-1}_{A2}(h)] \} dh_{A1} dh_{A2} / (\int_0^1 h_{A1} dh_{A1} \times \int_0^1 h_{A2} dh_{A2}) \end{aligned} \quad (2)$$

Now we can compute the representation of multiplication operation on two trapezoidal fuzzy numbers.

By formula (2), the representation of  $A_1 \otimes A_2$  is

$$\begin{aligned} P(A_1 \otimes A_2) = & \int_0^1 \int_0^1 \frac{1}{4} \{ [h_{A1} L^{-1}_{A1}(h) \times h_{A2} L^{-1}_{A2}(h)] + [h_{A1} L^{-1}_{A1}(h) \times h_{A2} R^{-1}_{A2}(h)] \\ & + [h_{A1} R^{-1}_{A1}(h) \times h_{A2} L^{-1}_{A2}(h)] + [h_{A1} R^{-1}_{A1}(h) \times h_{A2} R^{-1}_{A2}(h)] \} dh_{A1} dh_{A2} / (\int_0^1 h_{A1} dh_{A1} \times \int_0^1 h_{A2} dh_{A2}) \\ = & \int_0^1 \int_0^1 \frac{1}{4} \{ h_{A1}[c_1 + (a_1 - c_1)h_{A1}] \times h_{A2}[c_2 + (a_2 - c_2)h_{A2}] + h_{A1}[c_1 + (a_1 - c_1)h_{A1}] \times h_{A2}[d_2 + (b_2 - d_2)h_{A2}] \\ & + h_{A1}[d_1 + (b_1 - d_1)h_{A1}] \times h_{A2}[c_2 + (a_2 - c_2)h_{A2}] + h_{A1}[d_1 + (b_1 - d_1)h_{A1}] \times h_{A2}[d_2 + (b_2 - d_2)h_{A2}] \} \\ & \times dh_{A1} dh_{A2} / (\int_0^1 h_{A1} dh_{A1} \times \int_0^1 h_{A2} dh_{A2}) \\ = & \int_0^1 \frac{1}{4} \{ [\frac{1}{2} h_{A1}^2 c_1 + \frac{1}{3} h_{A1}^3 (a_1 - c_1)] \times h_{A2}[c_2 + (a_2 - c_2)h_{A2}] + \\ & [\frac{1}{2} h_{A1}^2 c_1 + \frac{1}{3} h_{A1}^3 (a_1 - c_1)] \times h_{A2}[d_2 + (b_2 - d_2)h_{A2}] + \\ & [\frac{1}{2} h_{A1}^2 d_1 + \frac{1}{3} h_{A1}^3 (b_1 - d_1)] \times h_{A2}[c_2 + (a_2 - c_2)h_{A2}] + \end{aligned}$$

$$\begin{aligned}
& \left[ \frac{1}{2} h_{A_1}^2 d_1 + \frac{1}{3} h_{A_1}^3 (b_1 - d_1) \right] \times h_{A_2} [d_2 + (b_2 - d_2) h_{A_2}] \} \Big|_0^1 dh_{A_2} \\
& / \left( \frac{1}{2} h_{A_1}^2 \Big|_0^1 \times \int_0^1 h_{A_2} dh_{A_2} \right) \\
= & \int_0^1 \frac{1}{4} \left\{ \left[ \frac{1}{2} c_1 + \frac{1}{3} (a_1 - c_1) \right] \times h_{A_2} [c_2 + (a_2 - c_2) h_{A_2}] + \right. \\
& \left[ \frac{1}{2} c_1 + \frac{1}{3} (a_1 - c_1) \right] \times h_{A_2} [d_2 + (b_2 - d_2) h_{A_2}] + \\
& \left[ \frac{1}{2} d_1 + \frac{1}{3} (b_1 - d_1) \right] \times h_{A_2} [c_2 + (a_2 - c_2) h_{A_2}] + \\
& \left. \left[ \frac{1}{2} d_1 + \frac{1}{3} (b_1 - d_1) \right] \times h_{A_2} [d_2 + (b_2 - d_2) h_{A_2}] \right\} dh_{A_2} / \left( \frac{1}{2} \times \int_0^1 h_{A_2} dh_{A_2} \right) \\
= & \frac{1}{4} \left\{ \left[ \frac{1}{2} c_1 + \frac{1}{3} (a_1 - c_1) \right] \times \left[ \frac{1}{2} h_{A_2}^2 c_2 + \frac{1}{3} h_{A_2}^3 (a_2 - c_2) \right] + \right. \\
& \left[ \frac{1}{2} c_1 + \frac{1}{3} (a_1 - c_1) \right] \times \left[ \frac{1}{2} h_{A_2}^2 d_2 + \frac{1}{3} h_{A_2}^3 (b_2 - d_2) \right] + \\
& \left[ \frac{1}{2} d_1 + \frac{1}{3} (b_1 - d_1) \right] \times \left[ \frac{1}{2} h_{A_2}^2 c_2 + \frac{1}{3} h_{A_2}^3 (a_2 - c_2) \right] + \\
& \left. \left[ \frac{1}{2} d_1 + \frac{1}{3} (b_1 - d_1) \right] \times \left[ \frac{1}{2} h_{A_2}^2 d_2 + \frac{1}{3} h_{A_2}^3 (b_2 - d_2) \right] \right\} \Big|_0^1 / \left( \frac{1}{2} \times \frac{1}{2} h_{A_2}^2 \Big|_0^1 \right) \\
= & \frac{1}{4} \left\{ \left[ \frac{1}{2} c_1 + \frac{1}{3} (a_1 - c_1) \right] \times \left[ \frac{1}{2} c_2 + \frac{1}{3} (a_2 - c_2) \right] + \left[ \frac{1}{2} c_1 + \frac{1}{3} (a_1 - c_1) \right] \times \left[ \frac{1}{2} d_2 + \frac{1}{3} (b_2 - d_2) \right] + \right. \\
& \left[ \frac{1}{2} d_1 + \frac{1}{3} (b_1 - d_1) \right] \times \left[ \frac{1}{2} c_2 + \frac{1}{3} (a_2 - c_2) \right] + \left[ \frac{1}{2} d_1 + \frac{1}{3} (b_1 - d_1) \right] \times \left[ \frac{1}{2} d_2 + \frac{1}{3} (b_2 - d_2) \right] \} / \left( \frac{1}{2} \times \frac{1}{2} \right) \\
= & \frac{1}{6} (c_1 + 2a_1 + 2b_1 + d_1) \times \frac{1}{6} (c_2 + 2a_2 + 2b_2 + d_2)
\end{aligned}$$

We have that

$$P(A_1 \otimes A_2) = \frac{1}{6} (c_1 + 2a_1 + 2b_1 + d_1) \times \frac{1}{6} (c_2 + 2a_2 + 2b_2 + d_2) \quad (3)$$

$$\text{By formula (1), } P(A_1) = \frac{1}{6} (c_1 + 2a_1 + 2b_1 + d_1), P(A_2) = \frac{1}{6} (c_2 + 2a_2 + 2b_2 + d_2)$$

$$P(A_1) \times P(A_2) = \frac{1}{6} (c_1 + 2a_1 + 2b_1 + d_1) \times \frac{1}{6} (c_2 + 2a_2 + 2b_2 + d_2)$$

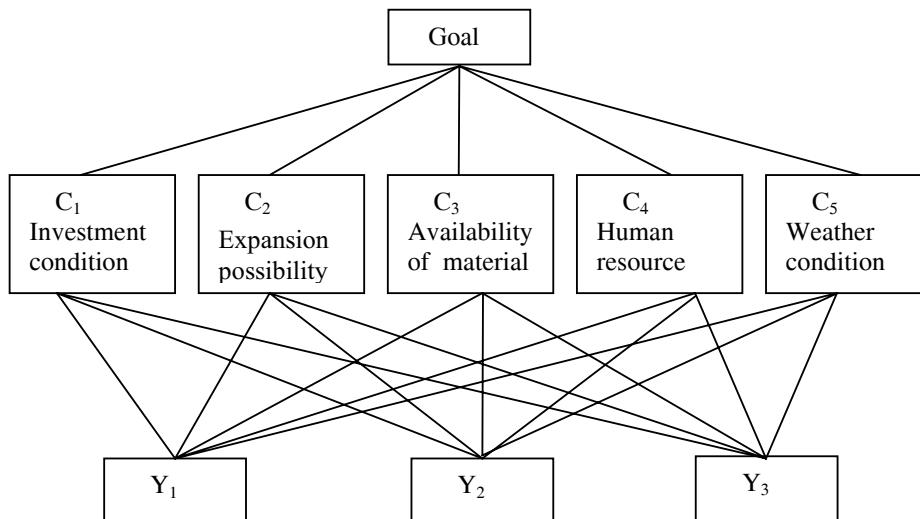
Thus, we also have that

$$P(A_1 \otimes A_2) = P(A_1) \times P(A_2).$$

#### 4 A Numerical Example of Fuzzy Multiple Criteria Decision Making

A manufacturing company desires to select a suitable city for establishing a new factory. After preliminary screening, three candidate cities  $Y_1$ ,  $Y_2$  and  $Y_3$  remain for further evaluation. The company considers five criteria to select the most suitable

candidate. Five criteria are the investment condition ( $C_1$ ), expansion possibility ( $C_2$ ), availability of required material ( $C_3$ ), human resource ( $C_4$ ), and weather condition ( $C_5$ ), respectively. The decision-making hierarchical structure of this location selection problem is shown as Figure 3.



**Fig. 3.** The decision-making hierarchy structure of location selection

The proposed representation of multiplication operation on two fuzzy numbers in this paper is currently applied to solve this location selection problem. The data shown in the following tables are given. The computational procedure is summarized as follows.

Step 1. The decision maker uses the linguistic weighting variables shown in Table 1 to assess the importance of the criteria and present it in Table 2.

**Table 1.** Linguistic variables for the importance weight of each criteria

Very Low	(0.0, 0.05, 0.1, 0.15)
Low	(0.1, 0.2, 0.3, 0.4)
Medium Low	(0.3, 0.4, 0.5, 0.6)
Fair	(0.4, 0.5, 0.6, 0.7)
Medium High	(0.5, 0.6, 0.7, 0.8)
High	(0.6, 0.7, 0.8, 0.9)
Very High	(0.7, 0.8, 0.9, 1.0)

**Table 2.** The importance weight of each criteria

$C_1$	Fair
$C_2$	High
$C_3$	Fair
$C_4$	Very High
$C_5$	Very Low

Step 2. The decision maker uses the linguistic rating variables shown in Table 3 to evaluate the rating of candidate alternatives with respect to each criteria and present it in Table 4.

**Table 3.** Linguistic variables for the preference of each alternative

Very Poor	(0, 1, 1.5, 2)
Poor	(1, 1.5, 2, 2.5)
Medium Poor	(2, 2.5, 3, 3.5)
Fair	(3.5, 4.5, 5.5, 6.5)
Medium Good	(6.5, 7, 7.5, 8)
Good	(7.5, 8, 8.5, 9)
Very Good	(8, 8.5, 9, 10)

**Table 4.** The rating of each candidate alternative with respect to each criteria

C <sub>1</sub>	Y <sub>1</sub>	Medium Poor
	Y <sub>2</sub>	Very Good
	Y <sub>3</sub>	Good
C <sub>2</sub>	Y <sub>1</sub>	Good
	Y <sub>2</sub>	Very Good
	Y <sub>3</sub>	Medium Good
C <sub>3</sub>	Y <sub>1</sub>	Fair
	Y <sub>2</sub>	Good
	Y <sub>3</sub>	Medium Good
C <sub>4</sub>	Y <sub>1</sub>	Very Good
	Y <sub>2</sub>	Good
	Y <sub>3</sub>	Good
C <sub>5</sub>	Y <sub>1</sub>	Fair
	Y <sub>2</sub>	Good
	Y <sub>3</sub>	Good

Step 3. Calculate the total rating of each candidate city. Assume there are N candidate locations (Y<sub>1</sub>, Y<sub>2</sub>, Y<sub>n</sub>..., Y<sub>N</sub>), I evaluation criteria (C<sub>1</sub>, C<sub>2</sub>, C<sub>i</sub> ..., C<sub>I</sub>). Let P<sub>(Y<sub>n,Ci</sub>)</sub> be the fuzzy rating assigned to the n<sup>th</sup> candidate location by the decision maker under criteria C<sub>i</sub>. And W<sub>i</sub> is the importance weights of C<sub>i</sub>. Let P<sub>ni</sub>= P(W<sub>i</sub> ⊗ P<sub>(Y<sub>n,Ci</sub>)</sub>) be the representation of W<sub>i</sub> ⊗ P<sub>(Y<sub>n,Ci</sub>)</sub>, where 1 ≤ n ≤ N, 1 ≤ i ≤ I. And let TP<sub>n</sub>

=  $\sum_{i=1}^I P_{ni}$  be the total fuzzy rating of the n<sup>th</sup> candidate location. For example, the

total rating of candidate location Y<sub>1</sub> is

$$TP_1 = \sum_{i=1}^I P_{1i} = \sum_{i=1}^I P(W_i \otimes P_{(Y_1, Ci)})$$

$$=[(0.4, 0.5, 0.6, 0.7) \otimes (2, 2.5, 3, 3.5) + (0.6, 0.7, 0.8, 0.9) \otimes (7.5, 8, 8.5, 9) + (0.4, 0.5, 0.6, 0.7) \otimes (3.5, 4.5, 5.5, 6.5) + (0.7, 0.8, 0.9, 1.0) \otimes (8, 8.5, 9, 10) + (0.0, 0.05, 0.1, 0.15) \otimes (3.5, 4.5, 5.5, 6.5)]$$

Then according to the representation of multiplication operation on two fuzzy numbers proposed in this paper.

$$P(A_1 \otimes A_2) = \frac{1}{6} (c_1 + 2a_1 + 2b_1 + d_1) \times \frac{1}{6} (c_2 + 2a_2 + 2b_2 + d_2)$$

The decision maker can obtain quickly the total rating of candidate location  $Y_1$ .  
 $TP_1=1.5125+6.1875+2.75+7.4375+0.375=18.2625$

Similarly, we can obtain easily the total rating  $TP_2=23.54375$  for candidate location  $Y_2$  and the total rating  $TP_3=21.59375$  for candidate location  $Y_3$ . Thus the decision maker of manufacturing company can rank quickly all candidate locations,  $TP_2=23.54375 > TP_3=21.59375 > TP_1=18.2625$ , and then select easily the candidate location  $Y_2$  as the best plant location.

## 5 Conclusion

The representation of multiplication operation on two fuzzy numbers is useful for decision makers who are in the fuzzy multiple criteria decision-making environment to rank all candidate alternatives and choose the best one. This paper proposes the representation of multiplication operation on two trapezoidal fuzzy numbers and then this representation is applied to solve a numerical example of plant location selection problem. Based on this representation of multiplication operation on two fuzzy numbers, decision makers can rank quickly the rank ordering of all alternatives and then choose easily the best one in the fuzzy multiple criteria decision making environment. This representation of multiplication operation on two fuzzy numbers proposed in this paper not only can be applied to solve the location selection problems but also can be applied to solve other fuzzy multiple criteria decision-making problems. The method proposed in this paper could be applied to solve other fuzzy problems such as the automated negotiation in the future work. Furthermore, the further study will proof this method is correct for multiplication operation of more than two fuzzy numbers.

## References

1. Adamo, J.M.: Fuzzy Decision Trees. *Fuzzy Sets and Systems* 4 (1980) 207-219
2. Campos, L., Verdegay, J.L.: Linear Programming Problems and Ranking of Fuzzy Numbers. *Fuzzy Sets and Systems* 32 (1989) 1-11
3. Chen, S.H., Hsieh, C.H.: Graded Mean Integration Representation of Generalized Fuzzy Number. Proceeding of 1998 sixth Conference on Fuzzy Theory and its Application (CD-ROM), filename: 031.wdl, Chinese Fuzzy Systems Association, Taiwan, Republic of China (1998) 1-6
4. Chou, C.C.: The Canonical Representation of Multiplication Operation on Triangular Fuzzy Numbers. *Computers and Mathematics with Applications* 45 (2003) 1601-1610
5. Delgado, M., Vila, M.A., Voxman, W.: On a Canonical Representation of Fuzzy Numbers. *Fuzzy Sets and Systems* 93 (1998) 125-135
6. Dubois, D., Prade, H.: Operations on Fuzzy Numbers. *Journal of Systems Sciences* 9 (1978) 613-626
7. Heilpern, S.: Representation and Application of Fuzzy Numbers. *Fuzzy Sets and Systems* 91 (1997) 259-268

8. Kaufmann A., Gupta, M.M.: Introduction to Fuzzy Arithmetic Theory and Applications. Van Nostrand Reinhold (1991)
9. Li, R.J.: Fuzzy Method in Group Decision Making. Computers and Mathematics with Applications 38 (1999) 91-101
10. Liou, T.S., Wang, M.J.: Ranking Fuzzy Numbers with Integral Values. Fuzzy Sets and Systems 50 (1992) 247-255
11. Luo, X., Jennings, N.R., Shadbolt, N., Leung, H., Lee, J.H.: A Fuzzy Constraint Based Model for Bilateral Multi-issue Negotiations in Semi-competitive Environments. Artificial Intelligence 148 (2003) 53-102
12. Ma, M., Friedman, M., Kandel, A.: A New Fuzzy Arithmetic. Fuzzy Sets and Systems 108 (1999) 83-90
13. Mizumoto, M., Tanaka, K.: The Four Operations of Arithmetic on Fuzzy Numbers. Systems Comput. Controls 7(5) (1976) 73-81
14. Nahmias, S.: Fuzzy Variables. Fuzzy Sets and Systems (1978) 97-111
15. Yager, R.R.: A Procedure for Ordering Fuzzy Subsets of the Unit Interval. Information Sciences 24 (1981) 143-161
16. Zadeh, L.A.: Fuzzy Sets. Information and Control 8 (1965) 338-353

# Finding a Natural-Looking Path by Using Generalized Visibility Graphs

Kyeonah Yu

Department of Computer Science  
Duksung Women's University  
Seoul 132-714, Korea  
kyeonah@duksung.ac.kr

**Abstract.** We propose to use the generalized visibility graph (Vgraph) to represent search space for finding a natural-looking path in computer games. The generalized Vgraph is the extension of the visibility graph to the generalized polygonal world which is produced as a result of dilating polygonal obstacles. That is, the generalized visibility graph is constructed on the expanded boundaries of obstacles so that the path keeps an amount of clearance from obstacles. We also introduce an algorithm that can efficiently incorporate a start and a goal location to the map represented in a generalized Vgraph for quick path-finding. The A\* algorithm with Euclidean distance is used for quick path-finding. The proposed approach is compared to other major search space representations analytically and empirically. The results show that the map can be generated efficiently by using the generalized Vgraph and the paths found look natural.

## 1 Introduction

In state-of-the-art games, non-player characters can move in a goal-directed manner so that they can move to the goal position without colliding obstacles. Many path-finding methods have been proposed and implemented and most game people agree that the A\* algorithm reign as the preferred path-finding algorithm [9][11]. However, a search algorithm is not all of the path-finding. How to represent the underlying search space is equally important for the performance and memory requirement of path-finding [10]. Regular grids, waypoint graphs, and navigation meshes are some major approaches to represent search space in games. A regular grid is simple but its path looks awkward due to frequent sharp turns. A waypoint graph has a reduced search space but can not guarantee the optimality. A navigation mesh is appropriate for various environments and shapes but suffers from storage of large number of polygons. Every representation method has strong and weak points in terms of the size of search space, optimality, path quality and etc. Another key consideration is the possibility of auto-generation of search space. This becomes more important as a game world is getting complex by featuring dynamic obstacles. To deal with these considerations the visibility graph can be a good candidate for representing search space and planning paths in games [11].

The visibility graph (Vgraph) is a well-known construct for planning the shortest collision-free path in Robotics. The Vgraph is an undirected graph whose nodes are vertices of obstacles and whose links are line segments connecting two nodes without intersecting obstacles. The Vgraph can automatically generate a simplified map for a game region and compute the optimal path with a proper search algorithm such as the A\* algorithm. The disadvantage is that the path found on the Vgraph keeps no clearance from the obstacles, which makes it unrealistic. Game people often call this problem the “wall-hugging” issue. This issue is due to the links defined on the obstacle boundaries. Therefore the wall-hugging can be overcome by constructing the Vgraph on the expanded obstacles instead on the original obstacles. The amount of expansion is determined by a level designer to provide sufficient clearance. The expansion is implemented as the positive solid offsetting and the expanded obstacles become locally convex generalized polygons. Then the search space is represented by a generalized visibility graph (generalized Vgraph) [6] which defines on not only line segments but also arc segments.

In this paper we show how to construct the generalized Vgraph for representing search space in a computer game world. We also present an efficient method to incorporate the user-specified start and goal positions into the generalized Vgraph so that a quick path-planning is possible. The quick incorporation of the start and the goal is important because this part is executed repeatedly once the map is built. The resulting path will look natural because paths contained on generalized Vgraphs consist of straight-lines and smooth turns. The performance of the generalized Vgraph will be compared with other representative methods in terms of the size of search space, search time, optimality, possibility of auto-generation, and path quality.

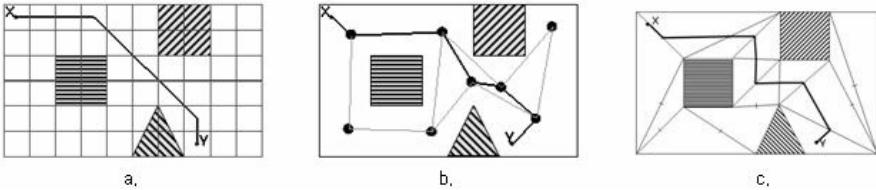
## 2 Related Work

In this section, we summarize some search space representation methods used in computer games and discuss their pros and cons: regular grids, waypoint graphs, and navigation meshes [10]. We also review the Vgraph and its related works and show how different it is from the other methods.

The search space using regular grids can be generated automatically and may be the simplest way to represent a game world. In regular grids the state space is the set of tiles and the operators are moves to the next tile in each compass direction. The movement to the four diagonal is often allowed. Typically search space is large for the fine granularity required for realism of a game. The paths found on square grids are often awkward due to frequent sharp turns (Fig. 1a). For a natural-looking path, path smoothing processing can be added in the post-processing stage. A hierarchical path-finding method can be applied to break large search space in the preprocessing stage [1][7][8].

A waypoint graph consists of nodes placed in the middle of free space and links connecting them [10]. The size of search space is small even though there is a trade-off between the size of search space and the path quality. It is easy to represent search space by a waypoint graph even for 3D game worlds because human designers place waypoints

at the game design level. However the manual node placement is a bottleneck in the process of game development because it is hard for designers at different levels to share the intention of placement and to modify search space as a game world changes. There are some variants of a waypoint graph such as a corner graph and a circle-based waypoint graph. Fig.1b shows an example of a path found on a waypoint graph.



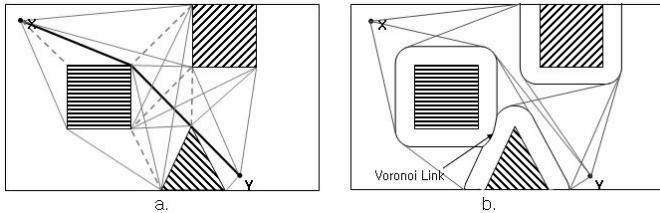
**Fig. 1.** a. A path on a grid of squares when the movements to four diagonal as well as four compass directions are allowed. b. A path found by a waypoint graph where nodes are placed by level designers. c. A path found by a triangle-based navigation mesh. These paths can be straightened and smoothed by some extra-processing.

A navigation mesh is a representation in which the nodes are convex polygons covering a game world and the edges are the links between neighbor polygons. Once the decomposition is completed, it is quite easy to find the path even for complex terrain. However it is not trivial to generate a navigation mesh itself both manually and automatically. Moreover it has the storage problem of large number of convex polygons. Triangle-based navigation meshes are the most popular but polygons with any numbers of sides can be used in modeling navigation meshes as long as they are convex polygons. Fig.1c shows a path found on a triangle-based navigation mesh when the links are defined through the center of the common edge between neighbor triangles.

The idea of the visibility test between two nodes has appeared in [4][12]. The quality of paths can be improved by straightening up multiple links. All the paths shown in fig.1 can be improved in this way. This idea can be implemented with the visibility graph (Vgraph). The Vgraph has originally developed for robot motion planning and recently gains game people's attention for path-finding with terrain analysis in Force 21 [11]. A Vgraph consists of a start node, a goal node, nodes placed at the vertices of the obstacles and links connecting them including obstacle boundaries. It is shown that the shortest path can be found with proper algorithms such as the Dijkstra algorithm or A\* (Fig. 2a). Using the reduced Vgraph is known to be enough for finding the shortest path where non-tangent links are classified as unnecessary links and removed [5].

Finding the shortest paths is a main advantage of using Vgraphs for path planning. However it causes the wall-hugging problem by which the character moves in contact with an obstacle. In this paper we suggest to construct the Vgraph on the expanded obstacles not on the original obstacles to guarantee sufficient clearance between a moving character and obstacles. The Vgraph constructed on generalized polygons has been defined in [6] and called a generalized visibility graph. It focused on the mathematical aspects of the generalized Vgraph and did not show how it can be efficiently used for path-planning. Constructing the Vgraph on the expanded obstacles can cause

the blocking of collision-free passages. Then we can add Voronoi links in blocked regions to provide the maximum clearance from obstacles which is similar to the Visibility-Voronoi complex [3]. Fig. 2b shows an example of the generalized Vgraph and the Voronoi link.



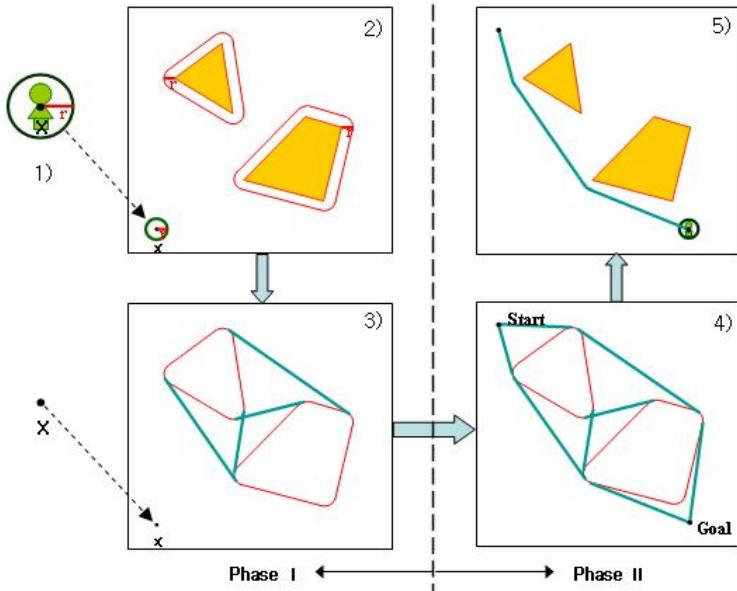
**Fig. 2.** a. A path found by a visibility graph, where plain lines indicate tangent links and dashed line non-tangent links. b. A Generalized visibility graph and a Voronoi link. Note that the triangular obstacle is moved to the left to create an example of overlapped regions.

### 3 Path-Finding Based on the Generalized Visibility Graph

To apply the Vgraph for path-planning, we first convert the original problem to the one in the configuration space(C-space) where the moving object is reduced to the point and obstacles are enlarged accordingly. The configuration of a moving character X is a point of  $R^3$ ,  $(x,y,\theta)$  when X is assumed to move in planes. The dimension of the configuration space can be reduced to two by modeling X as a circumscribed circle or larger circles. The larger radius of a circle provides the larger clearance but the more blocking regions. Fig. 3 describes the path-finding procedure. The procedure consists of four steps: Expand obstacles by  $r$ , construct the generalized Vgraph on the expanded obstacles, incorporate the start and the goal to the generalized Vgraph, and run A\*. The steps 1)-3) are executed only once for build the map. On the other hand the last two are executed repeatedly when the new start and goal locations are specified.

#### 3.1 Phase I: State Space Generation (Map Construction)

The expansion of obstacles can be implemented by positive solid offsetting, which is a special operator of the Minkowski sum. If we let  $S$  be the set representing a polygonal obstacle, the solid offsetting by  $r$  is defined as  $\{p : \exists q \in S, \|p - q\| \leq r\}$ . The implementation of the positive solid offsetting of an n-sided polygon represented in an ordered set of vertices,  $[v_0, v_1, v_2, \dots, v_{n-1}, v_n]$  ( $v_0 = v_n$ , in counterclockwise order), is as follows. Let  $n_{i-}$  and  $n_{i+}$  be the outward normal of line segments  $[v_{i-1}, v_i]$  and  $[v_i, v_{i+1}]$  respectively. If  $\theta = \text{angle}(n_{i-}, n_{i+})$  is positive,  $v_i$  is convex and is converted to the arc with the center at  $v_i$  and two endpoints extended from  $v_i$  in the direction of  $n_{i-}$  and  $n_{i+}$  by  $r$ . If  $\theta$  is negative,  $v_i$  is concave and is converted to the vertex extended from  $v_i$  in the middle direction of  $n_{i-}$  and  $n_{i+}$  by  $r/\sin(\theta/2)$ . The result of solid-offsetting polygonal obstacles is locally non-convex generalized polygons. That is, all the concave parts of the expanded obstacles are vertices which come from the intersection of line segments.



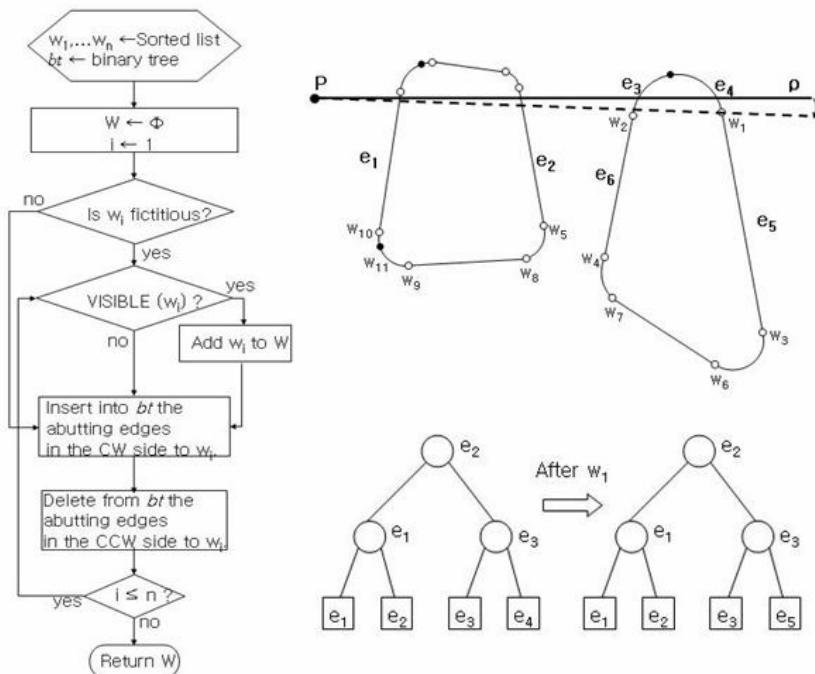
**Fig. 3.** The path-finding procedure: 1) a moving character is modeled by a circle of a radius  $r$ . 2) It is treated as a point and obstacles are enlarged by expanding their boundaries with a radius  $r$ . 3) The generalized Vgraph is constructed. 4) Start and Goal are incorporated. 5) The shortest path is found by the A\*. The path keeps the minimum clearance  $r$  from the obstacles.

To compute the generalized Vgraph, the visibility relation for generalized polygons (between a vertex and an arc or between two arcs) must be defined. Note that the expanded obstacles consist of convex arcs and concave vertices. This means that we need to consider only links between two arcs because the link to any concave vertex is non-tangent and unnecessary for finding the shortest paths. The visibility relation between two arcs is defined as follows: two arcs are visible if there exists at least one tangent segment  $[a,b]$  such that  $a$  and  $b$  are points on two arcs each and  $a$  and  $b$  are visible. Points  $a$  and  $b$  are called fictitious vertices [6]. Then the nodes of the generalized Vgraph are the vertices of the expanded obstacles and the fictitious vertices from the tangent segments to the arcs. The links of the generalized Vgraph are the obstacle boundaries and the visible links between the fictitious vertices. The fictitious vertices between two arcs are four at most. So the computation of the generalized Vgraph has the same time complexity as in the case of dealing with polygons. The naïve time complexity for computing the generalized Vgraph is  $O(n^3)$  as in the case of the Vgraph. The efficient methods for computing the Vgraph can be applied similarly and one of them using the rotational plane sweep algorithm is explained in the next section.

### 3.2 Phase II: Search (Path-Finding)

Once the generalized Vgraph on the expanded obstacles is constructed, the start  $s$  and the goal  $g$  nodes are added to find a path from  $s$  to  $g$ . This is the part executed repeatedly during runtime, so the rapid incorporation of  $s$  and  $g$  is critical in overall performance.

To find all visible arcs from  $s$  and  $g$ , we have to compute all the tangents to arcs and for every tangent we have to test whether it intersects any obstacle. This test costs  $O(n)$  for each tangent and there are  $O(n)$  tangents, leading to an  $O(n^2)$  running time where  $n$  is the number of vertices of obstacles. This running time can be improved to  $O(n \log n)$  by applying the rotational plane sweep algorithm. The main idea of the rotational plane sweep is using the information of the previous test for the rest in cyclic order. It works similar to the plane sweep paradigm used widely in computational geometry [1]. We use a rotating half-line instead of using a vertical or a horizontal line to sweep the plane. In this paper we introduce the rotational plane sweep algorithm modified for the generalized polygonal obstacles. See Fig. 4.



**Fig. 4.** The rotational plane sweep algorithm finds all the visible vertices from a point  $p$ .  $w_1, \dots, w_n$  are the sorted list in clockwise angle. If we let  $\rho$  be the half-line parallel to the positive x-axis starting at  $p$ ,  $bt$  stores the obstacle edges that are intersected by  $\rho$  in the order of distance from  $p$ .

Before sweeping starts, we first sort critical events in cyclic order. Critical events contain not only the vertices of obstacles but also the fictitious vertices which are defined as tangent points on obstacle arcs. In the example of Fig. 4 the fictitious vertices are indicated with filled dots while the ordinary vertices with blank dots. The binary tree is initialized by storing the obstacle edges which are intersected by the half-line  $\rho$  pointing into the positive x-direction. As the rotational plane sweep proceeds, this binary tree maintains the status of sweeping which is the ordered sequence of obstacle edges intersected by  $\rho$ . Fig. 4 shows the flow of the algorithm. By keeping the status in the binary tree,  $\text{VISIBLE}(w_i)$  can be checked only by examining whether

the edge in the leftmost leaf intersects with line segment  $[p, w_i]$ . The degenerate conditions such as  $p$  having multiple vertices can be handled in the same way to the polygonal cases [1]. The time complexity of this algorithm is  $O(n\log n)$ : It takes  $O(n\log n)$  to initialize the sorted list and the binary tree,  $O(\log n)$  to update the status of the binary tree, and  $O(\log n)$  to search the binary tree for intersection. The space complexity is  $O(n)$ .

The final procedure is to find the shortest path in a generalized Vgraph. We use the A\* algorithm with Euclidean distance to the goal as the heuristic. Because Euclidean distance is the straight line distance between two points, Euclidean distance is always less than or equal to the true distance of any path with multiple line and arc segments. Euclidean distance is admissible. Hence the A\* search with Euclidean distance guarantees to generate the shortest path in free space whenever a path exists and returns failure otherwise.

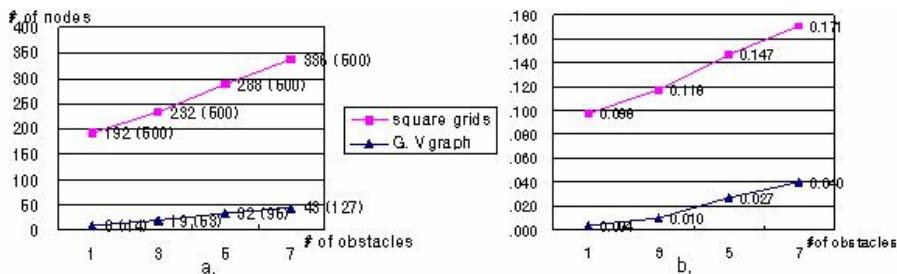
## 4 Results

The key considerations when selecting a search space representation are search speed, memory overhead, path quality, and how the search space is created. The first two are rather related because graphs with more nodes and more links will result in slower searches. We compare the proposed method to major representation methods, a regular grid, a waypoint graph, a navigation mesh, and a visibility graph. Table 1 summarizes the comparisons of the proposed method with them. The size of search space based on grids or a navigation mesh is usually large compared to a waypoint graph or a Vgraph which focus only critical points in free space. Optimality here implies the shortest path. Even though the path of the generalized Vgraph is inflated from the backbone path of the Vgraph, it can be said to be optimal among available paths satisfying the required clearance from obstacles. A waypoint graph has small search space but the hand-placed-waypoint approach limits its applications. The navigation mesh can be generated both automatically and manually and has the storage problem due to large search space. The representation methods 1),2),3) suffer from path quality and optimality issues, so the extra processing such as “string-pulling” and “smoothing” is followed by the usual path-finding to relieve these issues. The main difference of generalized Vgraphs from Vgraphs is the path quality: the path found by the generalized Vgraph has no sharp turns and provides required clearance, which makes it natural.

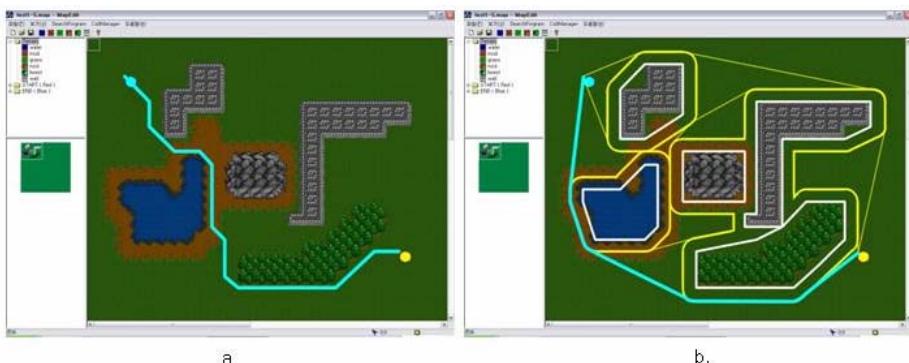
**Table 1.** Comparison of five representation methods: The generalized Vgraph has the small search space, guarantees the shortest path, and can be built automatically. The path found in the generalized Vgraph contains no sharp turns and provides moderate clearance.

Metrics Representations	Size of Search Space	Optimality (among avail- able paths)	Automatic Generation of Search Space	Path Quality (smooth turns/ clearance)
1) Grid-based	Large	No	Auto	No/Yes
2) Waypoint graph	Small	No	Manual	No/Yes
3) Nav- Mesh	Large	No	Auto&Manual	No/Yes
4) Vgraph	Small	Yes	Auto	No/No
5) Generalized Vgraph	Small	Yes	Auto	Yes/Yes

In general the size of search space is large in regular grids and a navigation mesh because the number of cells (tiles or convex polygons) must be large enough for fine granularity. It is true that the search space is steady in regular grids or a navigation mesh whereas the search space increases in a waypoint graph, a Vgraph, or a generalized Vgraph when the density of obstacles becomes large. To see how it affects the size of search space as the number of obstacles increases, we performed simulations using the real game editor. We first generated the game environment with one obstacle, represented it both in the grid-based method and in a generalized Vgraph, found a path using the A\* algorithm for each representation, and compared them in terms of the size of total search space, the number of searched nodes, and the search time. We repeated this simulation with increasing the number of obstacles. We put more obstacles in the way from the start to the goal so that the search is affected by adding obstacles. The results show that the path-finding by the generalized Vgraph examines much less nodes and is much faster as summarized in Fig. 5. The search on the generalized Vgraph explores only 9.7% of the nodes by the search on the grids and is



**Fig. 5.** Empirical results: a. comparison of the numbers of nodes in OPEN lists. The search space size is in the parenthesis. b. Comparison of the search times. Each number indicates the average of ten executions.



**Fig. 6.** Example paths: a. the path trajectory on the square grids. Some post-processing is needed for straightening and smoothing. b. the path trajectory(the bold one) on the generalized Vgraph. It looks natural with straight lines and smooth turns as generated.

6.6 times faster on the average. We can also observe that the search space of the generalized Vgraph keeps growing as the number of obstacles increases. That means there may be the case that the generalized Vgraph is not appropriate: A game layout with many tiny obstacles like a forest can be one example.

Fig. 6 shows two paths found in regular grids and generalized Vgraphs by using A\*. The path found on the square grids contains frequent angular turns when the game world is quite complicated. Therefore the post-processing for straightening and for smoothing paths must follow the regular path-finding. It is notable that visibility testing (also known as string-pulling) is a major technique to get rid of the zigzag effect of the path, which is generic in generating a generalized Vgraph. The paths in the generalized Vgraph look more natural with smooth turns and sufficient clearance.

## 5 Conclusions

The representation of search space is as important as the path-finding algorithm. In this paper we proposed to use the generalized Vgraph to create a minimal representation of the search space similar to the original Vgraph. We described how to expand polygonal obstacles boundaries by a radius of the circle which is large enough to enclose the moving object, how to construct the generalized Vgraph on the expanded obstacles, and how to incorporate the start and the goal to plan the shortest path. It has been shown that the proposed method overcame the drawbacks of earlier approaches in computer games as followings: The resulting search space is much smaller than that of the regular grids and a navigation mesh. Its generation is fully-automated whereas other major methods depend on level-designers with the generation of search space. It also guarantees to find the optimal path under the condition of providing the required amount of clearance. The empirical results showed that paths by the generalized Vgraph looked more realistic than other major representation methods.

In this paper we assumed that the naïve method to generate the Vgraph was applied to generate the generalized Vgraph even though there were many improved approaches in finding the Vgraph of a polygonal world. All of these methods cannot be applied to the generalized polygon directly. We will try to find efficient methods to build the Vgraph which can be applied to generalized polygons. One problem of expanding obstacles is that the free space may be blocked due to more clearance than enough, which causes a moving object to take a detour rather than the shortest path. This problem can be overcome by allowing movements in a blocked region with less clearance. One possible path trajectory in a blocked region is a Voronoi link which is defined as a segment of a line or a parabola equidistant to different obstacles. The idea of combining the Voronoi diagram with the Vgraph has been introduced and implemented as the VV-complex [3]. The VV-complex can provide paths with the maximal possible clearance in blocked regions where the preferred clearance can not be obtained. We plan to apply the VV-complex to the computer games and evaluate its performance.

## References

1. de Berg, M., van Kreveld, M., Overmars, M., Schwarzkorf, O.: Computational Geometry-Algorithms and Applications. 2<sup>nd</sup> edn. Springer-Verlag, Berlin Heidelberg New York (2000)
2. Botea, A., Muller, M., Schaeffer, J.: Near optimal hierarchical path-finding. Journal of game development, vol. 1(1) (2004) 1-22
3. Wein, R., Van der Berg, J.P., Halperin, D.: The Visibility-Voronoi Complex and Its Applications. In: Proc. European Workshop on Computational Geometry (2005) 151–154
4. Jung, D., Kim, H., Kim, J., Um K., Cho, H.: Efficient Path Finding in 3D Games by using Visibility Tests with the A\* Algorithm. In: Proceedings of the International Conference Artificial Intelligence and Soft Computing, Spain (2004)
5. Latombe, J.C.: Robot Motion Planning. Kluwer Academic Publishers, Boston (1991)
6. Laumond, J.P.: Obstacle growing in a nonpolygonal world. Inform. Proc. Letter, vol. 25(1) (1987) 41-50
7. Pinter, M.: Towards more realistic pathfinding. Game Developer Magazine, April (2001)
8. Rabin, S.: A\* speed optimizations and A\* Aesthetic Optimizations. In: Deloura, M. (eds.): Game Programming Gems. Charles Rive Media(2000) 264-287
9. Stout, B.: Smart moves: Intelligent path-finding. Game developer magazine, October (1996) 28-35
10. Tozour, P.: Search Space Representations. In: Rabin, S. (eds.): AI Game Programming Wisdom 2. Charles Rive Media (2004) 85-102
11. Woodcock, S.: Game AI: The state of the industry. Game Developer Magazine, August (2000)
12. Young, T.: Expanded Geometry for Points-of-Visibility Pathfinding. In: Deloura, M. (eds.): Game Programming Gems 2, Charles Rive Media (2001)

# Comparison Between Two Languages Used to Express Planning Goals: *CTL* and *EAGLE*

Wei Huang<sup>1</sup>, Zhonghua Wen<sup>1,2</sup>, Yunfei Jiang<sup>1</sup>, and Aixiang Chen<sup>1</sup>

<sup>1</sup> Software Research Institute, Sun Yat-sen University  
Guangzhou, Guangdong 510275, China  
[huangbodao@yahoo.com.cn](mailto:huangbodao@yahoo.com.cn)

<sup>2</sup> College of Information Engineering, Xiangtan University  
Xiangtan, Hunan 411105, China  
[zhonghua@xtu.edu.cn](mailto:zhonghua@xtu.edu.cn)

**Abstract.** The extended goals in non-deterministic domains are often expressed in temporal logic, particularly in *CTL* and *EAGLE*. No work has given a formal comparison between *EAGLE* and *CTL* on semantics, though it is said that the capability of representing the “intentional” aspects of goals and the possibility of dealing with failure are the main new features of *EAGLE* w.r.t. *CTL*.

According to the formal semantics for *EAGLE* and *CTL*, we prove that all the *EAGLE* formulas in which only  $LV_1$  operators (i.e. the operators representing the “intentional” aspects of goals) appear and some *EAGLE* formulas including  $LV_2$  operators (i.e. the operators dealing with failure and qualitative preferences) can be replaced by some *CTL* formulas without any change on semantics. Finally, we also find some basic and important goals in non-deterministic domains that exceed the expressive ability of *EAGLE*.

## 1 Introduction

Unlike classical planning [1,2], planning for extended goals in non-deterministic domains (e.g., robotics, scheduling, and control) is required to generate plans that satisfy conditions on their whole execution paths in order to deal with non-determinism and possible failures. Planning based on Markov Decision Processes, i.e. MDP-based planning [7,8] and planning based on model checking [3,4,5] are two main approaches to planning for extended goals in non-deterministic domains. In the former approach, planning goals are represented by means of utility functions; while in the latter approach, planning goals are expressed by formulas in temporal logic, particularly in *CTL* [6] and *EAGLE* [5].

Though it is said that the capability of representing the “intentional” aspects of goals and the possibility of dealing with failure are the main new features of *EAGLE* w.r.t. *CTL* in [5], little research has been devoted to the formal comparison between *EAGLE* and *CTL* on semantics. In this paper we prove that all the *EAGLE* formulas in which only  $LV_1$  operators (i.e. the operators representing the “intentional” aspects of goals) appear can be replaced by some

*CTL* formulas without any change on semantics, so can some *EAGLE* formulas including *LV*<sub>2</sub> operators (i.e. the operators dealing with failure and qualitative preferences). In addition, we find that some *CTL* formulas can not be expressed in *EAGLE*. Our purpose is not suggesting designers of systems have a choice between *EAGLE* and *CTL*. Actually, *EAGLE* and *CTL* have different features.

This paper is structured as follows. Section 2 illustrates the basic concepts of planning for extended goals in non-deterministic domains, and presents the formal semantics for *EAGLE* formulas and *CTL* formulas. The core of the paper, i.e. section 3 gives a formal comparison between *EAGLE* and *CTL* on semantics, and shows some limitations of the expressive power of *EAGLE* language. Section 4 concludes the paper by presenting some possibilities for further work.

## 2 Background

In this section, we briefly illustrate some definitions of planning for extended goals in non-deterministic domains [3] that are relevant to our work, and review the semantics for *CTL* formulas and *EAGLE* formulas over a Kripke structure [3]–[5]. Some examples of these definitions can be found in [3] and [5].

### 2.1 Planning for Extended Goals in Non-deterministic Domains

Following [3], a non-deterministic planning domain  $D$  is a tuple  $(B, Q, A, \rightarrow)$ , where  $B$  is a finite set of basic propositions,  $Q \in 2^B$  is the set of states,  $A$  is the finite set of actions, and  $\rightarrow \subseteq Q \times A \times Q$  is the transition relation.  $q \xrightarrow{a} q'$  denotes  $(q, a, q') \in \rightarrow$ . The relation  $\rightarrow$  is required to be total, i.e. for every  $q \in Q$  there is some  $a \in A$  and  $q' \in Q$  such that  $q \xrightarrow{a} q'$ .

In planning for extended goals, actions specified by plans do not only depend on the current state of the domain. The sequence of states that have appeared in the plan execution (i.e. the execution context) must be taken into account.

**Definition 1.** *A plan for a domain D is a tuple  $\pi = \langle C, c_0, act, ctxt \rangle$ , where C is a set of contexts,  $c_0 \in C$  is the initial context,  $act : Q \times C \rightarrow A$  is the action function, and  $ctxt : Q \times C \times Q \rightarrow C$  is the context function.*

A plan  $\pi$  is *executable* if, whenever  $act(q, c) = a$  and  $ctxt(q, c, q') = c'$ , then  $(q, a, q') \in \rightarrow$ . A plan  $\pi$  is *complete* if, whenever  $act(q, c) = a$  and  $(q, a, q') \in \rightarrow$ , then there is some context  $c'$  such that  $ctxt(q, c, q') = c'$  and  $act(q', c')$  is defined. In the following discussion, we consider only plans that are *executable* and *complete*.

In fact, any  $c \in C$  represents a class of sequences of states. To each class correspond the already accomplished subgoals.

**Definition 2.** *The execution structure of plan  $\pi$  in a domain D from state  $q_0$  is the structure  $K = \langle S, R, L \rangle$ , where  $S = \{(q, c) \mid act(q, c) \text{ is defined}\}$ ,  $R = \{((q, c), (q', c')) \in S \times S \mid q \xrightarrow{a} q' \text{ with } a = act(q, c) \text{ and } c' = ctxt(q, c, q')\}$ ,  $L(q, c) = \{b \mid b \in q\}$ .*

Indeed, the execution structure of any plan is a Kripke Structure [6], and it describes all the possible transition from  $(q, c)$  to  $(q', c')$  that can be triggered by executing the plan.

A path  $\delta$  in  $K$  is a sequence  $(q, c)_0(q, c)_1 \dots$  of states in  $S$  such that, for  $i \geq 0$ ,  $((q, c)_i, (q, c)_{i+1}) \in R$ . We write  $(q, c) \in \delta$  if  $(q, c)$  appears in  $\delta$ . Some notations on paths are now in order.  $\rho$  and  $\sigma$  denote infinite paths and finite paths respectively. The first state of  $\rho$  or  $\sigma$  is written as  $\text{first}(\rho)$  or  $\text{first}(\sigma)$ , and the last state of  $\sigma$  is written as  $\text{last}(\sigma)$ . If  $\text{last}(\sigma) = \text{first}(\sigma')$  then  $\sigma; \sigma'$  represents the path obtained by concatenating  $\sigma$  and  $\sigma'$ .  $\sigma \leq \sigma'$  denotes that  $\sigma$  is a prefix of  $\sigma'$ . We write  $\sigma < \sigma'$  if  $\sigma \leq \sigma'$  and  $\sigma \neq \sigma'$ . Given a set  $\Sigma$  of finite paths, the set of minimal paths in  $\Sigma$  is defined as:  $\text{min}(\Sigma) = \{\sigma \in \Sigma \mid \forall \sigma'. \sigma' < \sigma \Rightarrow \sigma' \notin \Sigma\}$ . In an execution structure  $K$ , a path  $\delta$  is called a *history* if,  $\delta \not< \delta'$  for any path  $\delta'$  appearing in  $K$ . Obviously, if a plan is executable and complete then all the histories appearing in its execution structure are infinite.

## 2.2 Semantics for *CTL* Formulas over Execution Structures

In planning under uncertainty, extended goals are often expressed as *CTL* formulas (such as [3] and [4]). Given a finite set  $B$  of basic propositions and let  $b \in B$ , the syntax of an extended goal  $g$  (i.e. a *CTL* formula) for a domain  $D$  is the following:

$$g ::= \top \mid \perp \mid b \mid \neg g \mid g \wedge g \mid g \vee g \mid AXg \mid EXg \mid A(gUg) \mid E(gUg) \mid A(gWg) \mid E(gWg)$$

$AFg$  and  $EFg$  are abbreviations of  $A(\top U g)$  and  $E(\top U g)$ , respectively.  $AGg$  and  $EGg$  are abbreviations of  $A(gW\perp)$  and  $E(gW\perp)$ , respectively. The set of *CTL* formulas is written as  $F_C$ .

The semantics for *CTL* formulas over an execution structure  $K = \langle S, R, L \rangle$  is defined by associating to each formula the set of states that satisfy it. Suppose  $p$  and  $t$  are *CTL* formulas and  $(q, c) \in S$ .  $K, (q, c) \models p$  denotes that  $p$  holds in the state  $(q, c)$  of  $K$ , and it is defined inductively as follows:

- $K, (q, c) \models \top$  and  $K, (q, c) \not\models \perp$
- If  $p \in B$ , then  $K, (q, c) \models p$  iff  $p \in L(q)$
- $K, (q, c) \models \neg p$  iff  $K, (q, c) \not\models p$
- $K, (q, c) \models p \wedge t$  iff  $K, (q, c) \models p$  and  $K, (q, c) \models t$
- $K, (q, c) \models p \vee t$  iff  $K, (q, c) \models p$  or  $K, (q, c) \models t$
- $K, (q, c) \models AXp$  iff for every history  $\delta = (q, c)_0(q, c)_1 \dots$  with  $(q, c)_0 = (q, c)$ , we have  $K, (q, c)_1 \models p$
- $K, (q, c) \models EXp$  iff there exists a history  $\delta = (q, c)_0(q, c)_1 \dots$  with  $(q, c)_0 = (q, c)$ , such that  $K, (q, c)_1 \models p$
- $K, (q, c) \models A(pUt)$  iff for all histories  $\delta = (q, c)_0(q, c)_1 \dots$  with  $(q, c)_0 = (q, c)$ , there exists  $i \geq 0$  such that  $K, (q, c)_i \models t$  and for all  $0 \leq j < i$ ,  $K, (q, c)_j \models p$
- $K, (q, c) \models E(pUt)$  iff there exists a history  $\delta = (q, c)_0(q, c)_1 \dots$  with  $(q, c)_0 = (q, c)$ , and a number  $i \geq 0$  such that  $K, (q, c)_i \models t$  and for all  $0 \leq j < i$ ,  $K, (q, c)_j \models p$

- $K, (q, c) \models A(pWt)$  iff for all histories  $\delta = (q, c)_0(q, c)_1 \dots$  with  $(q, c)_0 = (q, c)$ , we have:  $K, (q, c)_k \models p$  for any  $k \geq 0$ , or there exists  $i \geq 0$  such that  $K, (q, c)_i \models t$  and for all  $0 \leq j < i$ ,  $K, (q, c)_j \models p$
- $K, (q, c) \models E(pWt)$  iff there exists a history  $\delta = (q, c)_0(q, c)_1 \dots$  with  $(q, c)_0 = (q, c)$ , such that:  $K, (q, c)_k \models p$  for any  $k \geq 0$ , or there exists  $i \geq 0$  such that  $K, (q, c)_i \models t$  and for all  $0 \leq j < i$ ,  $K, (q, c)_j \models p$

Specially, if  $p$  is a propositional formula then  $K, (q, c) \models p$  is written as  $(q, c) \models p$ .

**Definition 3.** Let  $D$  be a planning domain and  $g$  be a CTL formula (i.e. a planning goal for  $D$ ). Let  $\pi$  be a plan for  $D$  and  $K$  be the corresponding execution structure. Plan  $\pi$  satisfies  $g$  from initial state  $q_0$ , written  $\pi, q_0 \models_C g$ , if  $K, (q_0, c_0) \models g$ . Plan  $\pi$  satisfies  $g$  from the set of initial states  $Q_0$  if  $\pi, q_0 \models_C g$  for each  $q_0 \in Q_0$ .

### 2.3 Semantics for $EAGLE$ Formulas over Execution Structures

In order to express preferences and capture intended meaning of some planning goals, Dal Lago et al [5] propose a new goal language named  $EAGLE$  language. Let  $B$  be a finite set of basic propositions and  $b \in B$ , the syntax of a propositional formula  $h$  and an extended goal  $g$  (i.e. an  $EAGLE$  formula) for a domain  $D$  are the following:

$$\begin{aligned} h ::= & \top \mid \perp \mid b \mid \neg h \mid h \vee h \mid h \wedge h \\ g ::= & h \mid g \text{ And } g \mid g \text{ Then } g \mid g \text{ Fail } g \mid \text{Repeat } g \mid \\ & \text{DoReach } h \mid \text{TryReach } h \mid \text{DoMaint } h \mid \text{TryMaint } h \end{aligned}$$

The set of  $EAGLE$  formulas is written as  $F_E$ .

The semantics of an  $EAGLE$  formula  $g$  over an execution structure  $K = \langle S, R, L \rangle$  is defined by, associating to each state  $(q, c)$  of  $K$  two sets  $S_g(q, c)$  and  $F_g(q, c)$  of finite paths in  $K$ .  $S_g(q, c)$  and  $F_g(q, c)$  represent the paths that lead to a success or to a failure in the achievement of goal  $g$  from  $(q, c)$ .

The formal definitions of  $S_g(q, c)$  and  $F_g(q, c)$  are given by induction on the structure of  $g$ :

- If  $g$  is a propositional formula then: if  $(q, c) \models g$  then  $S_g(q, c) = \{(q, c)\}$  and  $F_g(q, c) = \emptyset$ . Otherwise,  $S_g(q, c) = \emptyset$  and  $F_g(q, c) = \{(q, c)\}$ .
- If  $g = \text{TryReach } h$  then  $S_g(q, c) = \min\{\sigma \mid \text{first}(\sigma) = (q, c) \wedge \text{last}(\sigma) \models h\}$  and  $F_g(q, c) = \min\{\sigma \mid \text{first}(\sigma) = (q, c) \wedge \forall (q, c)' \in \sigma. (q, c)' \not\models h \wedge \forall \sigma' \geq \sigma. \text{Last}(\sigma') \not\models h\}$
- If  $g = \text{DoReach } h$  then: if there exists an infinite path  $\rho$  with  $\text{first}(\rho) = (q, c)$  such that  $(q, c)' \not\models h$  for each  $(q, c)' \in \rho$ , then  $S_g(q, c) = \emptyset$  and  $F_g(q, c) = \{(q, c)\}$ . Otherwise,  $S_g(q, c) = \min\{\sigma \mid \text{first}(\sigma) = (q, c) \wedge \text{last}(\sigma) \models h\}$  and  $F_g(q, c) = \emptyset$ .
- If  $g = \text{TryMaint } h$  then  $S_g(q, c) = \emptyset$  and  $F_g(q, c) = \min\{\sigma \mid \text{first}(\sigma) = (q, c) \wedge \text{last}(\sigma) \not\models h\}$ .
- If  $g = \text{DoMaint } h$  then: if  $(q, c)' \models h$  holds for all states  $(q, c)'$  reachable from  $(q, c)$  then  $S_g(q, c) = \emptyset$  and  $F_g(q, c) = \emptyset$ . Otherwise,  $S_g(q, c) = \emptyset$  and  $F_g(q, c) = \{(q, c)\}$ .

- If  $g = g_1$  **Then**  $g_2$  then  $S_g(q, c) = \{\sigma_1; \sigma_2 | \sigma_1 \in S_{g_1}(q, c) \wedge \sigma_2 \in S_{g_2}(\text{last}(\sigma_1))\}$  and  $F_g(q, c) = \{\sigma_1 | \sigma_1 \in F_{g_1}(q, c)\} \cup \{\sigma_1; \sigma_2 | \sigma_1 \in S_{g_1}(q, c) \wedge \sigma_2 \in F_{g_2}(\text{last}(\sigma_1))\}$ .
- If  $g = g_1$  **Fail**  $g_2$  then  $S_g(q, c) = \{\sigma_1 | \sigma_1 \in S_{g_1}(q, c)\} \cup \{\sigma_1; \sigma_2 | \sigma_1 \in F_{g_1}(q, c) \wedge \sigma_2 \in S_{g_2}(\text{last}(\sigma_1))\}$  and  $F_g(q, c) = \{\sigma_1; \sigma_2 | \sigma_1 \in F_{g_1}(q, c) \wedge \sigma_2 \in F_{g_2}(\text{last}(\sigma_1))\}$ .
- If  $g = g_1$  **And**  $g_2$  then  $S_g(q, c) = \min\{\sigma | \exists \sigma_1 \leq \sigma. \sigma_1 \in S_{g_1}(q, c) \wedge \exists \sigma_2 \leq \sigma. \sigma_2 \in S_{g_2}(q, c)\}$  and  $F_g(q, c) = \min(F_{g_1}(q, c) \cup F_{g_2}(q, c))$ .
- If  $g = \text{Repeat } g'$  then  $S_g(q, c) = \emptyset$  and  $F_g(q, c) = \{\sigma_0; \sigma_1; \dots; \sigma_n; \sigma' | \sigma' \in F_{g'}(\text{first}(\sigma')) \wedge \exists \sigma'_i \in S_{g'}(\text{first}(\sigma'_i)). \sigma_i = \sigma'_i; (\text{last}(\sigma'_i), \text{last}(\sigma_i))\}$ .

Now the notion of a plan satisfying a goal expressed in  $EAGLE$  language can be defined.

**Definition 4.** Let  $D$  be a planning domain and  $g$  be a  $EAGLE$  formula (i.e. a planning goal for  $D$ ). Let  $\pi$  be a plan for  $D$  and  $K$  be the corresponding execution structure. Plan  $\pi$  satisfies  $g$  from initial state  $q_0$ , written  $\pi, q_0 \models_E g$ , if  $F_g(q_0, c_0) = \emptyset$ . Plan  $\pi$  satisfies  $g$  from the set of initial states  $Q_0$  if  $\pi, q_0 \models_E g$  for each  $q_0 \in Q_0$ .

### 3 Comparison Between $CTL$ and $EAGLE$

From definition 3 and definition 4, it can be found that whether a plan  $\pi$  satisfies a goal expressed in  $CTL$  language or  $EAGLE$  language depends on the execution structure of  $\pi$ . So we can make a semantic comparison between  $CTL$  language and  $EAGLE$  language with a common execution structure.

**Definition 5.** Suppose  $g_1 \in F_C$  and  $g_2 \in F_E$ . If  $\pi, q_0 \models_C g_1 \Leftrightarrow \pi, q_0 \models_E g_2$  holds for any domain  $D = (B, Q, A, \rightarrow)$ , any  $q_0 \in Q$ , and any plan  $\pi$  for  $D$ , then  $g_1$  is equivalent to  $g_2$  on semantics, written  $g_1 \equiv_{\{C,E\}} g_2$  or  $g_2 \equiv_{\{C,E\}} g_1$ .

#### 3.1 Substitution of $CTL$ Formulas for $EAGLE$ Formulas

We now compare  $EAGLE$  language and  $CTL$  language on semantics by induction on the syntax of  $EAGLE$  formulas.

From definition 3, definition 4 and the definitions of  $(q_0, c_0) \models g$  and  $F_g(q_0, c_0)$ , we can get proposition 1 directly.

**Proposition 1.** If  $g \in F_C \cap F_E$ , i.e.  $g$  is a propositional formula then we have  $g \equiv_{\{C,E\}} g$ .

Minimal paths (see section 2.1) are often used to describe paths that lead to a success or to a failure in the achievement of an  $EAGLE$  goal  $g$  (see section 2.3). Here are a lemma about minimal paths, and it is useful in the proofs of the following propositions.

**Lemma 1.** Let  $\Sigma$  be a set of finite paths. Then  $\Sigma = \emptyset \Leftrightarrow \min(\Sigma) = \emptyset$ .

*Proof.* (1) If  $\Sigma = \emptyset$  then clearly the set of  $\{\sigma \in \Sigma | \forall \sigma'. \sigma' < \sigma \Rightarrow \sigma' \notin \Sigma\}$ , i.e.  $\min(\Sigma)$  is an empty set. So we have  $\Sigma = \emptyset \Rightarrow \min(\Sigma) = \emptyset$ . (2) Let  $\min(\Sigma) = \emptyset$ . Suppose  $\Sigma \neq \emptyset$ , then there must exist a path  $\sigma' \in \Sigma$  such that the number of states in  $\sigma'$  is not more than the number of states in  $\sigma$  for any  $\sigma \in \Sigma$ , because all the paths in  $\Sigma$  are finite. Therefore  $\forall \sigma''. \sigma'' < \sigma' \Rightarrow \sigma'' \notin \Sigma$  and  $\sigma' \in \min(\Sigma)$  hold, i.e.  $\min(\Sigma) \neq \emptyset$  (this is in contradiction to  $\min(\Sigma) = \emptyset$ ). So  $\Sigma = \emptyset$ , i.e.  $\min(\Sigma) = \emptyset \Rightarrow \Sigma = \emptyset$ . (3) From (1) and (2), we have  $\Sigma = \emptyset \Leftrightarrow \min(\Sigma) = \emptyset$ .  $\square$

Proposition 2, 3, and 4 say that all the basic  $EAGLE$  reachability goals and maintenance goal can be expressed in  $CTL$  language.

**Proposition 2.** *If  $h$  is a propositional formula then  $\mathbf{DoReach} h \equiv_{\{C,E\}} AF h$ .*

*Proof.* Let  $g = \mathbf{DoReach} h$ . Then  $\pi, q_0 \models_E g \Leftrightarrow$ (from definition 4)  $F_g(q_0, c_0) = \emptyset \Leftrightarrow$ (from the definition of  $F_g(q, c)$ ) there is no infinite path  $\rho$  from  $(q_0, c_0)$  such that  $(q, c)' \not\models h$  for each  $(q, c)' \in \rho \Leftrightarrow$  for all histories  $\delta = (q, c)_0(q, c)_1 \dots$  such that  $(q, c)_0 = (q_0, c_0)$ , there exists  $i \geq 0$  such that  $(q, c)_i \models h \Leftrightarrow K, (q_0, c_0) \models AF h \Leftrightarrow$ (from definition 3)  $\pi, q_0 \models_C AF h$ . So from definition 5 and definition 6, we have  $\mathbf{DoReach} h \equiv_{\{C,E\}} AF h$ .  $\square$

**Proposition 3.** *If  $h$  is a propositional formula then  $\mathbf{TryReach} h \equiv_{\{C,E\}} A(EFh W h)$ .*

*Proof.* Let  $g = \mathbf{TryReach} h$ . Then  $\pi, q_0 \models_E g \Leftrightarrow$ (from definition 4)  $F_g(q_0, c_0) = \emptyset \Leftrightarrow$ (from the definition of  $F_g(q, c)$  and lemma1 1)  $\{\sigma | first(\sigma) = (q_0, c_0) \wedge \forall (q, c)' \in \sigma. (q, c)' \not\models h \wedge \forall \sigma' \geq \sigma. last(\sigma') \not\models h\} = \emptyset \Leftrightarrow$  for any finite path  $\sigma$  such that  $first(\sigma) = (q_0, c_0)$  and  $(q, c)' \not\models h$  for each  $(q, c)' \in \sigma$ , there exists an finite path  $\sigma'$  such that  $\sigma' > \sigma$  and  $last(\sigma') \models h \Leftrightarrow$  for all histories  $\delta = (q, c)_0(q, c)_1 \dots$  such that  $(q, c)_0 = (q_0, c_0)$ , we have: there exists  $i \geq 0$  such that  $(q, c)_i \models h$ ; or for all  $(q, c)' \in \delta$ , there exists a finite path  $\sigma'$  from  $(q, c)'$  such that  $last(\sigma') \models h \Leftrightarrow$ (from the definitions of  $K, (q, c) \models A(pWt)$  and  $K, (q, c) \models EFp$ )  $K, (q_0, c_0) \models A(EFh W h) \Leftrightarrow$ (from definition 3)  $\pi, q_0 \models_C A(EFh W h)$ . So from definition 5 and definition 6, we have  $\mathbf{TryReach} h \equiv_{\{C,E\}} A(EFh W h)$ .  $\square$

**Proposition 4.** *If  $h$  is a propositional formula then  $\mathbf{DoMaint} h \equiv_{\{C,E\}} AG h \equiv_{\{C,E\}} \mathbf{TryMaint} h$ .*

*Proof.* Let  $g_1 = \mathbf{DoMaint} h$  and  $g_2 = \mathbf{TryMaint} h$ .

1.  $\pi, q_0 \models_E g_1 \Leftrightarrow$ (from definition 4)  $F_{g_1}(q_0, c_0) = \emptyset \Leftrightarrow$ (from the definition of  $F_{g_1}(q, c)$ )  $(q, c)' \models h$  holds for all states  $(q, c)'$  reachable from  $(q_0, c_0) \Leftrightarrow$  for all histories  $\delta = (q, c)_0(q, c)_1 \dots$  such that  $(q, c)_0 = (q_0, c_0)$ , we have  $(q, c)_k \models h$  for all  $k \geq 0 \Leftrightarrow K, (q_0, c_0) \models AG h \Leftrightarrow$ (from definition 3)  $\pi, q_0 \models_C AG h$ .
2.  $\pi, q_0 \models_E g_2 \Leftrightarrow$ (from definition 4)  $F_{g_2}(q_0, c_0) = \emptyset \Leftrightarrow$ (from the definition of  $F_{g_2}(q, c)$  and lemma 1)  $\{\sigma | first(\sigma) = (q_0, c_0) \wedge last(\sigma) \not\models h\} = \emptyset \Leftrightarrow$  for all finite paths  $\sigma$  with  $first(\sigma) = (q_0, c_0)$ , we have  $last(\sigma) \models h \Leftrightarrow$  for all histories  $\delta = (q, c)_0(q, c)_1 \dots$  such that  $(q, c)_0 = (q_0, c_0)$ , we have  $(q, c)_k \models h$  for all  $k \geq 0 \Leftrightarrow$ (similar to  $g_1$ )  $\pi, q_0 \models_C AG h$ .

3. From step 1, step 2, definition 5 and definition 6, we have  $\mathbf{DoMaint} h \equiv_{\{C,E\}} AG h \equiv_{\{C,E\}} \mathbf{TryMaint} h$ .  $\square$

Some notations are in order. The temporal operators (i.e. **DoMaint**, **TryMaint**, **DoReach** and **TryReach**) that could not be nested are called  $LV_1$  operators, and the temporal operators (i.e. **And**, **Then**, **Fail**, and **Repeat**) that can be arbitrarily nested are called  $LV_2$  operators. Let  $F_{Eo} = \{p | p \in F_E \text{ and } p \text{ has no } LV_2 \text{ operator}\}$ . The function  $fec : F_{Eo} \rightarrow F_C$  is given as: if  $h$  is a proposition formula then  $fec(h) = h$ ,  $fec(\mathbf{DoReach} h) = AFh$ ,  $fec(\mathbf{TryReach} h) = A((EFh)Wh)$  and  $fec(\mathbf{DoMaint} h) = AGh = fec(\mathbf{TryMaint} h)$ . From proposition 1–4, we have  $\forall g \in F_{Eo}. g \equiv_{\{C,E\}} fec(g)$ .

Now let's consider some  $EAGLE$  formulas that contain  $LV_2$  operators.

**Proposition 5.** *Let  $g_1 \in F_{Eo}$ ,  $g_2 \in F_E$ ,  $h$  be a propositional formula and  $g = g_1 \mathbf{Then} g_2$ . Suppose there is a CTL formula  $cg_2$  such that  $g_2 \equiv_{\{C,E\}} cg_2$ . Then (1) If  $g_1 = h$  then  $g \equiv_{\{C,E\}} h \wedge cg_2$ ; (2) if  $g_1 = \mathbf{DoMaint} h$  or  $g_1 = \mathbf{TryMaint} h$  then  $g \equiv_{\{C,E\}} AGh$ ; (3) if  $g_1 = \mathbf{DoReach} h$  then  $g \equiv_{\{C,E\}} AF(h \wedge cg_2)$ ; (4) if  $g_1 = \mathbf{TryReach} h$  then  $g \equiv_{\{C,E\}} A((EF(h \wedge cg_2))W(h \wedge cg_2))$ .*

*Proof.* Under the conditions given in this proposition, we have  $\pi, q_0 \models_E g \Leftrightarrow$  (from definition 4)  $F_g(q_0, c_0) = \emptyset \Leftrightarrow$  (from the definition of  $F_g(q, c)$ )  $F_{g_1}(q_0, c_0) \cup \{\sigma_1; \sigma_2 | \sigma_1 \in S_{g_1}(q_0, c_0) \wedge \sigma_2 \in F_{g_2}(\text{last}(\sigma_1))\} = \emptyset \Leftrightarrow$  (from the character of function  $fec$ )

$$K, (q_0, c_0) \models fec(g_1) \text{ and } \forall \sigma_1 \in S_{g_1}(q_0, c_0). K, \text{last}(\sigma_1) \models cg_2 \quad (1)$$

1. Assume  $g_1 = h$ . Then (1)  $\Leftrightarrow$  (from the definition of  $S_{g_1}(q, c)$ )  $K, (q_0, c_0) \models h$  and  $K, (q_0, c_0) \models cg_2 \Leftrightarrow K, (q_0, c_0) \models h \wedge cg_2$
2. Assume  $g_1 = \mathbf{DoMaint} h$  (or  $\mathbf{TryMaint} h$ ). Then (1)  $\Leftrightarrow$  (from the definition of  $S_{g_1}(q, c)$ )  $K, (q_0, c_0) \models AGh$
3. Assume  $g_1 = \mathbf{DoReach} h$ . Then (1)  $\Leftrightarrow K, (q_0, c_0) \models AFh$  and  $\forall \sigma_1 \in S_{g_1}(q_0, c_0). K, \text{last}(\sigma_1) \models cg_2 \Leftrightarrow$  (from the definition of  $K, (q, c) \models AFp$ ) for all histories  $\delta = (q, c)_0 (q, c)_1 \dots$  with  $(q, c)_0 = (q_0, c_0)$ , there exists  $i \geq 0$  such that  $K, (q, c)_i \models h$  and  $K, (q, c)_i \models cg_2 \Leftrightarrow$  (from definitions of  $K, (q, c) \models AFp$  and  $K, (q, c) \models p \wedge t$ )  $K, (q_0, c_0) \models AF(h \wedge cg_2)$
4. Assume  $g_1 = \mathbf{TryReach} h$ . Then (1)  $\Leftrightarrow K, (q_0, c_0) \models A((EFh)Wh)$  and  $\forall \sigma_1 \in S_{g_1}(q_0, c_0). K, \text{last}(\sigma_1) \models cg_2 \Leftrightarrow$  (similar to item 3)  $K, (q_0, c_0) \models A((EF(h \wedge cg_2))W(h \wedge cg_2))$

From item 1–4 and definition 3, proposition 5 holds.  $\square$

**Fail** operator deals with failure/recovery and with preferences among goals. Proposition 6 says that some  $EAGLE$  formulas that contain **Fail** operators can be expressed in *CTL* language.

**Proposition 6.** *Let  $g_1 \in F_{Eo}$ ,  $g_2 \in F_E$ ,  $h$  be a propositional formula and  $g = g_1 \mathbf{Fail} g_2$ . Suppose there is a CTL formula  $cg_2$  such that  $g_2 \equiv_{\{C,E\}} cg_2$ . Then (1) If  $g_1 = h$  or  $g_1 = \mathbf{DoMaint} h$  or  $g_1 = \mathbf{DoReach} h$  then  $g \equiv_{\{C,E\}} fec(g_1) \vee$*

$\neg fec(g_1) \wedge cg_2$  (2) if  $g_1 = \text{TryMaint } h$  then  $g \equiv_{\{C,E\}} AGh \vee EF\neg h \wedge A(h W (\neg h \wedge cg_2))$  (3) if  $g_1 = \text{TryReach } h$  then  $g \equiv_{\{C,E\}} fec(g_1) \vee \neg fec(g_1) \wedge A((\neg h \wedge EFh) W (h \vee AG\neg h \wedge cg_2))$ .

*Proof.* Under the conditions given in this proposition, we have  $\pi, q_0 \models_E g \Leftrightarrow$  (from definition 4)  $F_g(q_0, c_0) = \emptyset \Leftrightarrow$  (from the definition of  $F_g(q, c)$ )  $\{\sigma_1; \sigma_2 | \sigma_1 \in F_{g_1}(q_0, c_0) \wedge \sigma_2 \in F_{g_2}(\text{last}(\sigma_1))\} = \emptyset \Leftrightarrow F_{g_1}(q_0, c_0) = \emptyset$ , or  $F_{g_1}(q_0, c_0) \neq \emptyset$  and  $\forall \sigma_1 \in F_{g_1}(q_0, c_0). F_{g_2}(\text{last}(\sigma_1)) = \emptyset \Leftrightarrow$  (from the character of function  $fec$ )

$$\begin{aligned} K, (q_0, c_0) &\models fec(g_1), \text{ or} \\ K, (q_0, c_0) &\models \neg fec(g_1) \text{ and } \forall \sigma \in F_{g_1}(q_0, c_0). K, \text{last}(\sigma) \models cg_2 \end{aligned} \quad (2)$$

1. Assume  $g_1 = h$  or  $g_1 = \text{DoMaint } h$  or  $g_1 = \text{DoReach } h$ . Then (2)  $\Leftrightarrow$  (from the definition of  $F_{g_1}(q, c)$ )  $K, (q_0, c_0) \models fec(g_1)$ , or  $K, (q_0, c_0) \models \neg fec(g_1)$  and  $K, (q_0, c_0) \models cg_2 \Leftrightarrow K, (q_0, c_0) \models fec(g_1) \vee \neg fec(g_1) \wedge cg_2$
2. Assume  $g_1 = \text{TryMaint } h$ . Then (2)  $\Leftrightarrow$  (all the histories  $\delta = (q, c)_0 (q, c)_1 \dots$  with  $(q, c)_0 = (q_0, c_0)$  can be sorted into two classes: (a)  $h$  holds forever; (b)  $\exists i \geq 0. (q, c)_i \models \neg h$  and  $(q, c)_j \models h$  for all  $0 \leq j < i$ ). So any history of class (b) has only one prefix  $\sigma$  such that  $\sigma \in F_{g_1}(q_0, c_0)$ )  $K, (q_0, c_0) \models fec(g_1)$ , or  $K, (q_0, c_0) \models \neg fec(g_1)$  and  $K, (q_0, c_0) \models A(h W (\neg h \wedge cg_2)) \Leftrightarrow K, (q_0, c_0) \models AGh \vee EF\neg h \wedge A(h W (\neg h \wedge cg_2))$
3. Assume  $g_1 = \text{TryReach } h$ . Then (2)  $\Leftrightarrow$  (all the histories  $\delta = (q, c)_0 (q, c)_1 \dots$  with  $(q, c)_0 = (q_0, c_0)$  can be sorted into three classes: (a)  $\exists i \geq 0. (q, c)_i \models h$  and  $(q, c)_j \models \neg h$  for all  $0 \leq j < i$ ; (b)  $\neg h \wedge EFh$  holds forever; (c)  $\exists i \geq 0. (q, c)_i \models AG\neg h$  and  $(q, c)_j \models \neg h \wedge EFh$  for all  $0 \leq j < i$ . So any history of class (c) has only one prefix  $\sigma$  such that  $\sigma \in F_{g_1}(q_0, c_0)$ )  $K, (q_0, c_0) \models fec(g_1)$ , or  $K, (q_0, c_0) \models \neg fec(g_1)$  and  $K, (q_0, c_0) \models A((\neg h \wedge EFh) W (h \vee (AG\neg h) \wedge cg_2)) \Leftrightarrow K, (q_0, c_0) \models fec(g_1) \vee \neg fec(g_1) \wedge A((\neg h \wedge EFh) W (h \vee AG\neg h \wedge cg_2))$

From item 1–3 and definition 3, proposition 6 holds.  $\square$

The following proposition says that **And** operator works like the basic conjunction operator.

**Proposition 7.** Let  $g_1 \in F_E$ ,  $g_2 \in F_E$ ,  $h$  be a propositional formula and  $g = g_1 \text{And } g_2$ . Suppose  $cg_1 \in F_C$ ,  $cg_2 \in F_C$ ,  $g_1 \equiv_{\{C,E\}} cg_1$  and  $g_2 \equiv_{\{C,E\}} cg_2$ . Then  $g \equiv_{\{C,E\}} cg_1 \wedge cg_2$ .

*Proof.*  $\pi, q_0 \models_E g \Leftrightarrow$  (from definition 4)  $F_g(q_0, c_0) = \emptyset \Leftrightarrow$  (from the definition of  $F_g(q, c)$  and lemma 1)  $F_{g_1}(q, c) \cup F_{g_2}(q, c) = \emptyset \Leftrightarrow F_{g_1}(q, c) = \emptyset$  and  $F_{g_2}(q, c) = \emptyset \Leftrightarrow$  (from definition 3, 4 and 6)  $K, (q_0, c_0) \models cg_1 \wedge cg_2 \Leftrightarrow$  (from definition 3)  $\pi, q_0 \models_C cg_1 \wedge cg_2$   $\square$

The following proposition says that **Repeat** operator works like the *CTL* operator *AG* in some cases.

**Proposition 8.** Let  $g' \in F_{Eo}$ ,  $h$  be a propositional formula and  $g = \text{Repeat } g'$ . Then  $g \equiv_{\{C,E\}} AG(fec(g'))$ .

*Proof.*  $\pi, q_0 \models_E g \Leftrightarrow$ (from definition 4)  $F_g(q_0, c_0) = \emptyset \Leftrightarrow$ (from the definition of  $F_g(q, c)$ )  $F_{g'}(q_0, c_0) = \emptyset, \{\sigma_0; \sigma'\} = \emptyset, \dots \{\sigma_0; \sigma_1; \dots; \sigma_n; \sigma'\} = \emptyset, \dots$  where  $\sigma' \in F_{g'}(\text{first}(\sigma')) \wedge \exists \sigma'_i \in S_{g'}(\text{first}(\sigma'_i)). \sigma_i = \sigma'_i; (\text{last}(\sigma'_i), \text{last}(\sigma_i)) \Leftrightarrow$ (from the character of function  $\text{fec}$ )

$$K, (q_0, c_0) \models \text{fec}(g') \text{ and } K, \text{last}(\sigma_i) \models \text{fec}(g') \text{ for all } i \geq 0 \quad (3)$$

1. Assume  $g' = h$ . Then (3)  $\Leftrightarrow$ (from the definition of  $S_{g'}(q, c)$ ) for any history  $\delta = (q, c)_0 (q, c)_1 \dots$  with  $(q, c)_0 = (q_0, c_0)$ , we have  $K, (q, c)_0 \models h, K, (q, c)_1 \models h, \dots$  and so on  $\Leftrightarrow K, (q_0, c_0) \models AGh$
2. Assume  $g' = \text{DoMaint } h$  or  $g' = \text{TryMaint } h$ . Then (3)  $\Leftrightarrow$ (from the definition of  $S_{g'}(q, c)$ )  $K, (q_0, c_0) \models AGh \Leftrightarrow K, (q_0, c_0) \models AG(AGh)$
3. Assume  $g' = \text{DoReach } h$ . Then (3)  $\Leftrightarrow K, (q_0, c_0) \models AFh$  and  $K, \text{last}(\sigma_i) \models AFh$  for all  $i \geq 0 \Leftrightarrow$ (if  $K, (q, c) \models AFh$  then any history  $\delta$  from  $(q, c)$  has a prefix  $\sigma$  such that  $\sigma \in S_{g'}(q, c)$  and  $\forall \sigma \in S_{g'}(q, c) \forall (q, c)' \in \sigma. K, (q, c)' \models AFh$ )  $K, (q, c) \models AFh$  holds for all states  $(q, c)$  reachable from  $(q_0, c_0) \Leftrightarrow K, (q_0, c_0) \models AGAFh$
4. Assume  $g' = \text{TryReach } h$ . Then (3)  $\Leftrightarrow K, (q_0, c_0) \models A(EFh \ W h)$  and  $K, \text{last}(\sigma_i) \models A(EFh \ W h)$  for all  $i \geq 0 \Leftrightarrow$ (if  $K, (q, c) \models A(EFh \ W h)$  then: for any history  $\delta$  from  $(q, c)$ , there exists a prefix  $\sigma$  of  $\delta$  such that  $\sigma \in S_{g'}(q, c)$  or  $EFh \wedge \neg h$  holds forever in  $\delta$ ; and  $\forall \sigma \in S_{g'}(q, c) \forall (q, c)' \in \sigma. K, (q, c)' \models A(EFh \ W h)$ )  $K, (q, c) \models A(EFh \ W h)$  holds for all states  $(q, c)$  reachable from  $(q_0, c_0) \Leftrightarrow K, (q_0, c_0) \models AG(A(EFh \ W h))$

From item 1–4 and definition 3, proposition 8 holds.  $\square$

### 3.2 Beyond $EAGLE$ Language

Look up the definitions of  $S_g(q, c)$  and  $F_g(q, c)$  where  $g \in F_E$ , we can find that the paths having a fixed length, except the paths that have only one state, can not be described in  $EAGLE$  language. But in  $CTL$ , these paths can be described by using the “next time” temporal operator  $X$ .

From definition 4 and the definitions of  $F_g(q, c)$ ,  $\pi, q_0 \models_E g \Leftrightarrow \forall$  history  $\delta$  from  $(q_0, c_0). \forall \sigma < \delta. \sigma \notin F_g(q_0, c_0)$ , i.e. the semantics for  $EAGLE$  formulas is built on features of all the histories that are from initial state. So without the relevant operator that can replace “ $\neg$ ”, it is impossible to describe the properties that only hold in some histories. In  $CTL$ , we can describe these properties by making use of the path existential quantifier  $E$ .

In planning based on model checking, planning goals are expressed as temporal logic formulas, e.g.  $CTL$  formulas and  $EAGLE$  formulas. But these formulas are concerning the planning domains that take the form of  $(B, Q, A, \rightarrow)$ , they can not deal with some planning goals built on more complicated domains, e.g. the planning goals dealing with probabilities and competing criteria.

## 4 Conclusions

From the formal semantics for  $EAGLE$  formulas and  $CTL$  formulas, we have proved that all the  $EAGLE$  formulas in which only  $LV_1$  operators (i.e. the op-

erators representing the “intentional” aspects of goals) appear can be replaced by some *CTL* formulas without any change on semantics, so can some *EAGLE* formulas including *LV*<sub>2</sub> operators, i.e. the operators dealing with failure and qualitative preferences. On the other hand, from the definitions of paths leading to a success or to a failure in the achievement of  $g \in F_E$  from initial state, we have found that some essential goals expressed in *CTL*, e.g. *AXp*, *EXp* and *EFp* (where *p* is a propositional formula) could not be expressed in *EAGLE*. The comparison on semantics between *CTL* and other *EAGLE* formulas, i.e. the formulas in which *LV*<sub>2</sub> operators can be arbitrary nested is a topic for future research. This paper and [5] show that *EAGLE* and *CTL* have different features. How to find a way in which *EAGLE* can cooperate with *CTL* may be an interesting problem.

Just as planning based on model checking, MDP-based planning is another important approach to planning for extended goals under uncertainty. In MDP-based planning, goals are presented by means of utility functions that can embody quantitative preferences and competing criteria. But temporal logic formulas only deal with qualitative goals. Comparison of the expressive power between utility functions and temporal logic, and giving a new approach by which all the planning goals in the form of utility functions or temporal logic can be represented are also objectives in our intending research.

## References

1. Fikes, R.E., and Nilsson, N.J.: STRIPS: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*. **2(3–4)** (1971) 189–208
2. Penberthy, J., and Weld, D.: UCPOP: A sound, complete, partial order planner for ADL. In: Proceedings of KR-92, (1992)
3. Pistore, M., and Traverso, P.: Planning as model checking for extended goals in non-deterministic domains. In: Proceedings of the International Joint Conference on Artificial Intelligence, Morgan Kaufmann (2001) 479–484
4. Daniele, M., Traverso, P., and Vardi, M.: Strong cyclic planning revisited. In: Proceedings of the European Conference on Planning, (1999) 35–48
5. Dal Lago, U., Pistore, M., and Traverso, P.: Planning with a language for extended goals. In: AAAI/IAAI Proceedings, Edmonton (2002) 447–454
6. Emerson, E. A.: Temporal and modal logic. In: J. van Leeuwen, eds., *Handbook of Theoretical Computer Science, Volume B: Formal Models and Semantics*. Elsevier (1990) 995–1072
7. Dean, T., Kaelbling, L., Kirman, J., and Nicholson, A.: Planning with deadlines in stochastic domains. In: Proceedings of the National Conference on Artificial Intelligence (1993) 574–579
8. Thiebaux, S., Kabanza, F., and Slaney, J.: Anytime state-based solution methods for decision processes with non-Markovian rewards. In: Proceedings of the Conference on Uncertainty in Artificial Intelligence (2002)

# Trajectory Modification Using Elastic Force for Collision Avoidance of a Mobile Manipulator

Nak Yong Ko<sup>1</sup>, Reid G. Simmons<sup>2</sup>, and Dong Jin Seo<sup>1</sup>

<sup>1</sup> Dept. Control and Instrumentation Eng., Chosun Univ., Korea  
nyko@chosun.ac.kr

<sup>2</sup> The Robotics Institute, Carnegie Mellon Univ., USA  
reids@cs.cmu.edu

**Abstract.** This paper proposes a method for collision avoidance of a mobile manipulator. The method deals with the problem of driving a mobile manipulator from a starting configuration to a goal configuration avoiding obstacles. It modifies planned trajectory at every sampling time using elastic force and potential field force. It puts way poses throughout a planned trajectory, and the trajectory modification is accomplished by adjusting the way poses. The way poses are adjusted subject to the elastic force and the potential field force. The procedure repeats until the robot reaches its goal configuration through the way poses. This method results in smooth and adaptive trajectory in an environment with moving as well as stationary obstacles.

## 1 Introduction

This paper presents a collision avoidance method for a mobile manipulator. The method applies to the robots with nonholonomic mobile base as well as the holonomic one. It drives the robot from an initial configuration to a goal configuration avoiding obstacles which are moving as well as stationary. The pose and shape of the obstacles are assumed to be given in real time. In some cases the obstacles can be another robots or machines whose locations are known or monitored online. In other cases, some sensors such as scanning range sensors, sonars, or cameras can be used for obstacle detection.

Though it's hard to find a collision free trajectory for a mobile manipulator, there have been many well known approaches available for collision avoidance of a mobile base. The artificial potential field approach [1] and Vector Field Histogram [2] are popular methods for obstacle avoidance of mobile robots, and they have many variations for improvement [3]. While it is easy to apply and is widely used, it's local minimum problem requires some additional means to avoid trap or oscillatory situation. Another approach considering the path as the piecewise continuation of the circular segments [4,5] can be applied to non-holonomic mobile bases as well as holonomic bases. Also the dynamic windows approaches [6,7] and the elastic band approach [8] well suit for obstacle avoidance of the mobile robots. The ASL method combined geometric search and elastic force approach. It also features concurrent motion planning and modification,

and is used to guide a robot in crowded exhibition environment for extended periods [9].

Many robot control architectures usually separated motion planning and execution in two different control layers. In these frameworks, the motion planning level doesn't pay enough attention to the real time obstacle avoidance. As the computing ability improves, some methods incorporate motion planning and motion execution in one frame. Among them, the elastic strip method [10] modifies its planned motion for obstacle avoidance at every sampling time and traces the modified trajectory plan. Still, the elastic strip method is computation intensive because it requires computation of sweep volume, elastic tunnel, and checking the overlap of the elastic tunnel with obstacles. In some cases, a nonholonomic mobile base still cannot trace some of the planned trajectory because of its limited mobility.

The method proposed in this paper uses elastic force and potential field force, and is applicable to mobile manipulators with nonholonomic mobile base as well as holonomic one. The elastic force differs from usual attraction force toward goal pose. The elastic force not only attracts the robot toward the goal pose, it also keeps the trajectory continuous and smooth toward the goal pose. The potential field force deforms the trajectory for obstacle avoidance.

Unlike the Elastic Strip method, it doesn't use geometric calculation which has adverse effect on real time application. The method sets way poses between the initial pose and goal pose throughout the trajectory at predefined intervals. The robot follows through the way poses in sequence until it reaches to the goal pose. The section 2 explains the procedure of the method in detail. The section 3 shows some simulation results as well as a real implementation example. Section 4 concludes the paper with some discussions on the method.

## 2 Collision Avoidance Procedure

At every sampling time, the method modifies the trajectory planned at the previous sampling time. The modification of the trajectory means the modification of the way poses. The procedure is as follows.

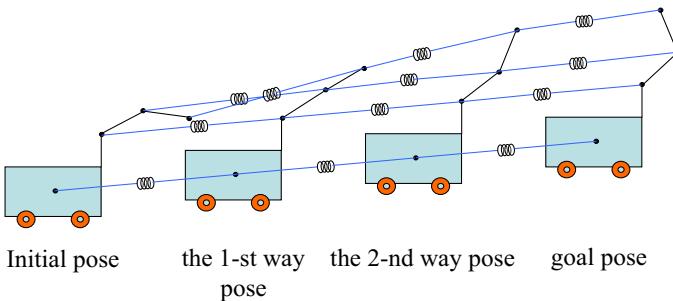
1. Set an initial trajectory and way poses along the trajectory.
2. Modify the trajectory by modifying the way poses using the following procedure.
  - (a) Calculate the elastic force and repulsive potential field force exerting to every control points on the way poses.
  - (b) Calculate the torque or force on the joints of the robot in the way poses due to the force acting on the control points.
  - (c) Calculate the deformation angle or distance of each joint due to the torque or force on the joint.
  - (d) Change the joint rotation angle or translational distance by the deformation angle or distance calculated in the step ( 2c), and establish the new way poses.

3. Trace the trajectory through the way poses.
4. Check if the robot reaches to the goal pose. If the robot reaches close enough to the goal pose then stop the procedure. If not, go to the step 2.

The following sections explain the procedure in detail.

## 2.1 Initial Trajectory and Way Pose

The method uses way poses for trajectory modification. It sets an initial trajectory, and modifies the trajectory at every sampling time. The trajectory is considered as an elastic thread connecting the starting pose to goal pose. Between the initial pose to goal pose we set a number of way poses, and virtual elastic string connects each point on the robot body from the initial pose to the final goal pose via the corresponding points on each way pose. Fig. 1 depicts the initial pose, final pose, way poses, and the elastic connections between them.



**Fig. 1.** Initial pose, final goal pose and way poses for a trajectory of a mobile manipulator

Though the path changes its shape subject to the elastic force and repulsive force, the topological property of the path relative to the location of the obstacles never change, because the path changes continuously and the path will neither be cut off nor be reconnected. If we set more way poses along the path, then the robot motion will become smoother. However, the increased number of way poses requires more calculations because the algorithm modifies every way pose at each sampling time. On the contrary, even though the way poses don't overlap with the obstacles, the robot can collide with some obstacles if consecutive way poses are not set close enough.

## 2.2 Elastic Force and Repulsive Force Acting on a Control Point

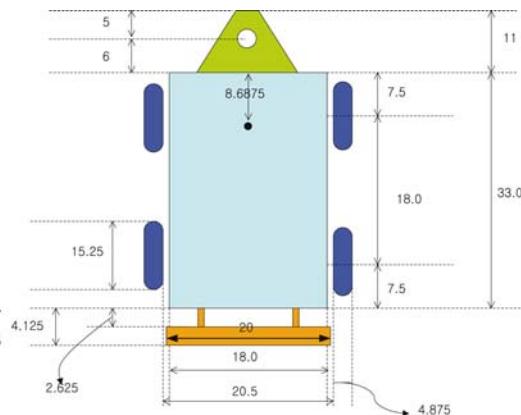
On the robot, some points called the control points are set. At the control points exert the elastic force and repulsive force. Though, it is the most desirable to apply and spread the elastic force and repulsive force at all the points throughout the body of the robot, only several points should be chosen for practical application. The control points should be chosen so that the effect of the forces

on them can be as equivalent as possible to the effect of the forces spreading throughout the whole body of the robot.

In our application, we use the robot Bullwinkle shown in the Fig. 2(a) for verification of the method. The Bullwinkle uses skid-steering for mobility of the base, and has a manipulator with five revolute joints. Fig. 2(b) shows the dimension of the base. The values of the kinematic parameters of the manipulator are  $d_1 = 223\text{mm}$ ,  $d_4 = 275\text{mm}$ ,  $a_1 = 63.5\text{mm}$ ,  $a_2 = 290\text{mm}$ ,  $a_3 = 252.7\text{mm}$ ,  $a_4 = 49.23\text{mm}$ . On the Fig. 3, the control point  ${}^j\mathbf{p}_i$  represents the  $i$ -th control point in the  $j$ -th way pose. The superscript represents the order of the way pose, and the subscript represents the order of the control point in the way pose.

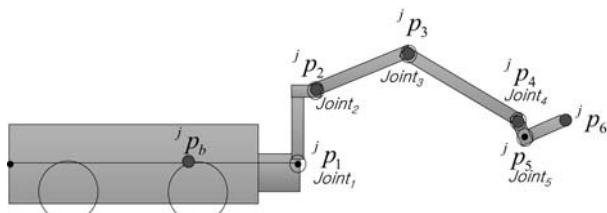


(a) The robot Bullwinkle has skid-steered mobile base and five joint manipulator



(b) Dimension of the base in inches

**Fig. 2.** The robot Bullwinkle and dimension of the base



**Fig. 3.** Mobile manipulator Bullwinkle and the control points on the body.  ${}^j\mathbf{p}_i$  represents the  $i$ -th control point on the robot in its  $j$ -th way pose.

The elastic force  ${}^j\mathbf{e}\mathbf{f}_i$  exerting to the control point  ${}^j\mathbf{p}_i$  is as follows.

$${}^j\mathbf{e}\mathbf{f}_i = k_c \left( \frac{d_{j-1,j}}{d_{j-1,j+1}} ({}^{j+1}\mathbf{p}_i - {}^{j-1}\mathbf{p}_i) - ({}^j\mathbf{p}_i - {}^{j-1}\mathbf{p}_i) \right)$$

In the equation, the vectors  ${}^{j-1}\mathbf{p}_i$ ,  ${}^j\mathbf{p}_i$ , and  ${}^{j+1}\mathbf{p}_i$  represent the location of the  $i$ -th control point in its  $(j - 1)$ -th,  $j$ -th, and  $(j + 1)$ -th way pose respectively.  $d_{j-1,j}$  is the distance between  ${}^{j-1}\mathbf{p}_i$  and  ${}^j\mathbf{p}_i$ .  $d_{j-1,j+1}$  is the distance from  ${}^{j-1}\mathbf{p}_i$  to  ${}^{j+1}\mathbf{p}_i$  via  ${}^j\mathbf{p}_i$ . The elastic force tends to align the way poses from the initial to final location as straight as possible. Though each control point is likely to align straight throughout the way poses, the kinematic structure of the joint and link limits the straightness of the trajectory. The forces on the control points compromise and result in elastically stretched trajectory.

While the elastic force connects the initial pose to the final pose through way poses, repulsive force also works to push the trajectory away from obstacles. In our application, force by artificial potential field is used as the repulsive force. The following shows the repulsive potential field  $V_{rep}({}^j\mathbf{p}_i)$  and repulsive force  ${}^j\mathbf{rf}_i$  at the control point  ${}^j\mathbf{p}_i$ .

$$V_{rep}({}^j\mathbf{p}_i) = \begin{cases} \frac{1}{2}k_r(d_r - d({}^j\mathbf{p}_i))^2, & \text{if } d({}^j\mathbf{p}_i) < d_r \\ 0, & \text{otherwise} \end{cases}$$

$${}^j\mathbf{rf}_i = -\nabla V_{rep}({}^j\mathbf{p}_i) = k_r(d_r - d({}^j\mathbf{p}_i)) \frac{{}^j\mathbf{p}_i - \mathbf{p}_o}{\|{}^j\mathbf{p}_i - \mathbf{p}_o\|}$$

In the equation,  $d_r$  is the radius of the range under the influence of the repulsive force.  $d({}^j\mathbf{p}_i)$  is the shortest distance between the control point  ${}^j\mathbf{p}_i$  and the location  $\mathbf{p}_o$  of an obstacle relevant to collision.  $k_r$  is the coefficient for the repulsive force to the distance. The net force  ${}^j\mathbf{f}_i$  exerting on the control point  ${}^j\mathbf{p}_i$  is the sum of the elastic force and the repulsive force.

$${}^j\mathbf{f}_i = {}^j\mathbf{ef}_i + {}^j\mathbf{rf}_i, \quad \text{for } i = 1, 2, \dots, 6$$

For notational simplicity, in the following sections, we deleted the superscript  $j$  which denotes a way pose if we only need to distinguish control points in a way pose. If there is no superscript in notations, it is supposed that all the control points are located in a way pose, for example in the  $j$ -th way pose.

### 2.3 Force and Torque Adjusting the Way Pose

The modification of the trajectory is accomplished by deforming the way poses. The deformation of the way pose means change of the configuration variables. The configuration variables for the base are  $\{x_b, y_b, \theta_b\}$ , and those for manipulator are  $\{\theta_1, \theta_2, \dots, \theta_5\}$ . The rotational or translational deformation of the robot configuration is proportional to the joint torque or joint force. The joint torque or force is due to the force and torque on the control points generated by the elastic force and potential field force.

To find the joint torque acting on the joints, we use the Jacobian of the manipulator.

$$\tau_i = \mathbf{J}_i^T \mathbf{f}_{i+1}$$

Here, the vector  $\tau_i = [\tau_1, \tau_2, \dots, \tau_i]^T$  consists of the torques on the joints  $1, 2, \dots, i$  and the vector  $\mathbf{f}_{i+1} = [f_{i+1,x}, f_{i+1,y}, f_{i+1,z}]^T$  represents the force acting on the

control point  $\mathbf{p}_{i+1}$ . The vector  $\mathbf{f}_{i+1}$  is the sum of the elastic force and repulsive force acting on the control point  $\mathbf{p}_{i+1}$ . The matrix  $\mathbf{J}_i$  is the  $3 \times i$  Jacobian matrix satisfying the following relationship,

$$\dot{\mathbf{p}}_{i+1} = \mathbf{J}_i \dot{\mathbf{q}}_i$$

where  $\mathbf{p}_{i+1} = [p_{i+1,x}, p_{i+1,y}, p_{i+1,z}]^T$  is the position vector of the  $(i+1)$ -th control point and  $\mathbf{q}_i = [\theta_1, \theta_2, \dots, \theta_i]^T$  is the joint variable vector.

To find the torque due to the forces exerting on all the control points  $\mathbf{p}_1, \dots, \mathbf{p}_6$ , it is required to sum all the torque vectors such that,

$$\tau = \sum_{k=1}^5 \begin{bmatrix} \tau_k \\ \mathbf{0} \end{bmatrix}_{5 \times 1}.$$

where  $\tau = [\tau_1, \tau_2, \dots, \tau_5]^T$  consists of all the five torques around the five manipulator joints.

For the mobile base, it is required to find the force and torque around the point  $\mathbf{p}_b = [p_{b,x}, p_{b,y}, p_{b,z}]^T$ . The force exerting on  $\mathbf{p}_b$  is the sum of the elastic and repulsive force on the control points  $\mathbf{p}_i$ , ( $i = 1, 2, \dots, 6$ ) and the elastic and repulsive force at the control point  $\mathbf{p}_b$  which is located on the mobile base itself. If we represent the elastic force and repulsive force on the point  $\mathbf{p}_b$  as  $\bar{\mathbf{f}}_b = [\bar{f}_{b,x}, \bar{f}_{b,y}, \bar{f}_{b,z}]^T$ , then the net force  $\mathbf{f}_b = [f_{b,x}, f_{b,y}, f_{b,z}]^T$  is found to be as the following.

$$\mathbf{f}_b = \bar{\mathbf{f}}_b + \sum_{k=1}^6 \mathbf{f}_i$$

Here, we ignore the force in the  $z$  direction since the mobile base doesn't have any mobility in the  $z$  direction. The torque  $\tau_b = [\tau_{b,x}, \tau_{b,y}, \tau_{b,z}]^T$  around  $\mathbf{p}_b$  is due to the force on  $\mathbf{p}_i$ , ( $i = 1, 2, \dots, 6$ ), and it is expressed as the following.

$$\tau_{b,x} = 0, \tau_{b,y} = 0, \tau_{b,z} = \sum_{k=1}^6 (p_{i,x} f_{i,y} - p_{i,y} f_{i,x})$$

We ignore the torque around the direction  $x$  and  $y$  since the mobile base doesn't have any rotational mobility around the direction  $x$  and  $y$ .

## 2.4 Finding the Way Poses

The way poses are updated every sampling time. The amount of deformation of the way pose for the manipulator is proportional to the force or torque applied to the joint. In our case, all the joints of the manipulator are revolute, so the rotation angles are found to be the followings.

$$\delta\theta_i = \alpha_i \tau_i, \quad (i = 1, 2, \dots, 5)$$

In this equation,  $\alpha_i$  is the coefficient for the rotation of the  $i$ -th joint to the joint torque  $\tau_i$ . For the mobile base, the rotation angle and displacement are as the followings.

$$\delta x_b = \beta_{b,x} f_{b,x}, \quad \delta y_b = \beta_{b,y} f_{b,y}, \quad \delta\theta_b = \alpha_{b,z} \tau_{b,z}$$

Here,  $\beta_{b,x}$  and  $\beta_{b,y}$  are the coefficients for the translation of the base to the force in the  $x$ -direction and  $y$ -direction respectively.  $\alpha_{b,z}$  is the coefficient for the rotation of the base around the  $z$ -direction to the torque  $\tau_{b,z}$ .

## 2.5 Tracing the Trajectory Through the Way Poses to the Goal Pose

The way poses represent the trajectory of the motion. The method drives the robot through the way poses in sequence. Once the robot reaches a way pose, it then moves toward the next way pose. As the robot moves toward the goal, the number of way poses passed through increases and the number of remaining way poses waiting to be passed through decreases. This process repeats until the robot passes the last way pose and reaches the final goal pose.

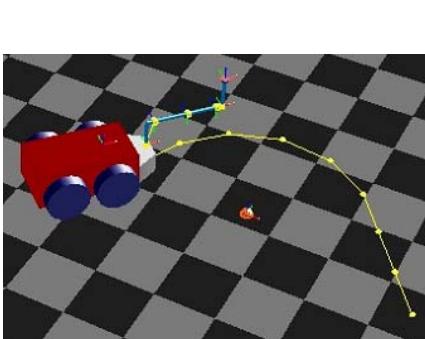
For easy application, the robot is considered to be at its goal pose if each of the configuration variables falls within a range which is set around its goal pose value with clearance small enough. This is expressed in the following inequalities.

$$\begin{aligned} |x_b - x_{b,g}| &< \varepsilon_{x,b}, \quad |y_b - y_{b,g}| < \varepsilon_{y,b}, \quad |\theta_b - \theta_{b,g}| < \varepsilon_{\theta,b} \\ |\theta_i - \theta_{i,g}| &< \varepsilon_{\theta,i}, \quad (i = 1, 2, \dots, 5) \end{aligned}$$

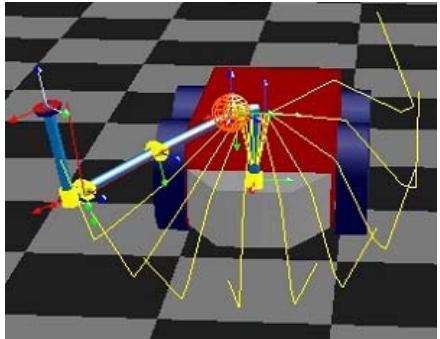
Here,  $\{x_{b,g}, y_{b,g}, \theta_{b,g}\}$  and  $\{\theta_{1,g}, \theta_{2,g}, \dots, \theta_{5,g}\}$  are the goal pose of the base and manipulator respectively.  $\varepsilon_{x,b}$ ,  $\varepsilon_{y,b}$ ,  $\varepsilon_{\theta,b}$ ,  $\varepsilon_{\theta,i}$ , ( $i = 1, 2, \dots, 5$ ) are the clearances for goal pose reach.

## 3 Simulations and Experiments

The proposed method is verified through simulations and an experimental implementation. The Fig. 4 and 5 show the results for simulations. In the Fig. 4(a),

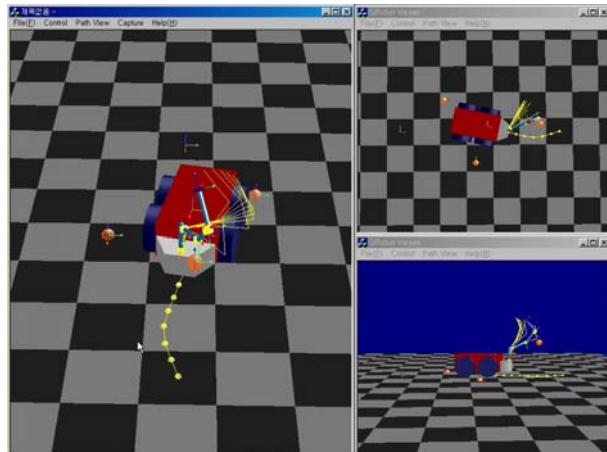


(a) Avoidance motion of the base

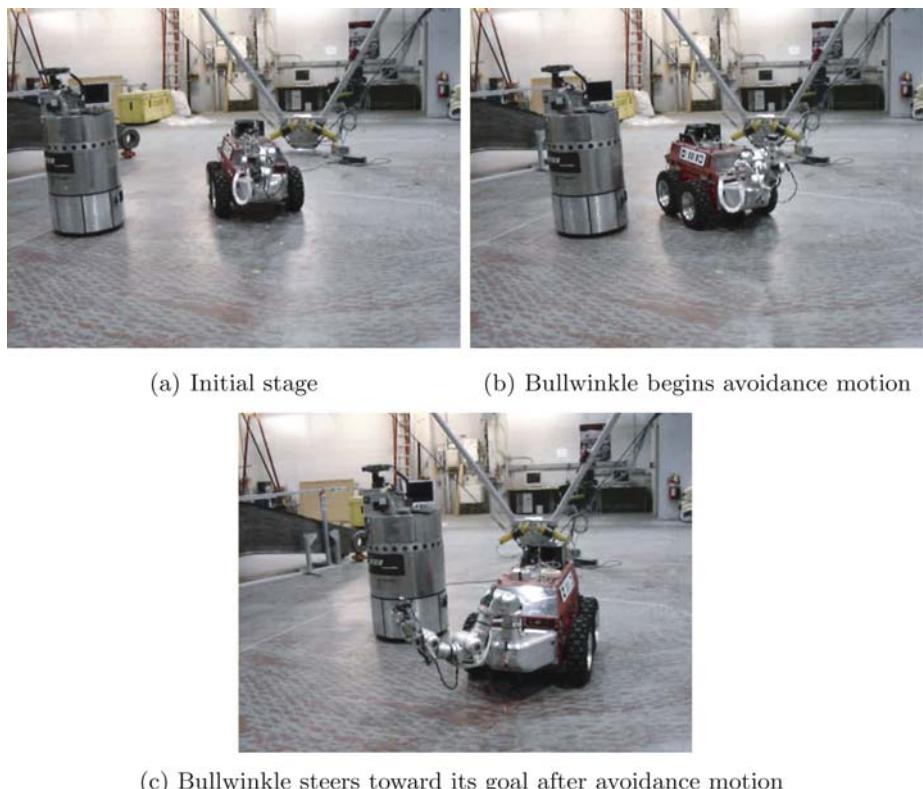


(b) Avoidance motion of the manipulator

**Fig. 4.** Simulation results. (a) Since the obstacle is on the floor, only the base motion is needed for collision avoidance. (b) The obstacle is far over the floor and the base doesn't need to care about collision.



**Fig. 5.** Obstacle avoidance motion of the base and manipulator. The motion is captured in three different directions. There are three obstacles. Some of the obstacles affect the motion of the base as well as the manipulator.



**Fig. 6.** Result of the experiment. The Xavier is used as a moving obstacle.

the obstacle is on the floor, and avoidance motion of only the base suffices collision avoidance. The manipulator is out of the range of potential field force from the obstacle. Likewise, in Fig. 4(b), the obstacle doesn't exert repulsive force to the base, and only the manipulator works for obstacle avoidance. Only the way poses of the arm lower down for obstacle avoidance under the repulsive potential field force.

In Fig. 5, there are 3 obstacles, and the mobile base as well as the manipulator works for obstacle avoidance. When the obstacles break into the queue of way poses, the way poses stretches out to avoid the obstacles. Later when the obstacles recede, the way poses restore back subject to the elastic force across the control points.

Fig. 6 shows the results of the experiment. The robot Xavier [11] is used as a moving obstacle. The Bullwinkle is commanded to go to a goal pose which is 5m ahead of the initial heading. Xavier moves into the Bullwinkle's path from the upper right side of the Bullwinkle with the speed of 5cm/s. For safety, the maximum speed of the Bullwinkle base is set to 10cm/s.

## 4 Conclusions

A method for a mobile manipulator to avoid obstacles is described. The method uses elastic force and potential field force to adjust the trajectory for obstacle avoidance. The method has the following features.

1. The approach allows real time application even for mobile manipulator which has considerably high degrees of freedom. It results in smooth and efficient collision free trajectory for a mobile manipulator.
2. The method adjusts robot trajectory every sampling time, and avoids obstacles which are moving as well as stationary.
3. Though it uses potential field force for repulsion from obstacles, it also uses elastic force to keep in the robot shape. Therefore, the robot way poses restore and results in smooth and efficient trajectory toward the goal pose when obstacles move away.

For further improvement, the following should be addressed.

1. As the pose of the obstacles changes and the collision free path becomes very complicated, rerouting for simple path is necessary.
2. The number of way poses should be determined adaptively to the task and environment. If the obstacles are near the robot, it should be close, and if the obstacles are far away it can be sparse.

## Acknowledgement

This study was supported by research funds from Chosun University, 2005.

## References

1. Khatib, O.: Real-time obstacle avoidance for manipulators and mobile robots. *The Int. J. Robotics Research*, Vol. 5, No. 1. (Spring 1986) 90–98
2. Boernstein, J., Koren, Y.: The vector field histogram-fast obstacle avoidance for mobile robots. *IEEE Transaction on Robotics Automation*, Vol. 7, No. 3. (June 1991) 278–288
3. Batavia, P., Nourbakhsh, I.: Path Planning for the Cye Personal Robot. *Proc. International Conference in Intelligent Robots and Systems(IROS 2000)*, Vol. 1. (Oct. 2000) 15–20
4. Simmons, R.: The Curvature-Velocity Method for Local Obstacle Avoidance. *Proc. 1996 IEEE International Conference on Robotics and Automation*, Minneapolis MN (April 1996) 3375–3382
5. Ko, N. Y., Simmons, R.: The Lane-curvature Method for Local Obstacle Avoidance. *Proc. International Conference in Intelligent Robots and Systems(IROS 1998)*, Victoria, B.C, Canada (Oct. 13-17 1998) 1615–1621
6. Brock, O., Khatib, O.: High-Speed Navigation Using the Global Dynamic Approach. *Proc. of the IEEE Int. Conf. on Robotics and Automation*, Vol. 1, (May 1999) 341–346
7. Fox, D., Burgard, W., Thrun,S.: The dynamic windows approach to collision avoidance. *IEEE Robotics and automation magazine*, 4(1) (March 1997)
8. Quinlan S., Khatib, O.: Elastic Bands: Connecting path planning and control. *Proc. 1993 IEEE Int. Conf. on Robotics and Automation*, Vol. 2. (1993) 802–807
9. Siegwart, R. et al.: Robox at Expo.02: A large scale Installation of Personal Robots. *Journal of Robotics and Autonomous Systems*, Vol. 42. (2003) 203–222
10. Brock, O. : Generating robot motion: The integration of planning and execution. PhD thesis, Stanford Univ. (1999)
11. <http://www.cs.cmu.edu/~Xavier/>

# A Hybrid Architecture Combining Reactive Plan Execution and Reactive Learning

Samin Karim<sup>1</sup>, Liz Sonenberg<sup>1</sup>, and Ah-Hwee Tan<sup>2</sup>

<sup>1</sup> Department of Information Systems

University of Melbourne

111 Barry St Carlton 3053, Melbourne, Australia

<sup>2</sup> School of Computer Engineering

Nanyang Technological University

Nanyang Avenue, Singapore 639798

**Abstract.** Developing software agents has been complicated by the problem of how knowledge should be represented and used. Many researchers have identified that agents need not require the use of complex representations, but in many cases suffice to use “the world” as their representation. However, the problem of introspection, both by the agents themselves and by (human) domain experts, requires a knowledge representation with a higher level of abstraction that is more ‘understandable’. Learning and adaptation in agents has traditionally required knowledge to be represented at an arbitrary, low-level of abstraction. We seek to create an agent that has the capability of learning as well as utilising knowledge represented at a higher level of abstraction.

We firstly explore a reactive learner (FALCON) and reactive plan execution engine based on BDI (JACK) through experiments and analysis. We then describe an architecture we have developed that *combines* the BDI framework to the low-level reinforcement learner and present promising results from experiments using our minefield navigation domain.

## 1 Introduction

For many applications, agents require the ability to learn and adapt, as well as utilise an adequate knowledge representation that offers engineering benefits by virtue of a high-level, semantic representation. One of the major tradeoffs between an agent that represents its knowledge using a low level of abstraction, to an agent that represents its knowledge using a high level of abstraction, is in the type of representation used. ‘Low-level’ agents are more adept at learning and reactive behaviour [14], but usually lack a semantically rich knowledge representation, whereas ‘high-level’ agents have a semantically rich knowledge representation, but traditionally lack adequate learning capabilities that are grounded to this knowledge base. A hybrid architecture that combines these disparate knowledge representations may be the solution to achieve a unified agent that can learn as well as represent knowledge at a higher level of abstraction that carries greater semantic clarity.

In this paper we describe and evaluate the behaviours of a high-level reasoner, JACK (based on the BDI framework [3]), and a reinforcement learner (RL), FALCON (based on an extension of Adaptive Resonance Theory (ART) [5]), in order to demonstrate the aforementioned tradeoffs and to set the stage for the development of a new hybrid architecture that *combines* these approaches. The symbolic approach based on BDI has advantages in that it has a more precise, semantically rich knowledge representation that promotes effective introspection over the knowledge. Conversely, the RL approach (FALCON) can extract the equivalent control knowledge without manual intervention, but is disadvantaged by a less accessible representation (ie. low-level of abstraction).

The approaches that we employ have their roots in cognitive science. The Belief-Desire-Intention (BDI) approach derives from folk psychology, and provides a platform for human modelling and logical reasoning. BDI systems encode goal directed behaviours by using action sequences, or plans, derived from domain expert knowledge, usually specified at design time, about the task domain.

In modern cognitive science, many have held the view that cognition is a process deeply rooted in the body's interaction with the world [4]. The FALCON model [17], an instance of reinforcement learning [10] that stems from this philosophy, learns a policy by creating *cognitive codes*, which essentially are *rules* that associate current states to actions that lead to desirable outcomes (rewards). The strategy is similar to that adopted by the complementary reinforcement backpropagation algorithm [1].

Thus we see complementary approaches to solve a task. On one hand we have an *explicitly represented* procedural knowledge base (plans) utilised by the high-level BDI agent, and which are defined at design-time by the domain expert. On the other hand we observe an *implicitly represented* knowledge base accessed by the low-level FALCON agent and specified during repeated execution and experience gathering trials. By studying these different approaches to solving the same task, and analysing the tradeoffs of each, we should be well positioned to propose a *hybrid architecture* that combines these two dichotomous approaches.

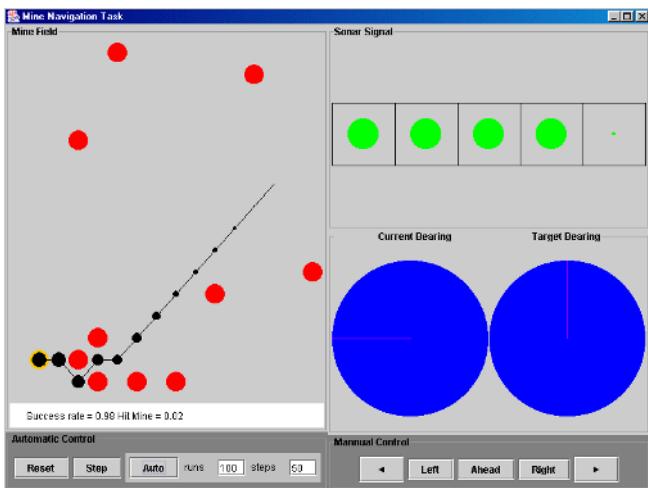
To begin our investigation we have chosen a relatively simple task, which we label the 'minefield navigation domain'. It is similar to the one developed at the Naval Research Laboratory (NRL) [7], and involves an autonomous vehicle (AV) learning to navigate through obstacles to reach a stationary target (goal) within a specified number of steps. Our study indicates that, separately, FALCON and BDI agents can achieve very good performance in terms of success rates. However, the knowledge used by the two agents is vastly different. This points to many issues and challenges in developing an integrated architecture that performs both deliberative planning and reactive skill learning.

The rest of the paper is organised as follows. Section 2 provides details on the minefield navigation task domain. Section 3.1 gives a summary of the reactive FALCON model for this domain and presents the experimental results. Section 3.2 presents the BDI approach to the minefield navigation problem and associated experimental results. In section 4, a preliminary design and implementation that combines these two approaches is discussed and analysed. Outcomes and

discussion of experiments we ran using all three systems is covered in section 5. Section 6 describes related work, and finally section 7 summarises and discusses outcomes, highlights more general issues and points to future work.

## 2 The Minefield Navigation Task

We chose a domain that has sufficient regularity to make unsupervised learning and a priori defined plans viable. At the same time, there should be sufficient uncertainty and complexity to make re-evaluation and/or change of plans/actions necessary. The *minefield navigation task* (Figure 1) requires an autonomous vehicle (AV) starting at a randomly chosen position in the field to navigate through the minefield to a randomly selected target position in a specified time frame without hitting a mine. A trial ends when the system reaches the target (success), hits a mine or runs out of time (failure).



**Fig. 1.** The minefield navigation simulator

The system has a rather coarse sensory capability with a 180 degree forward view based on five sonar sensors. For each direction  $i$ , the sonar signal is measured by  $s_i = \frac{1}{1+d_i}$ , where  $d_i$  is the distance to an obstacle (that can be a mine or the boundary of the minefield) in the  $i$  direction. Other input attributes include the range and the bearing of the target from the current position. In each step, the system can choose one of five possible actions: `MoveLeft`, `MoveFrontLeft`, `MoveFront`, `MoveFrontRight`, and `MoveRight`.

The complexity for learning the minefield navigation problem is largely determined by the dimension of the sensory(state) representation. The state space is  $S^5 \times B$  where  $S = [0, 1]$  is the range of the sonar signals in the five directions and  $B = \{0, 1, \dots, 7\}$  is the set of possible target bearings.

### 3 Two Different Approaches

We initially employed two different approaches to solve the minefield navigation task, which are described in the following sections.

#### 3.1 The Reactive Learning Approach: FALCON

FALCON is an extension of predictive Adaptive Resonance Theory (ART) networks [6,16]. For reinforcement learning, FALCON makes use of a 3-channel architecture, consisting of a sensory field  $F_1^{c1}$  for representing the current state, an action field  $F_1^{c2}$  for representing the available actions, a reward field  $F_1^{c3}$  for representing the values of the feedback received from the environment, and a cognitive field  $F_2^c$  for encoding the relations among the values in the three input channels.

The reactive FALCON model acquires an action policy directly by learning the mapping from the current states to the desirable actions. The interested reader is directed to [17] for details on FALCON. It is sufficient to highlight here that FALCON achieves standard RL outcomes. As we will see later in section 5, FALCON is a good example of how low-level learners can solve complex tasks without the extraction and use of higher level abstract knowledge.

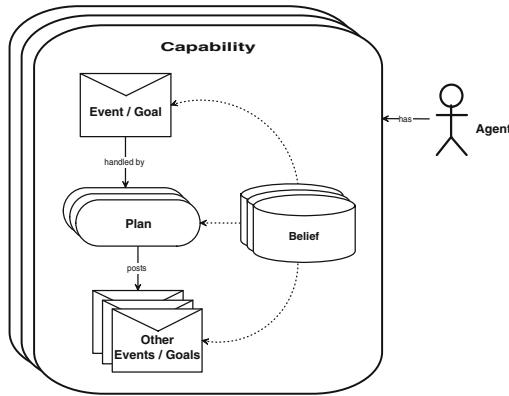
#### 3.2 The Plan Execution Approach: BDI

The basic definition of BDI agency is described in [13]. We chose to use JACK as our agent-oriented programming platform, which supports BDI concepts and functionality. Architecturally, JACK consists of several constructs: agents, capabilities, events, plans and beliefs (see Figure 2). *Agents* are at the highest level of abstraction, and represent entities with *autonomous behaviour* within the system. Plans are executed when a relevant event occurs, which are posted as a result of many reasons, mainly from goals becoming active, a percept trigger, or from within other plans. Plans are specified by a domain expert at design time to negotiate general scenarios, as was the case in our implementation for the minefield domain. A detailed description of the JACK system can be found in [11].

## 4 A Combined Approach: The FALCON-BDI Hybrid

#### 4.1 Towards a Unified, Learning Agent

An important goal of agent research is to develop an architecture that is able to learn and operate in real time. The RL mechanism of FALCON allows it to adapt its functional behaviour reasonably well but has inherent difficulties in representing and acquiring high level abstract knowledge such as those possible in BDI systems. Moreover, complex constructs such as context, action sequences and goals are not explicit features of low-level learners such as FALCON. Conversely, the BDI approach does not suffer from a nonintuitive representation such as the



**Fig. 2.** JACK agent-programming language constructs

RL approach. However, the BDI approach requires significant domain expert input and is not inherently conducive to learning techniques due to its high-level representation.

Due to these inherent limitations of each of these approaches, *combining* the plan-based, BDI approach with FALCON is an area we have tackled as part of an ongoing research effort. An approach is to have a *layered system*, with FALCON at the bottom level and the BDI-type system at a higher level, and their outputs appropriately combined. To this end, one possibility is to use the bottom level to *abstract concepts* that are used in the higher, BDI level. Abstraction of concepts from continuous feature values helps to produce higher level *symbolic knowledge*. Two different approaches were explored to achieve the abstraction of concepts. One approach was to abstract certain states to symbolic knowledge.

Another way to combine the reactive and plan-based approaches is to incorporate learning into BDI systems. JACK (BDI) plans are designed specifically for the task they were designed for, and can achieve almost perfect performance. However, in a less familiar environment, learning capabilities become desirable. There are several possible ways learning can take place in a BDI architecture.

1. Existing/predefined plans in a BDI architecture can be refined through the agent's subsequent experience with the environment. For example, the context of plans can be refined based on a rewarding strategy. The execution and actions of plans can be refined in a similar manner.
2. New plans can be learned in a BDI architecture. There are different approaches that could achieve this, such as: predefined knowledge that facilitates the learning of new plans (eg. pre-wired knowledge can be used to prime the learning process in, say, monitoring subsets of attributes or situations), or a low-level learner whose action recommendations are collated into plans.

The use of plans as opposed to rules (codes) results in a knowledge representation that has engineering advantages in that it is more comprehensible to humans.

Also, *policy compression* [14] is attained using plans (ie. the same information occupies less storage space). The approach described in the second point above is the basis for the architecture presented in the following sections.

## 4.2 Plan Generation from FALCON Codes

In this section we present our first step towards a unified architecture that merges the disparate approaches described in sections 3.1 and 3.2. The model consists of a BDI top-level (JACK) and RL bottom-level (FALCON). The resulting architecture *generates plans* via the BDI top-level from rules learned by the bottom-level. The crucial element of the system is that *a priori information* (specified by the domain expert) is used by the BDI top-level to assist in the generation of plans.

For our minefield navigation domain, plans consist of sequences of actions. The human domain expert supplies *a priori* data in the form of two agent goals that are associated to information pertaining to these goals. This *a priori* data can be likened to *clues* used by the BDI top-level for plan generation. The essential idea is that goals encapsulate *a priori* information that allows the BDI module to *converge* relevant FALCON action recommendations into plans:

1. **Goal:** Avoid mines.

**Clue:** Consider when the agent is adjacent to a mine, meaning one or more sonar distance is equal to one.

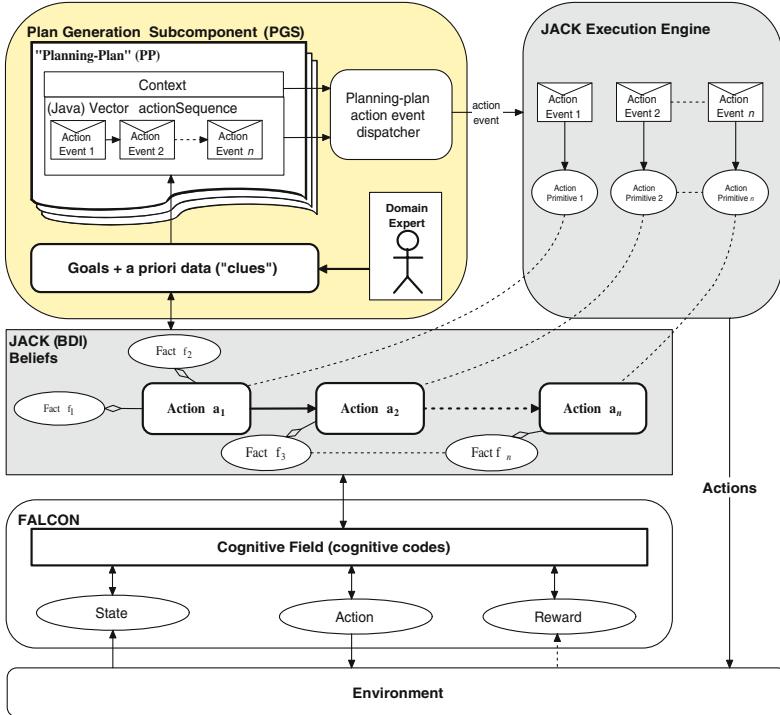
2. **Goal:** Reach target.

**Clue:** Consider when the agent is moving towards the target, meaning the distance to target is reducing from the previous step.

Plans will start being generated when these relevant conditions occur, as governed by the goals which encapsulate them. The ultimate effect is that actions are ‘recorded’ into plans. For the first goal, plans will start being generated when the agent is ‘close to’ a mine (ie. sonar reading indicates mine is adjacent to the agent). Similarly for the second goal, once the agent begins reducing its distance from the target, plans relevant to this goal will start being generated. So, after multiple runs, the BDI module will have accumulated a plan library of *condition* → *plan* profiles that can *subsume* the FALCON module when, for example, the utility (what ever this measurement may be) of the BDI plan is higher than the utility of the FALCON action. Plans which are executed repeatedly with success or failure will have their *confidence values* reinforced/penalised accordingly. For this purpose, the rewards in FALCON are utilised for adjusting the relevant plan’s confidence values.

## 4.3 The Design

The architecture is illustrated in Figure 3. The top-left box (highlighted) is the critical module of the architecture, the *plan generation subcomponent* (PGS), which integrates FALCON to BDI. PGS generates plans by utilising *a priori* data in the form of “clues” that are part of specified goals. The actions, which are part



**Fig. 3.** FALCON–BDI architecture

of plans, are *grounded* to symbolic knowledge in the BDI belief base. Actions can either be performed by the BDI level or the FALCON level.

Algorithm 1 outlines the overall heuristic for plan generation and interaction between the BDI and FALCON modules. In this algorithm, plans are generated when an existing plan is not currently being executed and when FALCON actions are executed. When a plan is executing, FALCON actions are suppressed and plans are executed to completion. Actions and plans are selected on the basis of respective utility functions. Also, note that  $Utility(a) = reward(a)$ , and  $Utility(p) = Utility(a_l) - Utility(a_1)$  (where  $a_l$  is the last step of a plan and  $a_1$  is the first step of a plan - see step 6 of Algorithm 1).

## 5 Experiments and Results

We ran experiments using the domain described in section 2. All the systems described in the preceding sections were trained for 1000 trials, where each trial involved the random placement of 10 mines, the agent starting point and the target within a  $16 \times 16$  grid. In each trial, the AV repeats the cycles of sense, act (and learn in the case of the FALCON and FALCON-BDI systems), until it reaches the target, hits a mine, or exceeds 30 sense-act(-learn) cycles. The same

---

**Algorithm 1.** The PGS plan generation and FALCON-BDI integration heuristic

---

**Require:**  $n > \text{threshold}_1$ , where  $\mathcal{C}$  is the set of FALCON cognitive codes, and  $\mathcal{C} = \{C_1, C_2, \dots, C_n\}$

- 1: **for** each consecutive state-action tuple occurrence,  $(s, a_f)$ , generated by FALCON **do**
- 2:     Update set of ‘active goals’,  $\mathcal{G}$  (ie. goals relevant to current situation denoted by  $(s, a_f)$ ).
- 3:     **for** each active goal  $G_i \in \mathcal{G}$  **do**
- 4:         Calculate  $Utility(a_f)$  and  $Utility(p_s)$
- 5:         **if** not already executing a plan **then**
- 6:             Select a plan  $p_s : p_s = \{a_1, a_2, \dots, a_c, \dots, a_l\}$ , where  $a_c$  is the current step of  $p_s$ , and  $a_l$  is the last step of  $p_s$
- 7:         **end if**
- 8:         **if**  $Utility(p_s) > Utility(a_f) \vee$  already executing a plan **then**
- 9:             Execute  $a_c$ , where  $a_c \in p_s$
- 10:          **else if**  $Utility(p_s) < Utility(a_f)$  **then**
- 11:             **if** there is an existing plan,  $p_e$ , in the *CustomPlan* library, which  $(s, a_f)$  can be appended to **then**
- 12:                 Append  $a_f$  to the end of  $p_e$
- 13:             **else**
- 14:                 Create a new *CustomPlan*,  $p_{new}$
- 15:                 Add  $a_f$  as first step of  $p_{new} : p_{new} = \{a_f\}$
- 16:             **end if**
- 17:             Execute  $a_f$
- 18:         **end if**
- 19:     **end for**
- 20: **end for**

---

**Table 1.** The success rates and hit mines ratio of the different systems with stationary and randomly moving mines. Number of trials = 1000 and maximum step allowance = 30 steps.

System	Mine Movement	
	Stationary	Random
BDI	100% / 0.0%	85.7% / 10.7%
FALCON	89.4% / 8.5%	80.7% / 16.8%
FALCON-BDI hybrid	88.6% / 9.2%	78.7% / 18.8%

set of 1000 randomly generated minefield configuration was used across all experiments.

These results show that the BDI implementation, as expected, yielded superior performance due to the domain expert defined control knowledge base. The FALCON implementation followed BDI, and marginally outperformed the FALCON-BDI hybrid system. The results of FALCON and FALCON-BDI hybrid systems, however, are very similar, which shows much promise. We also ran the same experiments with random moving mines. In this instance, the agent must

<b>Goal:</b> “Avoid Mines”
<b>Context:</b> Mines to ‘hard’ left, left, and straight ahead; Target bearing = North-East, Agent’s bearing = North-West
<b>Plan body/Action sequence:</b> Right → Hard right → Right → Straight → Straight → Straight

**Fig. 4.** Example PGS-generated plan

negotiate mines which move randomly within the grid. This presents the agent with a more complex task and hence proved to be considerably more difficult to solve, as evidenced by the performance decrease across all systems.

It is worth reiterating here that quantitative results alone are not the only component worth considering. The quality of plans and the aforesaid engineering advantages of the BDI plan representation (refer to section 4) are favourable outcomes not attained by low-level representations such as the rule-based representation of FALCON. An example plan generated by PGS is shown in Figure 4. Goals and the action sequence, or plan, are features of the BDI representation not present in the rule-based representation. To represent the same information in a rule-based representation would require separate rules for each step of the plan sequence. Unlike the BDI representation, the relationship between each step in a sequence of actions is not implicit in the rule-based representation.

## 6 Related Work

Sun [14] described a two-level model, known as CLARION, for learning reactive plans and extracting plans from reinforcement learners. The first three layers of the bottom level form a backpropagation network learning and computing Q-values and the fourth layer (the top level with only one node) determines probabilistically the action to be performed. CLARION is an example of a hybrid architecture consisting of explicit (high-level) and implicit (low-level) representations.

Independently, Heinze et al [9] demonstrated the synergistic coupling of a machine learning architecture (CLARET) with a BDI top-layer in a hybrid architecture that achieves intention recognition of aircraft. CLARET was used to recognise spatial trajectories of aircraft and other observable objects. The BDI layer processed these observations with higher-level, goal-directed reasoning.

Sun and Sessions [15] mention that scalability is an issue for classical reinforcement learners. For a relatively complex domain, abstraction of attributes or function approximators are often needed. As Sun and Sessions’s system made use of a radial basis function network or multilayer perceptron network, the system may not be able to learn and operate in *realtime*. However, a key strength of Sun and Sessions is the capability to perform *probabilistic planning*.

Also taking a bottom-up approach, the *DyKnow* framework by Doherty and Heintz [8] was essentially a signal-to-symbol transformer that, amongst other features, continually monitored perceived signals and created *higher-level cognitive*

*objects* facilitated by domain expert defined hypothesis tests. Temporal knowledge (plans) were among these higher level abstractions acquired, and are maintained as *chronologies*.

The Soar architecture [12] also achieves planning from a machine learning bottom-level by way of the well-known chunking mechanism. The PRODIGY architecture [18] combines a planning and reasoning framework with learning capability. The learning operates in three distinct ways: improving plan search efficiency, improving/adding planning operators (which are generalised atomic planning actions), and improving plan quality according to a predetermined metric.

These approaches and architectures have the common goal (amongst other goals) of *symbol grounding* high-level representations to low-level representations. Similarly, in this paper we presented an approach that *ties* a low-level RL module to a higher-level BDI module, essentially for the purpose of learning plans. However, the architecture presented in this paper is unique to current planning-learning hybrid systems in the literature with regard to the way plans are generated and with the assistance of domain expert knowledge, as well as the fact it is acquired in real-time.

## 7 Conclusion and Future Work

We have investigated ways in which different cognitively styled agents using knowledge representations of varying levels of abstraction can be combined into a hybrid architecture. Using a ‘minefield navigation’ domain, we firstly compared the performance and characteristics of a low-level reactive learner, FALCON, which is based on a reinforcement learning technique, to that of a high-level plan execution engine, JACK, that is based on the BDI framework. Secondly, we studied the way in which knowledge is represented and utilised in these two approaches, and then finally suggested a hybrid architecture that combines these two approaches to form a unified agent model.

The experiments reported have been designed to tease out issues relating to what knowledge to use and represent in our chosen domain. We have looked at four levels of knowledge representation: (i) the BDI approach (JACK) that supports the most expressive account of domain knowledge - procedures/plans are provided by a domain expert, in a rich language where that expertise “guarantees” task success; (ii) the reinforcement learner (FALCON) that works on primitive data where the system designer has provided minimal explicit guidance to the run-time engine, relying on the domain regularity to inform the learning, and (iii) a *hybrid architecture* that combines the plan-based and rule-based, or more generically, high-level and low-level representations, respectively, to achieve a system that is able to learn whilst maintaining the learned knowledge in a more abstract, higher-level BDI representation that is understandable to humans.

It is worthwhile noting that the hybrid architecture, presented in section 4, demonstrates a generalisable ‘interaction’ between different representations.

In this paper we present a BDI top-level that generates plans derived from an RL bottom-level. However, the bottom-level could quite easily be another type of module rather than the RL FALCON module (or more generally, a learning module) that is based on a representation with a similar low level of abstraction as FALCON. Furthermore, the application of this architecture goes beyond mobile robotics. For instance, plan generation, or sequence learning, has application in the biology domain [2].

It is hoped that further development of the hybrid architecture, whose preliminary design and evaluation is presented in this paper, will yield even better results in the future. Among consideration are improved plan selection, generation and management heuristics, as well as a more defined integration between the bottom and top levels. There are many avenues to consider in future. Diverse low-level learners other than FALCON will be considered, and as a major variation, the use of a human controller in place of the bottom-level will also be explored. Different knowledge rich domains where such a hybrid system would yield engineering and performance advantages will also be developed and investigated.

## References

1. D. Ackley and M. Littman. Generalizaion and scaling in reinforcement learning. In *Advances in Neural Information Processing Systems 2*, pages 550–557, 1990.
2. P. Baldi, S. Brunak, P. Frasconi, G. Pollastri, and G. Soda. *Bidirectional Dynamics for Protein Secondary Structure Prediction*, volume 1828. Springer, Jan. 2001.
3. M. Bratman, D. Israel, and M. Pollack. Plans and resource-bounded practical reasoning. *Computational Intelligence*, 4(4):349–355, 1988.
4. R. Brooks. *Cambrian Intelligence: The Early History of the New AI*. MIT Press (A Bradford Book), 1999.
5. G. A. Carpenter and S. Grossberg. A massively parallel architecture for a self-organizing neural pattern recognition machine. *Computer Vision, Graphics, and Image Processing*, 37:54–115, June 1987.
6. G. A. Carpenter, S. Grossberg, N. Markuzon, J. H. Reynolds, and D. B. Rosen. Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analog multidimensional maps. *IEEE Transactions on Neural Networks*, 3:698–713, 1992.
7. D. Gordan and D. Subramanian. A cognitive model of learning to navigate. In *Nineteenth Annual Conference of the Cognitive Science Society*, 1997.
8. F. Heintz and P. Doherty. DyKnow: A Framework for Processing Dynamic Knowledge and Object Structures in Autonomous Systems. In *Intl. Workshop on Monitoring, Security and Rescue Techniques in MAS*, 2004.
9. C. Heinze, S. Goss, and A. Pearce. Plan Recognition in Military Simulation: Incorporating Machine Learning with Intelligent Agents. In *Proceedings, IJCAI Workshop on Team Behaviour and Plan Recognition*, pages 53–63, Stockholm, Sweden, 1999.
10. L. Kaelbling, M. Littman, and A. Moore. Reinforcement learning: A survey. *Journal of Artificial Intelligence Research*, 4:237–285, 1996.
11. S. Karim and C. Heinze. Experiences with the Design and Implementation of an Agent-based Autonomous UAV Controller. In *Proceedings of the Fourth International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, University of Utrecht, The Netherlands, 2005.

12. A. Newell. *Unified Theories of Cognition*. Harvard University Press, Cambridge, Massachusetts, 1990.
13. A. S. Rao and M. P. Georgeff. BDI-agents: from theory to practice. In *Proceedings, First International Conference on Multiagent Systems*, San Francisco, 1995.
14. R. Sun. *Duality of the mind: A bottom-up approach toward cognition*. Lawrence Erlbaum Associates, Inc., Mahwah, NJ, 2002.
15. R. Sun and C. Sessions. Learning plans without a priori knowledge. *Adaptive Behavior*, 8(3/4):225–254, 2000.
16. A. H. Tan. Adaptive Resonance Associative Map. *Neural Networks*, 8(3):437–446, 1995.
17. A. H. Tan. FALCON: A fusion architecture for learning, cognition, and navigation. In *Proceedings, IJCNN, Budapest*, pages 3297–3302, 2004.
18. M. Veloso, J. Carbonell, A. Pérez, D. Borrajo, E. Fink, and J. Blythe. Integrating Planning and Learning: The PRODIGY Architecture. *Journal of Experimental and Theoretical Artificial Intelligence*, 7(1):81–120, 1995.

# A Knowledge-Based Modeling System for Time-Critical Dynamic Decision-Making

Yanping Xiang and Kim-Leng Poh

Department of Industrial and Systems Engineering, National University of Singapore,  
10 Kent Ridge Crescent, Singapore 119260, Singapore  
[yanping\\_xiang@yahoo.com.cn](mailto:yanping_xiang@yahoo.com.cn), [isepohk1@nus.edu.sg](mailto:isepohk1@nus.edu.sg)

**Abstract.** Knowledge-based model construction approach has been applied to many problems. Previous research doesn't provide an approach to construct models to deal with time-critical dynamic decision problems. This paper presents a Knowledge-based Time-critical Dynamic Decision Modeling system (KTDDM) for time-critical dynamic decision-making. The system architecture and functional modules are described. A new knowledge representation framework is provided to support the whole model construction process. This paper applies the KTDDM system prototype to construct a decision model for the time-critical dynamic medical problem on cardiac arrest to demonstrate the effectiveness of this approach.

## 1 Introduction

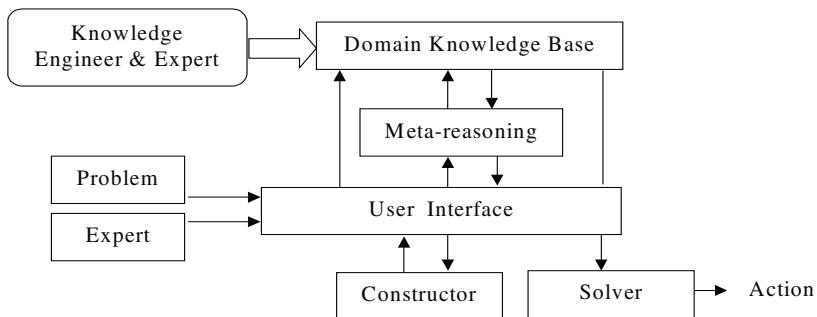
Knowledge-based model construction (KBMC) is the process of constructing a model for a problem instance from a knowledge base representing generic domain entities and their interrelationships. Recently, knowledge-based model construction approach has been applied to problems such as military situation assessment [1], student modeling for intelligent tutoring [2] and so on. However, these researches don't provide an approach to support time-critical dynamic decision modeling. Some research [3] proposes the meta-reasoning method for time-critical sequential decision making. Based on these previous researches, this paper presents a time-critical dynamic decision modeling system (KTDDM), which is designed for time-critical dynamic decision-making. The system constructs a situation-specific time-critical dynamic decision model from a knowledge base according to the resource availability and other specific requirements, reflecting the tradeoffs between decision quality and computational cost.

In developing time-critical dynamic decision-support systems, modeling time and the needs to deal with time-pressured situations are considered to be the greatest challenges. In a related work [4] [5], we proposed a formalism called time-critical dynamic influence diagrams (TDID), which provide explicit support for the modeling of time-critical dynamic decision problems. Four algorithms [4][5] were developed to evaluate the time-critical dynamic influence diagrams. We also proposed the use of meta-reasoning to select an optimal model and an optimal algorithm based on the best tradeoffs between decision quality and computational complexity [4]. Based on our

previous research[4][5], this paper presents the KTDDM system that can deal with time-pressured situations. A new knowledge representation framework is provided to support the whole model construction process. The KTDDM system uses meta-reasoning techniques to control the model and algorithm selection. In meta-level analysis, the best model and algorithm are suggested based on decision quality and computational cost under time-critical situations [4]. By using the best algorithm to evaluate the best model, the KTDDM system can generate an optimal policy that will direct action, and the maximum expected value of the optimal policy.

## 2 Overview of the KTDDM System

Figure 1 shows the architecture of the KTDDM system, with blocks indicating system components, and arrows indicating information inflows. This system consists of six components: domain knowledge base, outside input (i.e. problem requirements, expert judgment), meta-reasoning, constructor, solver and user interface.



**Fig. 1.** The KTDDM system architecture

*Domain knowledge base* stores a large number of models about the domain. These models' attributes are also recorded in the knowledge base. For any given problem, there should be some specific information about the problem and some judgments from experts as well. Based on these *outside inputs*, *Meta-reasoning* chooses the best model and algorithm from the knowledge base based on the consideration of computation cost and decision quality. Once the best model is suggested, the model can be modified on the component *constructor*. If model is changed, the attributes of the new model are computed and new knowledge is added into domain knowledge base. Given the best model and algorithm, the component *solver* shall provide an optimal policy and the value of the optimal policy. These results direct the timely action. *User interface* provides the mechanism for communication between user and the system. Visual C++ is used to implement the KTDDM system prototype. An example will be presented in Section 5 to demonstrate the effectiveness of the KTDDM system.

### 3 Knowledge Representation

In recent years, many knowledge representation schemas have been developed for the dynamic construction of probabilistic and decision models. These include first-order logic [6] [7] [8], probabilistic concept networks [9], multilevel influence diagrams [10], sequential influence diagrams [11] and so on. In this paper, a new knowledge representation framework is presented, which is able to support time-critical dynamic modeling. In the proposed knowledge representation framework, time-critical dynamic influence diagrams are used as the basis of the knowledge representation.

#### 3.1 Time-Critical Dynamic Influence Diagrams

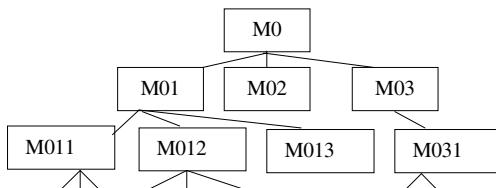
Our previous research [4] [5] has discussed TDID. This subsection reviews TDID briefly. TDID is designed to facilitate the modeling and solution of time-critical dynamic decision problems. It extends standard influence diagram [12] by including the concepts of temporal arcs and time sequences. It also incorporates dynamic influence diagrams [13] as a representation for inference purposes.

A TDID model has two forms: the *condensed form* and the *deployed form*. The condensed form is mainly used to represent or define the dynamic decision model and is the form used in the modeling process. It allows for the coexistence of nodes of different temporal details in the same model. The deployed form is used only for inference purposes and is constructed from the condensed form. Although in principle both forms can be converted to and from each other, they serve different purposes in the modeling and solution processes.

TDID supports model abstraction. It provides a method to represent models at different levels of space abstraction and temporal abstraction.

#### 3.2 Knowledge Base

For each domain, a tree corresponding to this domain is constructed. Figure 2 shows an example.



**Fig. 2.** An example of a domain knowledge base

Each node in the tree corresponds to a TDID model. The root node (e.g. M0) corresponds to the full information model, which includes all information that the domain expert provides, while the others correspond to the abstraction models. In other words, the full information model is the most precise model in the domain. These models, which correspond to child nodes (e.g. M01, M02, and M03), are the direct

abstractions of the model corresponding to their parent (e.g. M0). The hierarchical tree structure facilitates the knowledge acquisition, representation and retrieval.

For each model, there are two basic categories of information: key information and structure information. Key information represents the model attributes while structure information provides the structure information for the model. The two categories of information are described as follows:

Key information is used for retrieving the knowledge base. The key information includes Model ID, a set of algorithms, decision quality and computation time. *Model ID* is an identifier for the model. Each model has one unique ID. A *set of algorithms* are possible algorithms that are used to evaluate model. In this study, the set of algorithms include TDID value iteration algorithm (VI), Model  $\epsilon$ -Reduction algorithm (*M $\epsilon$ R*) and heuristic algorithm (MLS), which have been discussed in our previous research [4]. *Decision quality* [4] measures how good the decision is. It is estimated by model utility, which is further determined by the model structure, initial state and solution algorithm. In this study, the same initial state is used for all models in a tree to estimate decision quality. The decision quality of the combination of full information model and exact algorithm (VI) is considered as the standard, while the decision quality of the others combinations are expressed as fraction of the standard. *Computation time*, which is a function of structural complexity, algorithm and the speed of computer, is the length of time which takes for the system to make optimal policy. Computational cost is estimated by computational time.

Table 1 shows the key information table for a model. In the current work, three algorithms are provided, and hence there are three items in each key information table. Using different algorithms to evaluate the model, the decision quality and computation time are different. In other words, for each model, there are three combinations of model and algorithms.

**Table 1.** Key information table

Model ID	Algorithms	Decision Quality	Computation Time
M012	VI	Q1	t1
	MeR	Q2	t2
	MLS	Q3	t3

In this domain knowledge base, TDID is applied to represent model. For the same domain, models with different abstract levels are developed. Given a specific problem, an appropriate model for this problem should be chosen. If all models are stored in the domain knowledge base, a large amount of resources will be consumed. Figure 2 shows the relationships among these models: the children models are abstractions of their parent. The relationships allow the TDID model to be represented as the combination of structure information. Structure information includes full information model, fragment-list and temporal information.

The full information model which includes all information about the domain corresponds to the root node in a domain knowledge base tree. It is the TDID condensed form. The other models in the domain knowledge base are abstractions of the full information model. Thus, there is only one full information model in a domain

knowledge base. Fragment-list is a series of network fragments that correspond to operations which modify the full information model. There are two types of operations: chance variable abstraction and node deletion. The two types of operations implement space abstraction. Temporal Information is a set of time sequences. Each time sequence corresponds to a temporal variable in TDID condensed form. According to temporal information, the time pattern can be determined. Clearly, temporal information implements temporal abstraction. Our previous research [4] addressed model space abstraction and temporal abstraction in details.

In a domain knowledge base, each model is constructed by combining three kinds of information in structure information. For each TDID model, full information model must exist, while fragment-list and temporal information can be null. If Fragment-list is null, the model has no space abstraction. If temporal information is null, the model has no temporal abstraction. If both are null, the model is a full information model. To construct a TDID model, full information model is combined with fragment-list and temporal information. Then, data are distributed according to their corresponding abstraction.

## 4 Knowledge-Based Model Construction

In the KTDDM system, knowledge base is used to support the decision model construction process. For a specific domain, knowledge base stores the different abstraction models for the domain. The KTDDM system first selects the best model and its corresponding algorithm from the domain knowledge base under time-critical situations. Then, the selected algorithm is employed to evaluate the selected model. The following will describe the model construction process.

1. Given a time-critical dynamic decision problem, specify the problem requirements such as its urgency and deadline. If the deadline is uncertain, the anytime algorithm [4] can be used to solve the problem directly. The anytime algorithm can be interrupted at any point to supply an answer (a current best policy) whose quality increases with increasing computation time. If the cost of delay can be described by certain deadline, the next step is to select the best model and algorithm.
2. Meta-reasoning chooses the best model and algorithm from knowledge base. Our previous research [4] presented the meta-reasoning algorithm supported by a domain knowledge base for determining the best model and algorithm based on the consideration of computation cost and decision quality.
3. User can modify or customize the selected best model according to their requirements if necessary. If the model is modified, the new model will be added into the domain knowledge base.
4. The condensed form of the selected model is converted into the deployed form, some specific information will be entered according to the problem and the expert judgment, and then the selected algorithm is used to evaluate the selected model.

For a specific problem, the proposed model construction approach selects a situation-specific time-critical dynamic decision model and its corresponding algorithm from a knowledge base based on the domain knowledge base and external inputs. When the external inputs change, the selected decision model and algorithm might be

changed. Therefore, the selected model and algorithm may be constantly changed according to the situation.

There are many factors affecting model and algorithm selection. The deadline is one of the most important user's parameters. For uncertain deadline, one initial model is chosen and then it is refined using the anytime algorithm. The model changes over time. For fixed deadline, meta-analysis is used to select the optimal model and algorithm. Model and algorithm selection depends on tradeoff between decision quality and computation cost. Both are determined by some pre-computed attributes, which are recorded in the key information of knowledge base. The decision quality depends on models, algorithms, and time, while the computation cost is determined by estimated computational time.

## 5 Example

A medical example from the domain of cardiac arrest [14] is used to illustrate the knowledge representation and model construction process, and demonstrate the effectiveness of the KTDDM system. In the cardiac arrest problem, the goal of the medical treatment is to maintain life and prevent anoxic injury to the brain. The observable variable is the electrocardiogram or rhythm strip (cr). While the survival of patient is of primary importance, cerebral damage must be taken into account and can be viewed as part of the cost in a resuscitation attempt. The length of time that patient has been without cerebral blood flow (cbf) determines the period of anoxia (poa). If the patient has ineffective circulation for more than five minutes, there is a likelihood of sustaining cerebral damage (cd). This damage is persistent and its severity increases as the period of anoxia increases. Medical doctors treat a patient experiencing a cardiac arrest with a variety of interventions and medications.

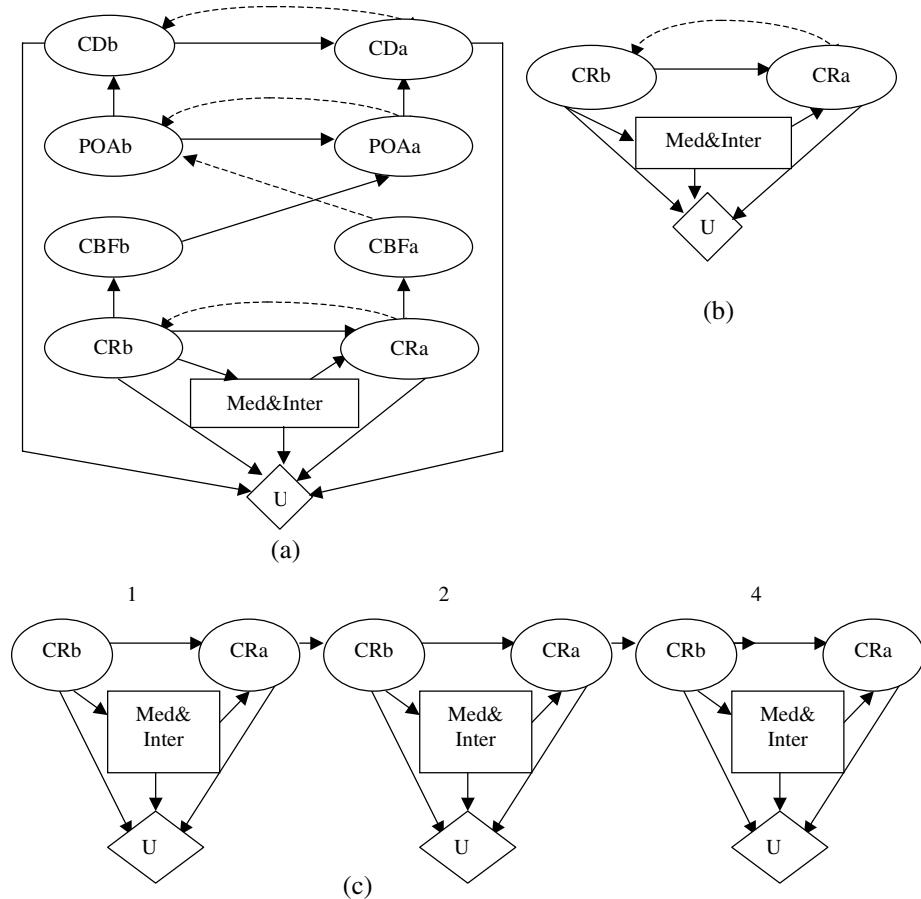
In the domain knowledge base for cardiac arrest, a tree is used to represent the domain of cardiac arrest. Each node in the tree corresponds to a TDID model.

For each model, its key information, which represents the model attributes, is recorded. Three algorithms (VI, M<sub>e</sub>R and MLS) are used to evaluate the model. For each algorithm, the value of optimal policy is used to estimate the decision quality, and the computation time is also recorded. Briefly, after a model is constructed, the model is evaluated by available algorithms and its attributes are entered into key information table.

For each model, its structure information provides the information for TDID model construction. An example of the structure information of a model is described as follows:

The full information model is shown in Figure 3 (a). It is the TDID condensed form with default time sequence (i.e. all time sequences are <0,1,2,3,4,5,6,7,8,9>) and time unit (minute). In the TDID condensed form, time is modeled discretely and the temporal pattern can be dynamically modified according to temporal sequence. Solid arcs represent instantaneous probabilistic relations. In a time slice, heart rhythm (CR) after the action is influenced by the rhythm before action, as well as medication and intervention. Cerebral blood flow (CBF) is determined by the heart rhythm. The period of anoxia (POA) after action is dependent on the period of anoxia and the cerebral blood flow (CBF) before action. The extent of cerebral damage (CD) after action

is determined by the cerebral damage before action and the period of anoxia after action. The broken arcs indicate time-lagged probabilistic relations. Heart rhythm after action influences the rhythm before action in the next time slice. In the full information model, all time sequences are  $<1, 2, 3, 4, 5, 6, 7, 8, 9>$ . Thus, nine temporal intervals are modeled. Heart rhythm after action in time slice  $n$  ( $1 \leq n \leq 8$ ) influences the rhythm before action in time slice  $n+1$ . The same explanation is valid for other broken arcs. All models in the domain share the full information model.



**Fig. 3.** TDID models for cardiac arrest domain

The fragment-list that represents space abstraction context is a list of fragments which modify the full information model by combining the full information model with the fragments. In the fragment list, there are two fragments: Deletion node (CDa) and Deletion node (CDb). Each fragment corresponds to operations which modify the full information model. Using the two deletion node operations, the nodes (CDa and

CD<sub>b</sub>) and all subsequent barren nodes are eliminated. The model shown in Figure 3 (b) is obtained by combining the full information model with the two fragments (i.e. Deletion node (CD<sub>a</sub>) and Deletion node (CD<sub>b</sub>)), and the combination implements space abstraction of the full information model.

In the temporal information, there is a set of time sequence (i.e.  $T_{CRb} <1, 2, 4>; T_{CRA} <1, 2, 4>; T_{med\&inter} <1, 2, 4>; T_U <1, 2, 4>$ ) which controls the modeling of time. Figure 3 (c) shows the deployed form after combining the temporal information with the model in Figure 3 (b), and the combination implements the temporal abstraction. In the deployed form, three temporal intervals are modeled.

In this example, the model in Figure 3 (c) is constructed by the full information model (Figure 3(a)), the fragment-list (i.e. Deletion node (CD<sub>a</sub>) and Deletion node (CD<sub>b</sub>)) and a set of time sequences (i.e. all time sequences are  $<1, 2, 4>$ ).

Given the domain knowledge base for cardiac arrest, the following will use a cardiac arrest case to demonstrate how the KTDDM system constructs a time-critical dynamic decision model for the medical problem on cardiac arrest.

The cardiac arrest case is described as follows: The patient experiences a massive myocardial infarction. Medical doctor decides the delay cost for the patient according to his conditions. The delay cost must be determined a-priori for different classes of patients since there is no time to do this once an individual patient is in cardiac arrest. The delay cost is represented by the inference related cost shown in Figure 4. Both CPR and LIDO are available actions.

In this case, the inference related cost function is an increasing function with delay time  $t$ . In this situation, meta-reasoning is used to select best model and algorithm and then evaluate the selected model with the selected algorithm.

The cost of meta-reasoning is negligible because the meta-reasoning time is very short. Figure 4 shows the result of model and algorithm selection. In the process of model and algorithm selection, the KTDDM system first generates a precise function of object-level utility. The object-level utility is the value associated with the information represented by the computational result of model and algorithm without regard to the cost of reasoning which may be necessary to generate this result. In Figure 4, each point in the object-level utility curve represents a specific model and algorithm, and its position shows its runtime and utility. The object-level utility function is such that the object-level utility increases with runtime. The inference-related cost is provided according to the conditions of the patient, which is represented by an increasing function. The comprehensive utility curve is the difference between the object-level utility and inference-related cost at each time point. It is observed that the greatest expected value of computation EVC[15], which is defined as the net change in the comprehensive utility in return for an allocation of some computational resource to reasoning, is reached at the time at which the comprehensive utility is maximized for this particular problem and the given cost function. The system uses the EVC to determine the best selection[4]. In Figure 4, the point in the circle corresponds to the best model and the optimal algorithm (i.e. MeR).

If user thinks that the selected model is good enough to solve the problem and doesn't need to be modified, the patient's initial state (i.e. the rhythm is ventricular fibrillation, the period of anoxia is 2 min and cerebral damage is none) will be entered. The system uses the selected algorithm (MeR) to evaluate the selected model. The solution produced by the MeR algorithm is an optimal policy indicating the

optimal expected value achievable in all possible states for all possible decision stages. The patient is initially in the state of ventricular fibrillation, with two minutes of anoxia and no cerebral damage. The solution is that the best treatment strategy is CPR and LIDO at the beginning of treatment. The following strategies follow the optimal policy. For example, if the patient is in the state of normal sinus rhythm (nsr) the five minutes later, the best strategy is none which means "stop CPR and LIDO".

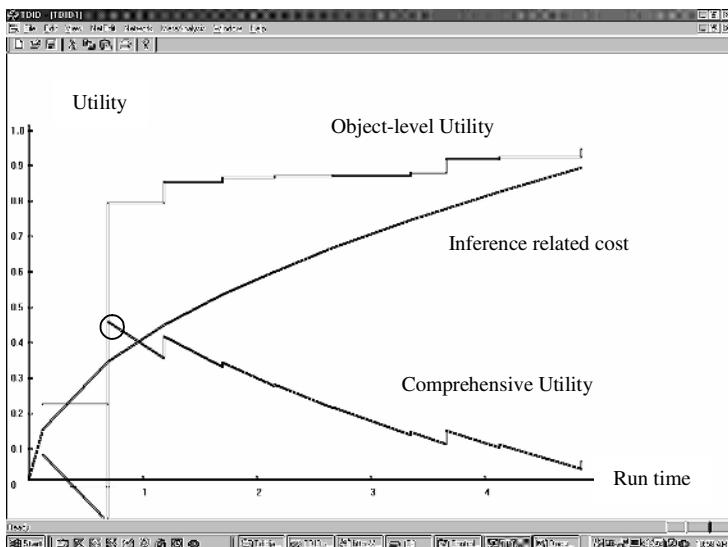


Fig. 4. Model and algorithm selection

In this experiment, it is noted that the object-level utility and the comprehensive utility are retrospective; decision maker faces a resource-bounded problem. It is also noted that the final solutions might depend on external inputs such as the computational environment, the initial state and the deadline. The results described here are based on particular external inputs. When the external inputs change, EVC might be changed. The change will affect the selected model and algorithm, and in turn lead to changes in the final solutions.

## 6 Conclusion

Previous research doesn't provide an approach to deal with time-critical dynamic decision problems. This paper presents the KTDDM system for time-critical dynamic decision making.

In developing the KTDDM systems, modeling time and the needs to deal with time-pressured situations are considered to be the greatest challenges. Based on our previous research, this work proposes a new knowledge representation framework that provides support for time-critical dynamic decision problems modeling and facilitates knowledge acquisition, representation and retrieval. To deal with time-pressured

situation, this study presents a knowledge based model construction process supported by the proposed knowledge representation framework. Previous research in KBMC has focused on bottom-up or top-down model construction. In this study, all models are stored in the domain knowledge base and meta-reasoning techniques are used to select the best model and algorithm based on the best tradeoffs between decision quality and computational complexity. To evaluate the feasibility and effectiveness of the KTDDM system, an example is used to test the prototype system. The prototype system produces adequate solutions for the example.

## References

1. Mahoney, S.M., Laskey, K.B.: Constructing Situation Specific Belief Networks. Proc. of Fourteenth Conference on Uncertainty in Artificial Intelligence (1998) 370-378
2. Gertner, A.S., Conati, C., VanLehn, K.: Procedural Help in Andes: Generating Hints Using a Bayesian Network Student Model. Proc. of American Association for Artificial Intelligence (1998) 106-111
3. Dean, T., Kaelbling, L.P., Kirman, J., Nicholson, A.: Deliberation Scheduling for Time-critical Sequential Decision Making. Proc. of ninth Conference on Uncertainty in Artificial Intelligence(1993) 309--316
4. Xiang, Y.P., Poh, K.L.: Knowledge-based Time-Critical Dynamic Decision Modeling. J. of the Operational Research Society, Vol. 53 (2002) 79-87
5. Xiang, Y.P., Poh, K.L.: Time-Critical Dynamic Decision Modeling in Medicine. Computers in Biology and Medicine, Vol. 32 (2002) 85-97
6. Breese, J.S.: Construction of Belief and Decision Networks. Computational Intelligence, Vol. 4 (1992) 624-647
7. Goldman, R.P., Charniak, E.: Dynamic Construction of Belief Networks. Proc. of sixth Conference on Uncertainty in Artificial Intelligence (1990) 90-97
8. Holtzman, S.: Intelligent Decision Systems. Addison-Wesley (1989)
9. Poh, K.L., Fehling, M.R., Horvitz, E.J.: Dynamic Construction and Refinement of Utility-Based Categorization Models. IEEE Transaction on Systems, Man, and Cybernetics, Vol. 24 (1994) 1653-1663
10. Poh, K.L., Wu, X.: Computer-aided Decision Modeling using Multilevel Influence Diagrams. Proc. of third International Conference on Systems Science and Systems Engineering (1998) 49-52
11. Jensen, F.V., Nielsen, T.D., Shenoy, P.P.: Sequential influence diagrams: A unified asymmetry framework, Proc. of the Second Workshop on Probabilistic Graphical Models (2004) 121-128
12. Howard, R.A., Matheson, J.E.: Influence Diagrams. In: Howard, R.A., Matheson, J.E. (eds.): Readings on the Principles and Applications of Decision Analysis. Strategic Decisions Group, Menlo Park, CA (1984) 719-726
13. Tatman, J.A., Shachter, R.D.: Dynamic Programming and Influence Diagrams. IEEE Transactions on Systems, Man and Cybernetics, Vol. 20 (1990) 365-379
14. Ngo, L., Haddaway, P., Helwig, J.: A Theoretical Framework for Context-Sensitive Temporal Probability Model Construction with Application to Plan Projection. Proc. of eleventh Conference on Uncertainty in Artificial Intelligence (1995) 419-426
15. Horvitz, E.J.: Computation and Action under Bounded Resources, PhD thesis, Stanford University (1990)

# Mining Frequent Itemsets for Protein Kinase Regulation

Qingfeng Chen<sup>1</sup>, Yi-Ping Phoebe Chen<sup>1</sup>, Chengqi Zhang<sup>2</sup>, and Lianggang Li<sup>3</sup>

<sup>1</sup> Faculty of Science and Technology

Deakin University

VIC 3125, Australia

{qifengch, phoebe}@deakin.edu.au

<sup>2</sup> Faculty of Information Technology

University of Technology Sydney

PO Box 123, Broadway NSW 2007, Australia

chengqi@it.uts.edu.au

<sup>3</sup> Biomedical Engineering Research Center,

Sichuan University, 61004, China

llianggang@sohu.com

**Abstract.** Protein kinases, a family of enzymes, have been viewed as an important signaling intermediary by living organisms for regulating critical biological processes such as memory, hormone response and cell growth. The unbalanced kinases are known to cause cancer and other diseases. With the increasing efforts to collect, store and disseminate information about the entire kinase family, it not only leads to valuable data set to understand cell regulation but also poses a big challenge to extract valuable knowledge about metabolic pathway from the data. Data mining techniques that have been widely used to find frequent patterns in large datasets can be extended and adapted to kinase data as well. This paper proposes a framework for mining frequent itemsets from the collected kinase dataset. An experiment using AMPK regulation data demonstrates that our approaches are useful and efficient in analyzing kinase regulation data.

## 1 Introduction

In recent years, there have been considerable efforts made to investigate DNA, RNA and proteins, and available biological information is constantly updated at a spectacular rate due to new and efficient experimental techniques. The completion of the sequencing of the human genome, and those of other organisms, is expected to lead to an attractive shift from genome mapping and sequencing to the determination of genome function, called functional genomics [1]. Protein is an important molecule that accomplishes most of a living cell's functions [2]. Thus, the analysis of protein kinases regulation in living organisms can play an important role in functional genomics. However, no systematic methods are available to analyze this data and further understand cellular regulation.

One goal in terms of analyzing protein kinases regulation data is to determine how external stimuli might affect the catalytic subunit and regulatory subunit of protein kinases. Each subunit consists of several genes and encoding isoforms. Another goal

is to determine what isoforms are expressed or unchanged in expression as a result of certain conditions. For example, AMP-activated protein kinase (AMPK) has recently emerged as a potential key signaling intermediary in the regulation of exercise-induced changes in glucose and lipid metabolism in skeletal muscle [3, 4, 5]; Mos/Raf kinases form part of the MAPKK Kinase family, are activated by growth factors and perform important enzyme functions to stimulate the growth of cells [6]. To achieve these goals, we need to find itemsets with a high frequency from raw data. This will contribute to further identifying frequent patterns and increase the understanding of protein kinases and relevant disease treatment.

Earlier investigations predominantly focus on discovering an association between the gene expression, gene sequence prediction and protein structure prediction. Protein regulation, with respect to cellular metabolism, has recently started to draw the attention of biologists. However, the development of methods for interpreting the protein kinases data lags behind the speed at which the data is accumulated. Also, the intricacies caused by the interactions among subunit isoforms and between subunit isoforms and external stimuli, increase the complexity of this data. Existing algorithms and approaches for the analysis of sequences or structures cannot be directly transferred to handle kinase data owing to status messages of subunit isoforms and stimulus factors. Thus, it demands analyst skills for the adapting of traditional approaches to interpret biological data of interest.

Recently, data mining techniques have emerged as a means of identifying patterns and trends from large quantities of data [7, 8]. Several attempts have been made to mine biological databases using association rule mining [9, 10]. Although many experiment results about kinase regulation have been published in literary documents, insufficient efforts have been made to collect the data and systematically analyze it.

In traditional market basket data, an itemset may be of the form  $\{X\}$  where  $X$  represents one or a collection of items. A popular example is ‘milk’ and ‘nappy’ in the context of a supermarket transaction. As for protein kinase regulation, the item may represent the subunit isoform of kinases that is highly expressed or unchanged in expression, and stimulus factors that indicate certain stimulation such as the *intensity* and *time length*. An example of items extracted from AMPK regulation data might be  $\{\text{moderate intensity}\}$  and  $\{\alpha_{1a} |, \alpha_{2a} \uparrow\}$  where  $\alpha_{1a}$  and  $\alpha_{2a}$  represent activation of isoforms of  $\alpha$  subunit of AMPK. The symbols  $|$  and  $\uparrow$  represent *no change* and *highly expressed* in terms of expression, respectively.

This paper presents a framework by which to analyze the protein kinase regulation data collected from experimental results taken from various literary publications. In particular, the missing values for particular subunit isoforms were intentionally untested in experiments. Furthermore, the items that are not tested by adequate experiments will be reserved in databases for further use. *FPTree* (Frequent Pattern Tree) algorithm [8] is used to find frequent itemsets. The conducted experiments using AMPK regulation data discover many useful frequent itemsets from data. This can be a good foundation by which to identify potential correlations between states of AMPK subunit isoforms or between stimulus factors and states of isoforms.

This paper is organised as follows. Section 2 highlights the definition of the problem under consideration. In section 3, the procedure to find frequent itemsets from kinase regulation data is introduced. Section 4 provides an analysis into the discovered frequent itemsets. Finally, this paper is concluded in section 5.

## 2 Definition of Problem

Protein kinases play a central role in regulating important biological processes in living organisms. Recent studies have indicated that kinases may cause various diseases when they become out of balance. For example, AMP-activated protein kinases participate in the regulation of exercise-induced changes. Changes which occur in a skeletal muscles's glucose and lipid metabolism, gene expression and protein synthesis. Usually, protein kinases perform their functions by congregating the subunit isoforms. Therefore, interpretation of kinase regulation data and identifying potential associations from these data are very important processes. Such processes will lead to an understanding of its structure, function and expression. However, only limited studies [11] have been devoted to analyzing such valuable data. In this paper, a framework is proposed that identifies frequent itemsets from kinase regulation data.

Given a set of experimental results,  $E$ , each result is described by a set of items. The items can be classified into two categories:

1. *Stimulus item (ST)* represents a parameter that can be used to measure stimuli. For example, intensity, load and duration are generally used to measure the exercise stimuli. ++, + and – indicates that the stimulus is “high”, “moderate” and “low”, respectively.
2. *State item (S)* represents isoforms of kinases such as  $\alpha$ ,  $\beta$  and  $\gamma$  subunits of AMPK.  $\uparrow$ ,  $|$  and  $\downarrow$  are used to indicate “highly expressed”, “no change in expression” and “highly repressed”, respectively.

The items collected from different experiments contain a great deal of hidden information that may in fact be meaningful. For example, in an observed experiment based on the training of a moderate intensity treadmill,  $\beta_2$  and  $\gamma_2$  isoforms of AMPK are found to be highly expressed in terms of activation in white quadriceps [12]. The initial itemset for this experiment can be written as  $A = \{\text{moderate intensity treadmill}\}$ ,  $B = \{\beta_{2a}\uparrow, \gamma_{2a}\uparrow\}$ . This is hypothetically derived from a single experiment. However, in practice, the frequent itemsets are mined from a collection of experimental results. On the other hand, in conventional market basket data, even if an item does not occur in a transaction, it is counted in the calculation of frequency. However, the experiment that does not contain an item will not be counted. In other words, if an item is not tested in an experiment, it will not be considered in the calculation of its frequency. Consequently, the raw data needs to be cleaned up before mining frequent itemsets.

When several experimental results are produced under similar conditions, frequent itemsets can be generated in terms of a given *minsupp*. A prime example situation is if  $\{\text{high intensity training}\}$ ,  $\{\beta_{2a}\uparrow\}$  and  $\{\text{high intensity training}, \beta_{2a}\uparrow\}$  are all frequent itemsets derived from the set of experimental data,  $E$ . It is possible that an association exists between itemsets  $\{\text{high intensity training}\}$  and  $\{\beta_{2a}\uparrow\}$  in terms of the best known search strategy of frequent itemsets called Apriori [13]. For the sake of simplicity, the computational details of frequency and confidence are not presented at this stage.

### 3 Mining Frequent Itemsets Protein Kinases Regulation

#### 3.1 Basic Definitions

In this work, the conventional data mining approaches are extended to the analysis of protein kinase regulation data. Suppose  $E = \{E_1, \dots, E_n\}$  is a set of experiments. Each experiment consists of an *eid* (experiment identifier) and two itemsets,  $E_i = (\text{eid}, S_i, ST_i)$ ;  $ST_i \Rightarrow S_i$  is treated as an initial rule. Let  $I = \{x \mid x \in S_i \cup ST_i, 1 \leq i \leq n\}$  be a set of items, and  $A \subseteq I$  be itemsets. An itemset  $A$  has *support*,  $s$ , in the set of experiments if  $s\%$  of experiments contains  $A$ . The minimum support (*minsupp*) is a threshold given by users or experts. Itemset  $A$  is a frequent itemset if, and only if, its support is equal to, or greater than, the specified minimum support.

#### 3.2 Deriving Initial Items

In general, mining of frequent patterns starts from the generation all frequent itemsets. The mined dataset can be derived from different experiment results that are under different experimental conditions. The external conditions have to be considered to in order to determine the correct itemsets. To locate frequent itemsets, initial items from experimental data derived under similar experimental conditions, such as *human skeletal muscle* and *endurance training*, must be determined. Also, the item derived from kinase data includes its *status measurement* and *item name* such as  $\alpha_{1a} \downarrow$  and  $\beta_{2a} \uparrow$ .

In protein kinase regulation data, *activity*, *protein expression* and *phosphorylation* are used as testing indexes for  $\alpha$ , whereas only *protein expression* is used as a testing index for  $\beta$  and  $\gamma$ . The following steps can be used to generate initial items from experimental data:

- Generate initial rules  $ST \Rightarrow S$  from experimental data where  $S$  and  $ST$  represent state itemset and stimulus itemset, respectively, and;
- Derive a set of initial items  $I = S \cup ST$ .

It is illustrated in the form of  $\{x_1, x_2, \dots, x_n\}$  where  $x_i \in I$ . Let experimental universe be  $EID = \{E_1, E_2, E_3, E_4\}$ . Table 1 represents an example of initial items derived from different experiments. Each row in Table 1 corresponds to an experiment for AMPK regulation. There are two initial rules, namely, 1)  $\{\phi^+, \varphi^-\} \Rightarrow \{\alpha_{1a} \downarrow, \alpha_{1p} \uparrow\}$  and 2)  $\{\phi^+, \varphi^-\} \Rightarrow \{\alpha_{1a} \downarrow, \alpha_{1e} \downarrow, \alpha_{1p} \downarrow\}$ , where subscripts  $a$ ,  $p$  and  $e$  represent *activity*, *phosphorylation* and *protein expression*, respectively. Therefore,  $\forall x \in \forall x \in \{\alpha_{1a} \downarrow, \alpha_{1p} \uparrow, \alpha_{1e} \downarrow, \alpha_{1p} \downarrow, \phi^+, \varphi^-\}$  can be viewed as an initial item.

**Table 1.** An Example of Experimental Database

EID	Items					
	$\alpha_1$		$\beta$	$\phi$	$\varphi$	
$E_1$	$\alpha_{1a} \downarrow$	—	$\alpha_{1p} \uparrow$	—	—	$\phi^+ \varphi^-$
$E_2$	$\alpha_{1a} \downarrow$	$\alpha_{1e} \downarrow$	$\alpha_{1p} \downarrow$	$\beta_{1e} \downarrow$	$\beta_{2e} \downarrow$	$\phi^+ \varphi^-$

### 3.3 Identifying Frequent Itemsets

The typical data format when using *FP-tree* algorithm is shown in Table 1. From observation, not every item occurs in all experiments. The experiment that does not contain a specific itemset is ignored when computing its frequency. Furthermore, if an itemset is not tested in most of the experiments, it will not be used immediately when finding frequent itemsets, however, it will be stored for future use.

Suppose  $occur(X)$  is a ratio of experiments in  $E$  containing itemset  $X \in S$  and represents the occurrence of  $X$  in  $E$ , it can be written:

$$occur(X) = |X(E_i)| / |E| \quad (1)$$

where  $X(E_i) = \{E_i \in E \mid E_i \text{ contains } X\}$ . For example,  $occur(\alpha_{le})$  in Table 1 is 1/2 or 50%. It is to be noted that the status measurement of  $x \in X$  is not considered in this paper and the following formula (2).

$X$  is a *high occurrence 1-itemset* if  $occur(X) \geq \text{minimum occurrence (minoccur)}$ , as provided by the training set. Otherwise, it is called a *low occurrence 1-itemset*. Nevertheless, we can get different  $X(E_i)$  for different  $X$  that may result in inconsistent example spaces during the computation of their frequency. In Table 1, for example,  $\alpha_{le}(E) = \{E_2\} \neq \alpha_{la}(E) = \{E_1, E_2\}$  because  $\alpha_{le}$  is not tested in  $E_1$ . To find correct frequent itemsets, the maximal subset  $E'$  is applied to ensure that the included experiments have common tested items. Therefore, the association rule will be searched in  $E'$  instead of  $E$ . Suppose  $X$  represents a *high occurrence 1-itemset* of  $E$  and  $X \neq \emptyset$ .

$$E' = \bigcap_{|X|=1} X(E_i) \quad (2)$$

Formula (2) reduces experimental datasets and ensures that the correct frequent itemsets are identified from data. For example, in Table 1, let,  $\text{minoccur} = 0.6$ . Thus,  $\alpha_{la}$  and  $\alpha_{lp}$  are high occurrence 1-itemsets. We obtain  $E' = \{E_1, E_2\}$ . To obtain the commonly tested items in  $E'$ , we need the interaction of  $E'$ .

$$IE' = \bigcap_{E_i \in E'} E_i \quad (3)$$

According to formula (3), we have  $IE' = \{\alpha_{la}, \alpha_{lp}\}$ . The item derived from kinase regulation data comprises its *status measurement* as well as *item name* such as  $\alpha_{la} \uparrow$ . Table 2 presents the frequent 1-itemsets of Table 1 with the status measurement of items. Let  $\text{minsupp} = 0.3$  and  $\text{minoccur} = 0.6$ .  $\alpha_{le}$ ,  $\beta_{le}$  and  $\beta_{2e}$  are pruned due to their low occurrences.

**Table 2.** Frequent 1-itemsets of Table 1 with status measurement

Itemset	Frequency	Frequency > minsupp
$\alpha_{la}$	1	true
$\alpha_{lp}$	1	true
$\phi^+$	1	true
$\phi^-$	1	true

## 4 Experiments

### 4.1 Experiment Data of AMPK Regulation

In this research paper, the experimental data is obtained by searching the NCBI database. It is related to endurance training with respect to human skeletal muscle and within which AMPK is activated. Two stimulus indexes, namely, intensity,  $I$  and time length,  $t$ , are considered. There are 6 attributes, 20 items and 24 experiments in the data. AMPK consists of state items: *catalytic subunit* ( $\alpha$ ) and *regulatory subunit* ( $\beta$ ,  $\gamma$ ). In this paper, we focus on *activity* and *protein expression* of catalytic subunit  $\alpha$ .

Presented in Table 3, is an example with respect to the activity and expression of AMPK in skeletal muscle. The entire data range is available at <http://www.deakin.edu.au/~qifengch/pricai06/pricai06.zip>.  $t^-$  represents the duration under 30 minutes,  $t$  represents the duration between 30 and 90 minutes,  $t^+$  represents the duration between 90 and 180 minutes and  $t^{++}$  represents the duration above 180 minutes,  $I^{++}$  represents the *maximal oxygen uptake* (VO(2)max) above 80%,  $I^+$  represents VO(2)max between 60% and 70%,  $I$  represents VO(2)max between 50% and 60% and  $I^-$  represents VO(2)max under 50%. In particular, if an item is not tested in an experiment, it will be marked with ‘-’.

**Table 3.** Activity and expression of AMPK in skeletal muscle

EID	Item					
	$\alpha_{1a}$	$\alpha_{1e}$	$\alpha_{2a}$	$\alpha_{2e}$	Intensity	Time
$E_1$	$\alpha_{1a} \downarrow$	-	$\alpha_{2a} \uparrow$	-	$I^+$	$t$
$E_2$	$\alpha_{1a} \downarrow$	-	$\alpha_{2a} \downarrow$	-	$I$	$t^+$
$E_3$	$\alpha_{1a} \downarrow$	-	$\alpha_{2a} \uparrow$	-	$I^+$	$t^-$
$E_4$	$\alpha_{1a} \downarrow$	-	$\alpha_{2a} \downarrow$	-	$I$	$t$

In experiment  $E_1$  and  $E_2$  (Jorgen *et al.*, 2000), Isoform-specific and exercise intensity-dependent activation of 5-AMP-activated protein kinase in human skeletal muscle was studied; In experiment  $E_3$  (Mei *et al.*, 2003), metabolic and mitogenic signal transduction in human skeletal muscle after intense cycling exercise was investigated;  $E_4$  (Mattew *et al.*, 2000) presents the beta-adrenergic stimulation of skeletal muscle which suggests that HSL can be overridden by AMPK signaling.

It aims at generating all frequent itemsets for further identification of frequent patterns which seeks to make sense biologically. Unlike traditional methods, the process starts with pruning items of low occurrence, rather than directly generating *frequent itemsets*. The initial items from experiments are divided into state items and stimulus items. In the experimental data, the rows and columns represent corresponding experiments and specific items, respectively. The process is repeated until all frequent itemsets are extracted.

### 4.2 Generating Frequent Itemsets

To generate frequent itemsets, the following parameters are used, including  $minsupp = 0.3$  and  $minoccur = 0.25$ ; these empirical values are based on the values specified by biological domain experts.

In the AMPK regulation data set, the untested subunit isoforms of AMPK in experiments do not impact on computing the support of corresponding items. It is to be noted that, in Table 3,  $\alpha_{1e}$  and  $\alpha_{2e}$  were not tested in any experiments. As discussed in Section 3.2,  $occur(\alpha_{1e}) = occur(\alpha_{2e}) = 0 < minoccur$ . Therefore, they are pruned and will not be considered in the finding of frequent itemsets. However, if they are tested in some experiments but their occurrences are smaller than  $minoccur$ , they are still useful and stored in the database. When additional testing results are available in the future, we can eventually find a more meaningful correlation between them.

Based on the formulae (1), (2) and (3) in section 3.3, we have  $E' = \{E_5, E_6, E_{12}, E_{13}, E_{20}\}$ ,  $IE' = \{\alpha_{1a}, \alpha_{1e}, \alpha_{2a}, \alpha_{2e}\}$ . Therefore, the mining will focus on discovering frequent itemsets among  $\alpha_{1a}$ ,  $\alpha_{1e}$ ,  $\alpha_{2a}$ ,  $\alpha_{2e}$ ,  $I$  and  $t$ . We use an adapted *FP-tree* algorithm published in <http://www.cs.concordia.ca/db/dbdm/dm.html> to identify frequent itemsets. Thus, before conducting experiments, we need to transfer the items in the dataset into non-negative integer. Table 4 shows the conversion of items in Table 3.

**Table 4.** Conversion of items to non-negative integer

EID	Item					
	$\alpha_{1a}$	$\alpha_{1e}$	$\alpha_{2a}$	$\alpha_{2e}$	Intensity	Time
$E_1$	1	8	14	20	92	101
$E_2$	1	8	13	20	91	102
$E_3$	1	8	15	20	92	100
$E_4$	1	8	13	20	91	101

**Table 5.** Reconversion of frequent itemsets

Frequent Itemset	Reconverted Frequent Itemset
15 92	$\alpha_{2a} \uparrow, I^+$
18 6 91	$\alpha_{1e} \downarrow, \alpha_{2e} \downarrow, I$
18 14 6	$\alpha_{1e} \downarrow, \alpha_{2e} \downarrow, \alpha_{2a} \downarrow$
91 1 6	$\alpha_{1a} \downarrow, \alpha_{1e} \downarrow, I$

Unlike traditional data, itemsets supported by two or more experiments can be viewed as frequent itemsets because the items are derived from demonstrated experiments. Finally, we get 39 frequent itemsets by mining the AMPK regulation data. Table 5 represents reconverted frequent itemsets from non-negative integer.

From the observation, we can see the possible existence of the associations listed below:

1.  $I^+ \Rightarrow \alpha_{2a} \uparrow$  implies that  $\alpha_{2a}$  may be high expressed under high intensity stimulation;
2.  $I \Rightarrow \alpha_{1e} \downarrow \wedge \alpha_{2e} \downarrow$  implies if given a moderate intensity stimulation  $\alpha_{1e}$  and  $\alpha_{2e}$  can have no change;
3.  $\alpha_{2a} \downarrow \Rightarrow \alpha_{1e} \downarrow \wedge \alpha_{2e} \downarrow$  implies if  $\alpha_{2a}$  has no change  $\alpha_{1e}$  and  $\alpha_{2e}$  may have no change too;

4.  $I \Rightarrow \alpha_{1a} \downarrow \wedge \alpha_{1e} \mid$ , which implies if given a moderate intensity stimulation  $\alpha_{1a}$  may have no change and  $\alpha_{1e}$  may be expressed.

The experiments demonstrate that data mining can not only discover frequent itemsets but can also reduce the cost of labor, resources and other associated activities. Experiments can be conducted based on the implied association, thereby reducing the number of experiments. For example, if  $\alpha_{2a}$  has no specific conditions, we can predict that  $\alpha_{1e}$  and  $\alpha_{2e}$  based on the assumed third association do not change in expression. Furthermore, it can reduce the time of experiments by avoiding unnecessary stimuli. For example, the first association implies that the *high intensity* exercises can result in high expressed  $\alpha_{2a}$ . If we want to observe  $\alpha_{2a}$  to be high expressed, we do not need an *intense* stimuli.

## 5 Conclusions

Protein kinase has emerged as an important indicator in the regulation of cell metabolism. A framework has been proposed with the aim of finding frequent itemsets. Unlike the conventional market basket data, the status of items derived from kinase regulation data is considered important. Also, the items that have low occurrence in existing experiments are pruned and not used in generating frequent itemsets. Moreover, additional pruning criteria and *support* are used to eliminate uninteresting itemsets, thereby vastly reducing the search space. Our framework was demonstrated using the AMPK regulation dataset with reference to human skeletal muscle. Most of the identified frequent itemsets are logical in a biological sense, and will provide a good foundation for further investigation.

## Acknowledgement

The work reported in this paper was partially supported by the Australian Research Council's Discovery Project Grant DP0559251.

## References

1. Hieter P. and Boguski M., Functional Genomics: It's All How You Read It the entire genome sequence, *SCIENCE*, 278(24), pp 601-602, Oct 1997.
2. Hunter L., *Artificial Intelligence and Molecular Biology*, AAAI, 1993.
3. Musi N, Fujii N, Hirshman MF, Ekberg I, Froberg S, Ljungqvist O, Thorell A, and Goodyear LJ (2001), AMP-activated Protein Kinase (AMPK) Is Activated in Muscle of Subjects with Type 2 Diabetes During Exercise, *Diabetes*, **50**, pages 921-927.
4. Beg Z.H., Allmann D.W. and Gibson D.M. (1973), Modulation of 3-hydroxy-3-methylglutaryl Coenzyme: A Reductase Activity with cAMP and with protein fractions of rat liver cytosol, *Biochem Biophys Res Commun*, **54**, pp 1362-1369.
5. Carlson D., Fryer L.G. and Woods A. (1973), Regulation of Hepatic Acetyl Coenzyme A Carboxylase by Phosphorylation and Dephosphorylation, *Journal of Bio Chem*, **248**, pp 378-380.

6. Fischer RS, Quinlan MP, While E1A can facilitate epithelial cell transformation by several dominant oncogenes, the C-terminus seems only to regulate rac and cdc42 function, but in both epithelial and fibroblastic cells. 269(2), *Virology*, pp 404-419, 2000.
7. Chengqi Zhang, and Shichao Zhang, Association Rule Mining: Models and Algorithms, LNAI 2307, Springer-Verlag, Germany.
8. Han J., Pei J. and Yin Y., Mining frequent patterns without candidate generation, Proceedings of the ACM SIGMOD International Conference on Management of Data, pp 1-12, 2000.
9. Doddi S, Marathe A, Ravi SS, Torney DC., Discovery of Association Rules in Medical Data, *Med Inform Internet Med*, **26**, pp 25-33, 2001.
10. Stilou S, Bamidis PD, Maglaveras N, Pappas C., Mining Association Rules from Clinical Databases: An Intelligent Diagnostic Process in Healthcare, *Medinfo*, **10**, pp 1399-1403, 2001.
11. Sakamoto K., McCarthy A., Smith D., Green. K.A., Hardie D.G., Ashworth A. and Alessi D.R. (2005), Deficiency of LKB1 in skeletal muscle prevents AMPK activation and glucose uptake during contraction, *The EMBO Journal*, **24**, pp 1810-1820, 2002.
12. Durante PE., Mustard KJ., Park SH., Winder WW. and Hardie DG., Effects of Endurance Training on Activity and Expression of AMP-activated Protein Kinase Isoforms in Rat Muscles, *Am J Physiol Endocrinol Metab*, **283**, pp 178-186.
13. Agrawal R., Imielinski T., and Swami A. (1993), Mining Association Rules between Sets of Items in Large Databases, *Proceeding of ACM-SIGMOD International Conference on Management of Data*, pp 207-216, 1993.

# Constructing Bayesian Networks from Association Analysis

Ohm Sornil and Sunatashee Poonvutthikul

Department of Computer Science, National Institute of Development Administration  
118 Seri Thai Road Bangkapi Bangkok, Thailand 10240  
[{osornil, sunatashee}@as.nida.ac.th](mailto:{osornil, sunatashee}@as.nida.ac.th)

**Abstract.** This paper presents an automatic Bayesian network construction algorithm where association analysis is employed to guide the construction of network structure. The proposed method is studied in context of data imputation together with a previously proposed technique for automatic Bayesian network construction, Backpropagation neural networks, and two traditional data imputation techniques. The results show that the proposed method performs better or at least as well as does the best of other methods in 84.62% of the cases.

## 1 Introduction

Bayesian network (or Bayesian belief network) [18] is a powerful model for reasoning and representing uncertainty in knowledge by capturing conditional dependencies between different knowledge components. A Bayesian network is a directed acyclic graph with a conditional probability distribution associated with each node. Nodes in a network represent variables while edges represent dependence relations between variables.

Bayesian networks are usually constructed from the knowledge of domain experts. A number of automatic network learning algorithms have been proposed. A large class of techniques relies on heuristic searches of structures that maximize some scoring criteria. During the exploration process, the fitness of each candidate structure is evaluated against the data. Various functions are proposed, such as those based on entropy [8, 15], Bayesian scoring [2, 5, 9], as well as minimum description length [24]. Another class of algorithms [4, 16, 22, 26] analyzes independence relationships among variables and attempt to find a network that represents those relationships the most.

Some research combine both scoring and independence relationships in the construction of Bayesian networks. [5, 21] treat both approaches separately and combine the results to generate a network while [1] takes into account the relationships in the calculation of network scores. [14] proposes a semi-automatic approach by providing an automated tool for acquiring domain expert knowledge to supplement the construction of topologies and the estimation of probabilities.

In this paper, we present an automatic approach for learning Bayesian networks from data whose structures are guided by association analyses. The proposed method is evaluated against other methods in the context of data imputation on standard data sets.

The rest of the paper is organized as follows: Section 2 introduces Bayesian network and association analysis; Section 3 presents the proposed algorithm for constructing Bayesian networks from data; Section 4 describes the experimental evaluation and results; and Section 5 provides the concluding remarks.

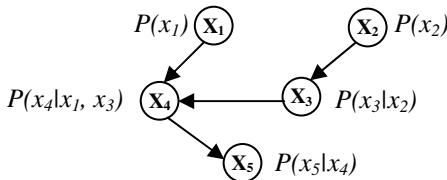
## 2 Preliminaries

### 2.1 Bayesian Networks

A Bayesian network is a classification model which can predict class membership probabilities for output variables. The model is based on Bayes' theorem. A Bayesian network, shown in Figure 1, consists of two main components: a directed acyclic graph (DAG) whose nodes represent random variables and edges represent probabilistic dependences; and a set of conditional probability tables (CPTs), one for each random variable which specifies the conditional distribution  $P(X | Parents(X))$ , where  $Parents(X)$  are the parents of  $X$ . The joint probability of any tuple  $(x_1, x_2, \dots, x_n)$  corresponding to the variables  $X_1, X_2, \dots$ , and  $X_n$  can be calculated as:

$$P(x_1, \dots, x_n) = \prod_{i=1}^n P(x_i | Parents(x_i))$$

The value of  $P(x_i | Parent(x_i))$  is an entry in  $X_i$ 's CPT. Any node in the network can be selected as the output variable, and its class label can be predicted.



**Fig. 1.** A Bayesian network

### 2.2 Association Analysis

Association analysis [13] attempts to discover interesting associations from data, which has been used to support business decision makings, such as analyzing customer buying transactions to extract associations among different items that customers place in their shopping baskets. The patterns discovered can be represented in form of association rules.

Let  $X, Y$  are random variables, and  $x, y$  are values of variables  $X$  and  $Y$ , respectively. An association rule can be written as:

$$X = x \Rightarrow Y = y \quad [\text{support} = s\%, \text{confidence} = c\%]$$

Rule support and confidence are two measures of rule interestingness. A support of  $s\%$  indicates that  $s\%$  of all records in the data contain  $X=x$  and  $Y=y$ . A confidence of  $c\%$  indicates that  $c\%$  of records containing  $X=x$  also contain  $Y=y$ . Association rules

are considered interesting if they satisfy both a minimum support threshold and a minimum confidence threshold which can be specified by users.

$$\begin{aligned}\text{support}(X=x \Rightarrow Y=y) &= \text{Prob}(X=x \cap Y=y) \\ \text{confidence}(X=x \Rightarrow Y=y) &= \text{Prob}(Y=y \mid X=x)\end{aligned}$$

Association rules can have multiple variables in their antecedents, such as:

$$X_1=x_1, X_2=x_2, \dots, X_n=x_n \Rightarrow Y=y \quad [\text{support}=s\%, \text{confidence}=c\%]$$

where

$$\begin{aligned}\text{support} &= \text{Prob}(X_1=x_1 \cap X_2=x_2 \cap \dots \cap X_n=x_n \cap Y=y) \\ \text{confidence} &= \text{Prob}(Y=y \mid X_1=x_1, X_2=x_2, \dots, X_n=x_n).\end{aligned}$$

### 3 Bayesian Network Construction

We first review the principle of d-separation [17], which plays an important role in our algorithm, then present the network construction algorithm.

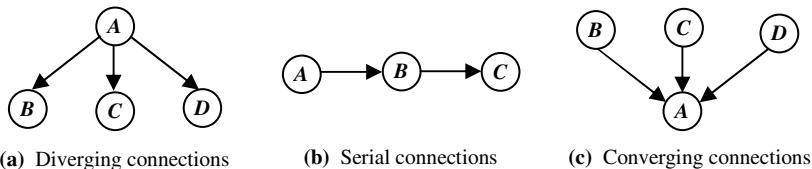
#### 3.1 Principle of D-Separation

In a Bayesian network, DAG captures dependence relationships among variables. Since every independence statement in the network satisfies a set of axioms stated in [18], we can construct a Bayesian network from data by analyzing conditional independence relationships [4]. The principle of d-separation expresses conditional independence in a graphical context.

If two variables are d-separated [17] relative to a variable  $Z$  in a directed graph, then they are conditionally independent on  $Z$  in all probability distributions such a graph can represent. That is, two variables  $X$  and  $Y$  are conditionally independent on  $Z$  if knowledge about  $X$  gives no extra information about  $Y$  once we have knowledge about  $Z$ .

Given a directed acyclic graph  $G = (V, E)$  where  $V$  is a set of nodes (vertices) and  $E \subseteq V \times V$  is a set of edges. Let  $S \subseteq V$ ; nodes  $X, Y \in V - S$ ; and a node  $Z \in S$ . The path between  $X$  and  $Y$  is blocked by  $Z$  if:

- (1) Node  $Z$  is on an undirected path between  $X$  and  $Y$ ; and  $Z$  has diverging connections toward the other nodes (see Figure 2a).
- (2) Node  $Z$  is on a serial connection path between nodes  $X$  and  $Y$  (see Figure 2b).
- (3) Node  $v \in V$  as well as its descendants are not in  $S$  and have converging connections toward them from other nodes (see Figure 2c).



**Fig. 2.** Blockages as defined in the principle of d-separation

### 3.2 Bayesian Network Construction Algorithm

The Bayesian network construction algorithm consists of three phases. In the first phase, association analysis is performed. A set of association rules is selected and passed through the next phase. In the second phase, an initial network is constructed by considering local independence. It is possible that in this phase edges are wrongly added into the network. The third phase is needed to prune out those edges.

#### **Phase 1:** Generating Association Rules from Data

Association analysis identifies associations among variables which can naturally be used to guide the causal relationships among variables of the network. Association analysis is performed; and rules with support and confidence values above specified thresholds are selected as output of this phase.

#### **Phase 2:** Constructing An Initial Network Topology Based on Local Independence

Once a set of association rules are identified, they are used to construct an initial structure of the Bayesian network.

##### *NetworkConstruction* (a set of association rules)

1. Sort association rules according to their confidence values, support values, then number of antecedents in a non-increasing order;
2. For each association rule considered {
3.     If a rule is  $X_1=x_1, X_2=x_2, \dots, X_n=x_n \Rightarrow Y=y$  {
4.         For  $i = 1$  to  $n$  {
5.             If (a directed edge from  $X_i$  to  $Y$  does not exist)  
AND ( $X_i$  and  $Y$  are not d-separated)  
AND (adding a directed edge from  $X_i$  to  $Y$  does not create a cycle  
involving any of them) {
6.                 Place and edge from node  $X_i$  to node  $Y$ ;
7.         }
8.     }
9. }
10. }

#### **Phase 3:** Edge Pruning

For each edge in the network, we temporarily remove the edge and test whether its two end nodes are d-separated without it. If they are, the edge is inserted back to the network; otherwise the edge is permanently removed.

##### *EdgePruning* (a network graph $(V,E)$ )

1. For each directed edge  $e$  connecting  $X_i$  and  $X_j$  {
2. Temporarily remove edge  $e$ ;
3.     If (*D-SeparationTest*( $X_i, X_j, (V, E - \{e\})$ ) == True) {
4.         Insert  $e$  back to the network;
5.     } else {
6.         Permanently remove edge  $e$ ;
7.     }
8. }

##### *boolean D-SeparationTest*( $X_i, X_j, (V,E)$ )

1. Find all paths between  $X_i$  and  $X_j$ ;
2. While there still is a path between  $X_i$  and  $X_j$  {

3. Find a node  $v_i$  which can block maximum number of paths and put it in a set  $C$ ;
4. Remove all paths blocked by  $v_i$ ;
5. Shorten existing paths by removing nodes in  $C$ ;
6. }
7. For each combination of  $x_i$  and  $x_j$  {
8.     Compute dependence score as:  $\frac{P(X_i = x_i \cap X_j = x_j \cap_{C_k \in C} C_k = c_k)}{P(\cap_{C_k \in C} C_k = c_k)}$
9.     If (dependence score >  $\epsilon$ ) {
10.         Return False;
11.     }
12. }
13.     Return True;

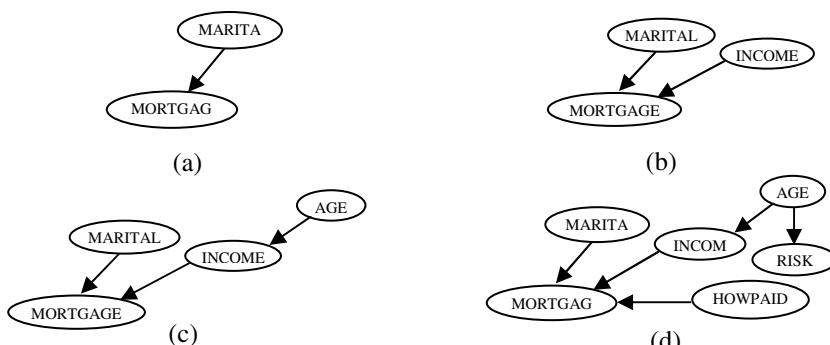
Once the network structure is defined, CPT at each node can be calculated normally [17].

Next, we give an example computation in Phase 2 of the algorithm. Suppose the association rules extracted from the data are sorted and given in Table 1. The first rule is considered, and the corresponding graph is displayed in Figure 3a. Next, the second rule is considered. Since the edge from MARITAL to MORTGAGE has existed, only an edge from INCOME to MORTGAGE is added into the network (see Figure 3b).

The third rule is then considered; an edge from AGE to INCOME is placed (see Figure 3c). After considering the fourth and the fifth rules, the resulting network is

**Table 1.** An example set of association rules, sorted by confidence then support values

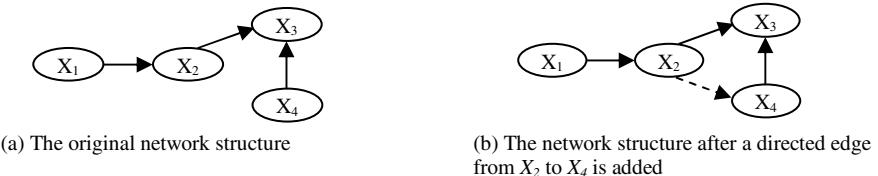
Consequent	Antecedent	% Support	% Confidence
mortgage=y	marital=married	50.63	100
mortgage=y	marital=married and income=A	33.87	100
income=A	age=B	34.66	99.22
risk=bad profit	age=B	34.66	88.47
mortgage=y	howpaid=monthly	49.20	88.15
income=A	risk=bad profit	58.46	86.47
mortgage=y	income=A	67.88	74.19



**Fig. 3.** Structures of a Bayesian network constructed from data in Table 1

shown in Figure 3d. Considering the remaining rules does not suggest any new edge to the network, thus the final network is shown in Figure 3d.

Next, we discuss the role of Phase 3 of the algorithm. According to local independence, in Figure 4b a directed edge from  $X_2$  to  $X_4$  (the dashed line) is inserted into the original network (Figure 4a) since the insertion does not violate the requirements in Phase 2 of the algorithm. However,  $X_2$  and  $X_3$  are now d-separated by  $X_4$  which violates the principle. Phase 3 is thus required to ensure that the principle of d-separation holds for the entire network.



**Fig. 4.** An edge from  $X_2$  to  $X_4$  is added to the network due to local independence

## 4 Performance Evaluation

In this section, the proposed algorithm is evaluated in the context of data imputation, against a previously proposed Bayesian network construction method by Cheng *et al.* [4], Backpropagation neural networks, and two traditional approaches for imputing missing values, i.e., mode and median imputation.

The data used in this study are the Risk, Nursery, and Shopping data sets, from the UCI KDD Archive [25]. Their characteristics are shown in Table 2.

Numeric variables are first discretized by arranging values into multiple bins with equal frequency. Bayesian network construction software described in [3, 4] is provided by the authors. Backpropagation neural network models are created by Clementine™ software. Apriori algorithm [13] in Clementine™ is used to generate association rules. The minimum support and minimum confidence values for our algorithm are 10% and 50%, respectively.

The results for the Risk collection are shown in Table 3. We can see that in 3 out of 4 different cases, the proposed algorithm outperforms other imputation techniques

**Table 2.** Characteristics of the data sets used in the study

Data Sets	Number of Variables	Number of Records
Risk	11	4117
Nursery	9	12,960
Shopping	15	786

when the network is constructed from association rules with maximum number of antecedents of 2 and 3. However, networks with maximum antecedent of 3 does not yield significant improvement over those with maximum antecedent of 2 while complexities of association analysis as well as network construction are far greater. Thus,

in further experiments we limit our study at the maximum number of antecedents for association rules at 1 and 2 only.

Results for the Nursery and Shopping data sets are shown in Table 4 and 5, respectively.

## Risk Data Set

Training: 3,705 records

Testing: 412 records

**Table 3.** Results of the Risk data set

Missing Variable <b>Age</b>	Mode	Median	Neural Network	Bayesian Network in [3]	<b>Proposed Network</b>		
					<b>Max Antecedent</b>	<i>I</i>	<i>2</i>
#Correct Predictions	138	126	253	262	276	280	280
%Correct Predictions	33.50	30.58	61.41	63.60	66.99	67.96	67.96
Missing Variable <b>Mortgage</b>	Mode	Median	Neural Network	Bayesian Network in [3]	<b>Proposed Network</b>		
					<b>Max Antecedent</b>	<i>I</i>	<i>2</i>
#Correct Predictions	326	326	326	316	326	326	326
%Correct Predictions	79.13	79.13	79.13	76.70	79.13	79.13	79.13
Missing Variable <b>Howpaid</b>	Mode	Median	Neural Network	Bayesian Network in [3]	<b>Proposed Network</b>		
					<b>Max Antecedent</b>	<i>I</i>	<i>2</i>
#Correct Predictions	209	209	231	277	269	285	285
%Correct Predictions	50.73	50.73	56.07	67.23	65.29	69.17	69.17
Missing Variable <b>Marital</b>	Mode	Median	Neural Network	Bayesian Network in [3]	<b>Proposed Network</b>		
					<b>Max Antecedent</b>	<i>I</i>	<i>2</i>
#Correct Predictions	213	213	320	369	320	329	329
%Correct Predictions	51.70	51.70	77.67	89.56	77.67	79.85	79.85

## Nursery Data Set

Training: 11,664 records

Testing: 1,296 records

**Table 4.** Results of the Nursery data set

Missing Variable <b>Nurse</b>	Mode	Median	Neural Network	Bayesian Network in [3]	<b>Proposed Network</b>	
					<b>Max Antecedent</b>	<i>1</i>
#Correct Predictions	875	870	1123	1143	1146	1149
%Correct Predictions	67.52	67.13	86.65	88.19	88.43	88.66
Missing Variable <b>Health</b>	Mode	Median	Neural Network	Bayesian Network in [3]	<b>Proposed Network</b>	
					<b>Max Antecedent</b>	<i>1</i>
#Correct Predictions	875	866	835	969	969	969
%Correct Predictions	67.52	66.82	64.43	74.77	74.77	74.77

**Table 4.** (*continued*)

Missing Variable <b>Finance</b>	Mode	Median	Neural Network	Bayesian Network in [3]	<b>Proposed Network</b>	
					<b>Max Antecedent</b>	<b>1</b>
#Correct Predictions	648	648	658	-	646	662
%Correct Predictions	50.00	50.00	50.77	-	49.85	51.70
Missing Variable <b>Parent</b>	Mode	Median	Neural Network	Bayesian Network in [3]	<b>Proposed Network</b>	
					<b>Max Antecedent</b>	<b>1</b>
#Correct Predictions	404	429	542	551	543	551
%Correct Predictions	31.17	33.10	41.82	42.52	41.90	42.52
Missing Variable <b>Housing</b>	Mode	Median	Neural Network	Bayesian Network in [3]	<b>Proposed Network</b>	
					<b>Max Antecedent</b>	<b>1</b>
#Correct Predictions	411	445	445	480	471	480
%Correct Predictions	31.71	34.34	34.34	37.04	36.34	37.04

Results for the Nursery and Shopping data sets show that the proposed technique with maximum antecedent of 2 outperforms all other techniques in 44.44% of the cases and performs equally well with the best method of the reference group in 44.44% of the cases. In addition, the proposed algorithm is able to impute data for cases that the algorithm in [3,4] is not able to.

## Shopping Data Set

Training: 708 records

Testing: 78 records

**Table 5.** Results of the Shopping data set

Missing Variable <b>Age</b>	Mode	Median	Neural Network	Bayesian Network in [3]	<b>Proposed Network</b>	
					<b>Max Antecedent</b>	<b>1</b>
#Correct Predictions	20	21	30	30	32	32
%Correct Predictions	25.64	26.92	38.46	38.46	41.03	41.03
Missing Variable <b>Gender</b>	Mode	Median	Neural Network	Bayesian Network in [3]	<b>Proposed Network</b>	
					<b>Max Antecedent</b>	<b>1</b>
#Correct Predictions	40	40	40	42	37	37
%Correct Predictions	51.28	51.28	51.28	53.85	47.44	47.44
Missing Variable <b>Marital</b>	Mode	Median	Neural Network	Bayesian Network in [3]	<b>Proposed Network</b>	
					<b>Max Antecedent</b>	<b>1</b>
#Correct Predictions	19	15	25	25	26	26
%Correct Predictions	24.36	19.23	32.05	32.05	33.33	33.33
Missing Variable <b>Income</b>	Mode	Median	Neural Network	Bayesian Network in [3]	<b>Proposed Network</b>	
					<b>Max Antecedent</b>	<b>1</b>
#Correct Predictions	65	65	65	65	65	65
%Correct Predictions	83.33	83.33	83.33	83.33	83.33	83.33

## 5 Conclusion

Bayesian network is a powerful tool for reasoning and representing uncertainty in knowledge by capturing conditional dependence between different knowledge components. In this paper, association analysis which has been used to support business decision makings is proposed as a guide for the construction of Bayesian networks. A series of comparative studies is conducted in the context of data imputation, against a previously proposed technique, Backpropagation neural networks, and two traditional data imputation methods. The results demonstrate the applicability of the proposed technique which performs better or at least as well as other methods in 84.62% of the cases, with 46.15% of best performance.

## References

1. S. Acid and L. M. De Campos, A Hybrid Methodology for Learning Belief Networks: Benedict International Journal of Approximate Reasoning, 1910, 309-315, 2001.
2. W. Buntine, Theory Refinement of Bayesian Networks, in Proceedings of the Seventh Conference on Uncertainty in Artificial Intelligence, 52-60, 1991.
3. J. Cheng (2001). Bayesian Belief Network Software [Computer Program]. Available: <http://www.cs.ualberta.ca/~jcheng/bnpc.htm>
4. J. Cheng, D. A. Bell, W. Liu, "An algorithm for Bayesian belief network construction from data" in Proc. 6th International Workshop on Artificial Intelligence and Statistics, 1997.
5. G. F. Cooper and E. Herskovits, A Bayesian Method for the Induction of Probabilistic Networks from Data, Machine Learning, 9, 309-348, 1992.
6. D. Dash and M. Druzdzel, A Hybrid Anytime Algorithm for the Construction of Causal Models from Sparse Data, in Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence, 142-194, 1999.
7. R. O. Duda, P. E. Hart and D. G. Stork, Pattern Classification, 2nd ed. New York: A Wiley-Interscience Publication, 2000.
8. L. M. De Campos, Independency Relationships and Learning Algorithms for Singly Connected Networks, Journal of Experimental and Theoretical Artificial Intelligence, 10, 511-549, 1998.
9. N. Friedman and D. Koller. (2001). Learning Bayesian Networks from data [Online]. Available: <http://www.cs.huji.ac.il/~nirf/Nips01-Tutorial/Nips-tutorial.pdf>
10. S. Glesner, D. Koller, "Constructing Flexible Dynamic Belief Networks from First-Order Probabilistic Knowledge Bases." in 1995 Proc. European Conference on Symbolic and Quantitative Approaches to Reasoning and Uncertainty (ECSQARU '95), pp. 217--226.
11. P. Haddawy and R. A. Krieger, "R.A., Principled construction of minimal bayesian networks from probability logic knowledge bases." Submitted to Journal of AI Research.
12. P. Haddawy, "Generating Bayesian Networks from Probability Logic Knowledge Bases." in Proc. Tenth Conference on Uncertainty in Artificial Intelligence (UAI '94), Morgan Kaufmann , 1994. pp. 262—269.
13. J. Han and M. Kamber, Data Mining: Concepts and Techniques, San Francisco: Morgan Kaufmann publishers, 2001.
14. E. M. Helsper, L. C. van der Gaag, A. J. Feelders, W. L. A Loeffen, P. L. Geenen, and A. R. W. Elbers, "Bringing order into Bayesian-network construction." in Proc. 3rd International Conference on Knowledge Capture, ACM Press, New York, 2005. pp. 121-128.

15. E. Herskovits, Computer-Based Probabilistic Networks Construction, Ph.D. thesis, Medical Information Sciences, Stanford University, 1991.
16. J. F. Huete and L. M. De Campos, Learning Causal Polytrees, Lecture Notes in Computer Science, 747, 180-185, 1993.
17. F. V. Jensen, Bayesian Networks and Decision Graphs, New York: Springer-Verlag New York,.Inc., 2001.
18. J. Pearl, Probabilistic Reasoning in Intelligence Systems: Networks of Plausible Inference, Morgan Kaufmann, 1988.
19. K. Murphy. (1998). A Brief Introduction to Graphical Models and Bayesian Networks. [Online]. Available: <http://www.ai.mit.edu/~murphyk/Bayes/bayes.html>
20. C. Murray. Bayesian Belief Networks [Online]. Available: <http://www.murrayc.com/learning/AI/bbn.shtml>
21. M. Singh and M. Valtorta, An Algorithm for the Construction of Bayesian Network Structures from Data, in Proceedings of the Ninth Conference on Uncertainty in Artificial Intelligence, 259-265, 1993.
22. P. Spirtes and C Meek, Learning Bayesian Networks with Discrete Variables in Data, in Proceedings of the First International Conference on Knowledge Discovery and Data Mining, 294-299, 1995.
23. A. Storkey. (2003). Tutorial: Introduction to Belief Networks [Online]. Available: <http://www.anc.ed.uk/amos/belief.html>
24. Suzuki, A Construction of Bayesian Networks from Databases Based on the MDL Principle, in Proceedings of the Ninth Conference on Uncertainty in Artificial Intelligence, 266-273, 1993. 25.
25. UCI KDD Archive (2006). Available: <http://kdd.ics.uci.edu/>
26. N. Wermuth and S. Lauritzen, Graphical and Recursive Models for Contingence Tables, Biometrika, 72, 537-552, 1983.

# Bayesian Approaches to Ranking Sequential Patterns Interestingness

Kuralmani Vellaisamy and Jinyan Li

Institute for Infocomm Research  
21 Heng Mui Keng Terrace, Singapore 119613  
{vkmani, jinyan}i2r.a-star.edu.sg

**Abstract.** One of the main issues in the rule/pattern mining is of measuring the interestingness of a pattern. The interestingness has been evaluated previously in literature using several approaches for association as well as for sequential mining. These approaches generally view a sequence as another form of association for computations and understanding. But, by doing so, a sequence might not be fully understood for its statistical significance such as dependence and applicability. This paper proposes a new framework to study sequences' interestingness. It suggests two kinds of Markov processes, namely Bayesian networks, to represent the sequential patterns. The patterns are studied for statistical dependencies in order to rank the sequential patterns interestingness. This procedure is very shown when the domain knowledge is not easily accessible.

## 1 Introduction

Understanding the relationships between the attributes in a vast database is very difficult if a database is an essential part of any complex industrial processes. In [1] and [2] methods such as association mining and sequential mining have been introduced to reveal the underlying relationships or patterns from a database. Association mining focuses on the permutation of events while sequential mining concentrates on an ordered list of events, [2-4]. An ordered list of events are sequential events which are time dependent events consist of past (antecedent) and present (consequent) events. All the generated patterns by these methods are not really worthy. Because some do not have good properties like non-randomness, strong association, etc. Identifying rich and useful patterns from the list needs some good metrics. Many metrics are recommended in the literature to identify the interesting patterns [5-8]. But they mainly focus on the association, not on the sequential patterns. However, in general, these metrics are for: i) understanding the associations between the elements in a database, ii) verifying the validity of associations, and iii) understanding the intra-relationships such as cause and effect. But most of these metrics pay poor attention on the second proposition, because they always assume the found associations between elements is true. For instance, in an association  $\langle ABC \rangle$ , the associations between  $A \& B$ ,  $A \& C$ , and  $B \& C$  are considered to be true just because  $\langle ABC \rangle$  is true in an association rule  $\langle AB \rightarrow C \rangle$ . This means a sequence is being verified for the association between the past and present events assuming that the past events are truly associated themselves. In the above example, the association between  $A$  and  $B$  are

considered to be true. In case, if it is not then the chance of appearing this association rule again in a transaction will be lower. This will eventually create a lot of false associations. To qualify for a useful association pattern, a pattern must confirm all the three stated propositions are true. However, in the case of sequential mining, these requirements are not alone sufficient enough to support for achieving an interesting sequential pattern. Because it needs conditional dependencies in addition to simple dependency in order to ensure its forward association (directionality). In general, a sequence needs to have sufficient intra dependencies among the elements and also conditional dependencies in order to qualify for a true or non-random sequence apart from exhibiting a good representation in the dataset. Thus, this work aims to understand and identify the potential associations between the elements in order to qualify for a useful sequential pattern, which is the primary motivation of this work. In this work a pattern is being defined as an interesting pattern if it has significant dependency between the elements.

This paper proposes a new framework to study a sequence's interestingness in order to understand a sequence as a whole by means of their statistical dependencies. A more statistical dependency makes a sequence is more interesting, because of its closeness to the Pearson correlation theory. This work introduces two kinds of Bayesian networks where each network emphasis one assumption of dependency such as either strong or weak. A strong dependency provides a sequence is more interesting sequence and a weak dependency provides a sequence is less interesting.

Various developments in the sequential mining are shown in [9]. Five methods are discussed for computing the strengths such as support in [10]. A user specified methodology to reduce the number of uninteresting patterns from the discovered rules has been proposed in [11]. Various rule-interesting measures have been discussed in [12] in addition to introducing a new criterion to measure attribute surprisingness, as a factor influencing the interestingness of discovered rules. In [13], two pruning methodologies have been proposed and in which one rule suggests that a repeated occurrence of an event should be discarded. About thirty-four interesting metrics are discussed in [8]. A similar kind is also proposed in [14] to observe the interestingness of an association rule. Further information about sequential mining and its interestingness are addressed in [15-18].

This paper is organised in 6 sections. Section 2 discusses the association and sequential patterns; section 3 proposes a framework for identifying interesting sequences. Section 4 gives an algorithm while section 5 studies a simulated and real examples. Section 6 concludes the work.

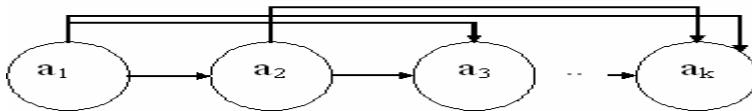
## 2 Patterns and Their Effects

A pattern is said to be a sequential pattern if it has an ordered list of elements, which usually occur one after another at different phases of time. An association pattern has a group of elements which all occur at a particular point of time. The first one observes sequential dependency whereas the second one maintains group dependency.

Let  $\mathcal{D}$  be a transactional database in which  $\Omega = \{a_{ij}\}$  with  $i = 1, \dots, n$  and  $j = 1, \dots, m$ . Let a set  $\Phi$  has a set of  $m$  attributes  $\langle a_1, \dots, a_m \rangle$  and  $n$  be a list of transactions occurred in an operational data where each transaction is noted with an identifier

$N_i$ ,  $i = 1, \dots, n$ . Assume further that  $\sum a_{ij} = 1$  if  $j$  is appeared in the database otherwise it becomes zero. Suppose  $\sum_k a_{ijk} \geq 1$  for each  $i$  and  $j$  (each cell) then  $\sum_k a_{ijk} = 1$  for all  $i$  and  $j$ , which assumes a sampling without replacement. Following definitions clarify further about the sequential and association patterns:

*Definition 1:* Let  $\Phi$  be a generalized sequence  $\Phi = \langle a_1, \dots, a_m \rangle$ . We define a sequence  $\Lambda$  as  $\langle a_1, \dots, a_k \rangle \in \Phi$  such that  $k \leq m$  and  $t(a_j) \leq t(a_{j+1})$ , where  $t(a_j)$  is a time of occurrence of an event  $a_j$ . Then  $\Lambda$  can be represented by an influence diagram shown in Fig.1.



**Fig. 1.** Influence diagram of a sequential pattern

The joint probability of  $\Lambda$  is:

$$P(a_1 \dots a_k) = P(a_1) \prod_2^k P(a_j / a_1 \dots a_{j-1}) \quad (1)$$

*Definition 2:* With the definition 1, we further define a sequence  $\Lambda = \langle a_1, \dots, a_k \rangle \in \Phi$  such that  $k \leq m$  and  $t(a_j) = t(a_{j+1})$  where  $t(a_j)$  is the time of an event  $a_j$ . This defines an association.

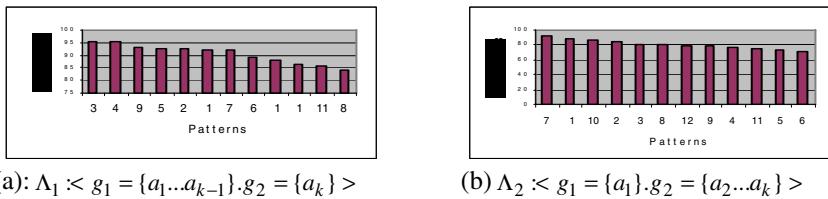
It is noted that if a sequence  $\langle a_j, \{a_{j+h_2}, a_{j+h_1}\} \rangle$  with  $t(a_{j+h_2}) = t(a_{j+h_1})$  and  $h_2 > h_1$ , then it is formatted as  $\langle a_j, a_{j+h_1}, a_{j+h_2} \rangle$  based on the hierarchy. Further, if  $P_a$  and  $P_s$  represent association and sequential joint probabilities then they satisfy the relation  $P_a(\Lambda) \geq P_s(\Lambda)$  when  $N_j = C_j$  for all  $j$ , where  $C$  is a subject id. Following definitions define the measures that are frequently used in the literature for identifying interesting sequential and association patterns.

*Definition 3:* Let  $g_1 = \{a_1, \dots, a_l\}$  and  $g_2 = \{a_{l+1}, \dots, a_k\}$  be set of two events in a  $\mathcal{D}$ . Then support of a sequence  $\Lambda : g_1 \rightarrow g_2$  is denoted as  $P(\Lambda) = P(g_1 g_2)$ . Alternatively, an absolute count or a relative percentage of happening of  $\Lambda$  in  $\mathcal{D}$  is known as support.  $P(g_1 g_2)$  and  $P(g_i)$  represent joint and marginal probabilities.

*Definition 4:* For a given sequence  $\Lambda : g_1 \rightarrow g_2$ , the lift is  $P(g_1 g_2) / P(g_1)P(g_2)$ . A ratio of conditional expectation of an element in a rule  $\Lambda$  of  $\mathcal{D}$  to the marginal expectation of that element in a  $\mathcal{D}$  is known as lift.

*Definition 5:* For a given  $\Lambda : g_1 \rightarrow g_2$ , the *confidence* is measured by  $P(g_2 / g_1)$ . A relative percentage of items satisfying rule  $\Lambda$  in  $g_1$  is termed as *confidence*. This is also known as conditional probability of  $g_2$  given  $g_1$ .

*Support* identifies the existence of association in a  $D$ . *Lift*, roughly, facilitates to identify a sequence's dependency or randomness of an association. The cause and effect in a sequence is usually measured with *confidence*. Similar to the association, a sequence has multiple measures of *confidence* and *lift* with respect to the size of  $g_1$  and  $g_2$ . For instance, consider an example from [6]. There are 11 sequences in this example and each sequence satisfies *minimum support* criteria, but they have different listings, see Fig.2. Take a set of  $g_1$  and  $g_2$  such as  $\Lambda_1 : g_1 = \{a_1 \dots a_{k-1}\} \cdot g_2 = \{a_k\} >$  and  $\Lambda_2 : g_1 = \{a_1\} \cdot g_2 = \{a_2 \dots a_k\} >$  for each sequence. With respect to these settings, the order of sequences are  $\{3, 4, 9, 5, 2, 1, 7, 6, 10, 12, 11, 8\}$  and  $\{7, 1, 10, 2, 3, 8, 12, 9, 4, 11, 5, 6\}$ . Identifying a best list is very hard, because it does not have any unique measure to show its interestingness. But it is easy to understand a sequence as a whole rather than with two pieces. Following section provides a detailed framework how to do this.

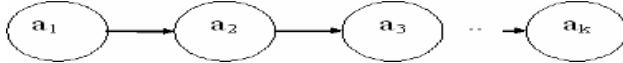


(a):  $\Lambda_1 : g_1 = \{a_1 \dots a_{k-1}\} \cdot g_2 = \{a_k\} >$       (b):  $\Lambda_2 : g_1 = \{a_1\} \cdot g_2 = \{a_2 \dots a_k\} >$

**Fig. 2.** Two kinds of interesting rules observed based on the two settings

### 3 Interestingness Framework

Patterns are represented in various forms like classification, association, clustering etc. However, finding relevant and useful patterns from the discovered patterns is a demanding task for data miners, because of the poor knowledge they have in the respective domains. Particularly, identifying a potential sequential pattern is very useful considering the amount of rules generated by the algorithms. To avoid losing a potential pattern while in the process of elimination of unwanted patterns, a good statistical methodology is necessary. Although many measures are addressing these issues in the literature, but primarily they focus on the end results ignoring the fact of sequential dependence, [7,8,12]. This work is mainly interested in taking advantage of sequential dependence to measure a sequential pattern's interestingness. Sequential dependence is one way or other is very much related to Bayesian networks (BN) because of their similarities in their property such as directed acyclic graphical (DAG) model, [19]. In this work two kinds of networks are suggested. The first network is shown in Fig.1 and the following is the second one.



**Fig. 3.** An another influence diagram of a sequential pattern

The joint probability of this sequence  $\Lambda$  is:

$$P(a_1..a_k) = P(a_1) \prod_{j=2}^k P(a_j / a_1..a_{j-1}) \quad (2)$$

The reason for choosing these two kinds of networks is of to address two sorts of problems such as strong and weak dependencies within a sequence. To understand it clearly, we have illustrated the scenarios in Fig.4. Fig.4a and 4b indicate strong and weak dependencies. The occurrence of event {ABC} in Fig.4a was very much influenced by events {A}, {B} and {AB} whereas the same event was caused only by previous event alone such as {B} given {A} as shown in Fig.4b. With these, the following measures are defined to help to identify the interesting patterns:

A	B	C	D
P(A)-->	P(AB)-->	P(ABC)-->	P(ABCD)

(a): Strong Dependence

A	B	C	D
P(A)-->	P(AB)-->	P(BC)-->	P(CD)

(b): Weak Dependence

**Fig. 4.** (a) and (b) represent definition 1 and figure 3 respectively

*Definition 6:* If  $\Lambda = \langle a_1, \dots, a_k \rangle$  and  $P(a_k) \geq 0$  for all  $k$ , then Bayesian strong dependency, BSD, is defined as:

$$P(a_1..a_k) = \prod_{j=2}^k P(a_j / a_1..a_{j-1}) / P(a_j) \quad (3)$$

*Definition 7:* If  $\Lambda = \langle a_1, \dots, a_k \rangle$  and if  $P(a_k) \geq 0$  for all  $k$ , then Bayesian weak dependency, BWD, is defined as:

$$P(a_1..a_k) = \prod_{j=2}^k P(a_j / a_{j-1}) / P(a_j) \quad (4)$$

*Definition 8:* If  $\Lambda = \langle a_1, \dots, a_k \rangle$  and if  $P(a_k) \geq 0$  for all  $k$ , then Bayesian dependency ratio, *BSD/BWD*, is defined as:

$$BDR = \prod_{j=2}^k P(a_j / a_1..a_{j-1}) / P(a_j / a_{j-1}) \quad (5)$$

Definition 6 and 7 compare the marginal probabilities against their joint probabilities to understand its closeness of association within its elements. Definition 8 verifies the marginal probabilities of these two and provides a common ratio to help to classify

the magnitude of associations. For instance, for  $BDR \geq 1$ , a sequence signs for a strong dependency within its elements.

**Lemma 1:** If  $\Lambda = \langle a_1, \dots, a_k \rangle$  and if  $P(a_k) \geq 0$  for all  $k$ , then  $BDR$  will be unity if the events are either independent or conditionally independent. Let  $k=2$ , then we have  $BDR = P(a_1 a_2) / P(a_1)P(a_2 | a_1)$ . Assume  $a_1$  and  $a_2$  are independent then  $BDR = P(a_1)P(a_2) / [P(a_1)P(a_2)]$ . For  $k=3$ , assume that  $a_1, a_2$  and  $a_3$  are conditionally independent such that  $P(a_1 a_3) = P(a_1)P(a_3)$  then  $BDR$  becomes one. By induction, lemma is proved.

The above definitions provide a strong basis for the proposition two mentioned in the introduction. In order to qualify for a sequential pattern, it is also necessary to prove that a sequence has a strong serial (forward) dependency. As discussed before, this could be accomplished by verifying serial conditional dependency of the events and we call it as Bayesian confidences (BC). Following theorem provides a basis for deriving Bayesian confidences.

**Theorem 1:** If a sequence  $\Lambda_k \in \Phi$  and if  $P(a_k) \geq 0$ , then  $BC(\Lambda) = \prod_{j=j+2}^k P(\Lambda_j) / P(\Lambda_{j-1})$ .

Proof: We use induction. For  $k = 2$ , the theorem is definition 5. Suppose the theorem hold for  $k-1$  events. Let  $B = a_1, \dots, a_{k-1}$ . Then by definition 5, we have  $P(B, a_k) = P(B a_k) / P(B)$ . Next we replace  $P(B)$  by its value given in the inductive hypothesis:  $P(B) = \prod_{j=j+2}^k P(\Lambda_{j-1}) / P(\Lambda_{j-2})$  and we get the results.

From the above theorem, we can deduce the confidences as shown below:

By Eqn.1, the Bayesian confidence for strong dependency is derived as:

$$BSC = \prod_{j=j+2}^k P(a_1..a_j) / P(a_1..a_{j-1}) \quad (6)$$

By Eqn.2, the Bayesian confidence for weak dependency is derived as:

$$BWC = \prod_{j=j+2}^k P(a_j a_{j-1}) / P(a_{j-1}). \quad (7)$$

## 4 An Algorithm

In summary, this paper suggests the following algorithm in choosing an interesting pattern from a dataset. The algorithm is presented in two phases. Phase one describes that how one can choose an interesting pattern without disintegrating a sequence based on the dependency criteria proposed here. This phase identifies a rule's integrity. Phase two demonstrates that how one can choose an interesting pattern from a list of patterns that are obtained from phase one. The algorithm in phase two is very similar to the existing methodologies available in the literature. Based on the above equations and theorem, the following algorithm is designed to answer the propositions made in the introduction:

Phase 1: Find an interesting sequence  $\Lambda_{int} = \langle a_{in1}, \dots, a_{ink} \rangle$ . The aim of this phase is to identify a potential and interesting sequence:

- a. Generate all frequent or maximal patterns using Apriori [1] or any similar algorithm
- b. Let  $k$  be a maximum number of patterns observed,  $\Lambda_i$ ,  $i = 1, \dots, k$
- c. For  $i=1$  to  $k$
- d. Compute  $BSD(\Lambda_i)$ ,  $BSC(\Lambda_i)$ ,  $BWD(\Lambda_i)$ ,  $BWC(\Lambda_i)$  and  $BDR(\Lambda_i)$
- e. If  $BSD(\Lambda_i) \geq 1$ , (defn.6), and  $BWD(\Lambda_i) \geq 1$ , (defn.7), then it is known as non-random sequence, else remove it
- f. If  $BDR(\Lambda_i) \geq 1$ , then the sequence signs for strong rule else for weak rule
- g. If  $BDR(\Lambda_i) \geq 1$  and  $BSC(\Lambda_i) \geq \eta$ , then its known as very interesting rule, else remove it
- h. If  $BDR(\Lambda_i) < 1$  and  $BWC(\Lambda_i) \geq \eta$ , then its known as an interesting rule, else remove it

Phase 2: A sequential rule generation process:  $\Lambda_{int} = \langle g_{1int}, g_{2int} \rangle$ . The objective is to find right  $g_i$  so as to obtain a best rule, which will be useful to predict a future element:

- i. Let  $\Lambda$  be an interesting pattern and it has  $m$  elements
- j. Let  $g_{1j} = \langle a_1, \dots, a_j \rangle$  and  $g_{2j} = \langle a_{j+1}, \dots, a_m \rangle$  and  $\Lambda_j = \langle g_{1j}, g_{2j} \rangle$
- k. Let  $LHS=g_{1j}$  and  $RHS=g_{2j}$ . For  $j=1$  to  $m$
- l. If  $Lift(\Lambda_j) > \lambda$ , (by definition 4), then it is known as a non-random pattern, else discard
- m. If  $Conf(\Lambda_j) > \eta$ , (by definition 5), then it is known as interesting pattern, else remove it.

This algorithm is further explained with numerical examples in section 5.

## 5 Empirical Examples

**Example 1:** In this example, three elements are assumed in a sequence,  $\Lambda_i = \langle a_j, a_k, a_l \rangle$ , generated from a vast database with  $k$  attributes. Assuming that there are ten thousand sequences are generated based on random sampling satisfying definition 1 for different combination of  $j$ ,  $k$  and  $l$  of  $k$  attributes and varying support levels for each  $a_j$ . The patterns are ranked for varying user-defined thresholds. In sorting the patterns, two thresholds settings are used, one is with lift and another one is with confidence. Comparing the regular and the proposed methods, the proposed methods produce very less number of interesting patterns. For the regular method, association analogy is used as defined in definitions 3-5. Further, strong dependency produces less number of rules than the weak one. The resultant number of patterns is shown in Table 1. Applying  $BDR$  instead of  $BSD$  and  $BWD$  separately, reduces the rules further, for instance, it has only 1575 patterns consist of 759 strong and 816 weak patterns.

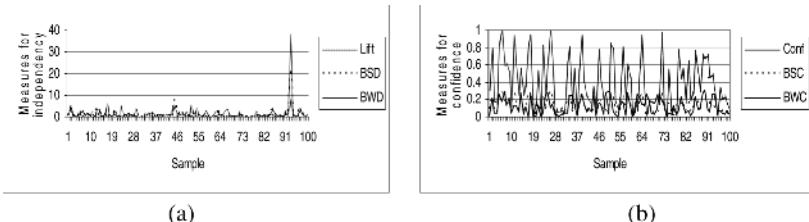
**Table 1.** A simulated results showing the number of interesting rules observed with minimum threshold levels

Threshold	Lift	BSD	BWD	Threshold	Conf	BSC	BWC
$\geq 1$	3345	2316	2536	$\geq .1$	7373	4809	5346
$\geq 2$	1322	1057	1130	$\geq .2$	5897	2959	3480
$\geq 3$	723	600	640	$\geq .3$	4784	1882	2300
$\geq 4$	473	422	446	$\geq .4$	3869	1180	1499
$\geq 5$	320	300	341	$\geq .5$	3048	714	934
$\geq 6$	239	238	263	$\geq .6$	2291	433	530
$\geq 7$	184	205	219	$\geq .7$	1662	195	258
$\geq 8$	142	168	172	$\geq .8$	1050	88	93
$\geq 9$	114	145	146	$\geq .9$	492	20	22

Apart from the dependency verification, pruning the patterns with Bayesian confidences such as BSC and BWC, brings in further substantial reduction in the total number of rules as shown in Table 2 which is about 90% of the entire set of rules. These rules can be further filtered with phase two algorithm, but it this has not been attempted in this work. When comparing the overall behavior with standard procedures, the proposed procedures show consistency at every threshold levels and also at each sample. Fig.5 evidences these.

**Table 2.** Interesting rules observed from the proposed methods

Patterns	Confidence Thresholds								
	>=0.1	>=0.2	>=0.3	>=0.4	>=0.5	>=0.6	>=0.7	>=0.8	>=0.9
BSC	441	275	180	117	81	50	21	9	1
BWC	523	371	244	162	109	64	22	10	1



**Fig. 5.** (a) Bayesian measures and the lift and (b) Bayesian confidences and the confidence

**Example 2:** Further to the above, the algorithm has been implemented on the previously discussed example in the introduction. The computed measures are shown in Table 3.

**Table 3.** A Comparison of Existing and Proposed Methods

Patterns	Count	Metrics			BSD	BWD	BDR	BSC	BWC	
		Supp	Conf	Lift						
9	low, medium, low	266	0.639	0.927	1.065	1.039	0.939	1.107	0.806	0.734
12	low, low, low, low, low	280	0.673	0.952	1.094	1.349	1.249	1.080	0.767	0.736
7	medium, medium, medium	269	0.647	0.921	1.134	1.206	1.131	1.066	0.748	0.702
11	low, low, low, low	294	0.707	0.955	1.097	1.233	1.181	1.043	0.883	0.846
10	low, medium, medium	256	0.615	0.892	1.098	1.071	1.030	1.040	0.776	0.752
6	medium, medium, low	251	0.603	0.860	0.988	1.050	1.030	1.019	0.698	0.685
5	medium, low, low	266	0.639	0.933	1.073	1.039	1.024	1.015	0.758	0.747
8	low, low, low	308	0.740	0.925	1.063	1.124	1.117	1.005	0.805	0.800
4	High,high	250	0.601	0.883	1.299	1.298	1.298	1.000	0.883	0.883
3	Low,low	333	0.800	0.920	1.057	1.057	1.057	1.000	0.920	0.920
1	medium, low	285	0.685	0.843	0.969	0.969	0.969	1.000	0.843	0.843
2	Medium, medium	292	0.702	0.864	1.063	1.063	1.063	1.000	0.864	0.864

As mentioned before, the BSD and BWD represent Bayesian strong and weak links among the elements in the sequences. These two parameters indicate whether a found sequence is a real sequence or false one. In this example, except sequence 9, the rest do possess strong characteristics for potential sequences. BSC and BWC ensure

further that whether a sequence has a reasonable confidence to say that the found sequence is an interesting sequence. This example shows the minimum Bayesian confidence as 0.685. By setting a reasonable threshold, poor sequences could be eliminated. For instance, for 80% of minimum threshold, we observe six sequences while the regular metrics provides 12 sequences.

Chosen interesting sequences based on this procedure can be processed for finding interesting sequential patterns. For instance, consider sequences 6 and 5 from the above table. These two sequences can be converted into {antecedent, consequent} patterns as shown in the following table:

**Table 4.** Choosing Interesting Sequential Patterns

	Sequence	Support	Support for Antecedent	Support for Consequent	Conf	Lift
6a	medium,medium --> low	0.603	0.702	0.87	0.8590	0.987
6b	medium --> medium,low	0.603	0.813	0.685	0.7417	1.083
5a	medium,low --> low	0.639	0.685	0.87	0.9328	1.072
5b	medium --> low,low	0.639	0.813	0.8	0.7860	0.982

This table shows two choices for each sequence because the total number of elements in each sequence is only three. Choosing a best sequential combination in order to use for target marketing or any prediction, the choosing the one with highest lift is most useful though it has less confidence in some cases, because it eliminates false sequences. The highlighted rows indicate the best sequential interesting patterns to use.

## 6 Conclusion

This work proposes a generalized framework to measure a sequence's interestingness. The proposed framework is very effective in identifying the interesting sequences from a list of candidate sequences. These methods use Bayesian networks as a tool in measuring the characteristics of a sequence by means of their intra-dependencies. These approaches do not require a sequence to split into subsequences at the beginning to see their' potentials. However, it also provides a method to see an interesting sequential rule by appropriately modifying antecedents and consequents in a found interesting sequence. Overall, the proposed procedure show substantial reduction in the total number of rules which eventually turns into a few interesting rules compared to the standard procedures. This methodology is very useful in a situation where the domain knowledge is not easily reachable. Further work can be done on seeing the veracity of these measures in different contexts. This idea can easily be extended further to the association mining and also with more BNs.

## References

1. Agrawal, R., Imielinski, T., and Swami, A: Mining association rules between sets of items in large databases. In Proceedings of the ACM SIGMOD Conference on Management of Data, Washington D.C. (1993) 207-216
2. Agrawal, R. and Srikant, R: Mining sequential patterns. In Proceedings of the 20<sup>th</sup> Int'l Conference on Very Large Databases, Santiago, Chile (1994)

3. Srikant, R. and Agrawal, R. Mining sequential patterns:generalisations and performance improvements. IBM Research Report RJ9994 (1995)
4. Joshi, M., Karypis, G. and Vipin Kumar: A universal formulation of sequential patterns. In Proceedings of the KDD'2001 Workshop on Temporal Data Mining, San Fransisco, August (2001)
5. Spiliopoulou, Myra (1999): Managing Interesting Rules in Sequence Mining, PKDD99, Prag, Czech Republic (1999)
6. Susan Jensen: Mining medical data for predictive and sequential patterns: PKDD'01, Freiburg, Germany (2001)
7. Huang Xiangji, Aijun An, Nick Cercone: Comparison of Interestingness Functions for Learning Web Usage Patterns, CIKM'02, November 4-9, McLean, Virginia, USA (2002)
8. Xuan-Hiep Huynh, Fabrice Guillet, and Henri Briand: An exploratory analysis tool for interestingness measures, International Conference on Applied Stochastic Models and Data Analysis, ASMDA, Brest, France (2005)
9. Garofalakis, M., Rastogi, R. and Shim, K. Spirit: Sequential pattern mining with regular expression constraints. IEEE Transactions on knowledge and data engineering 14 (2002) 530-552
10. Ahola, J: Mining sequential patterns. Research Report TTE1-2001-10, VTT Information Technology, Finland (2001)
11. Show-Jane Yen and Yue-Shi Lee: An efficient data mining technique for discovering interesting sequential patterns. Proceedings of the 2001 IEEE International Conference on Data Mining (ICDM'01) San Jose, USA (2001) 663-664
12. Alex A. Freitas: On rule interestingness measures, Knowledge-Based Systems 12(1999) 309-315
13. Xiangji Huang, Aijun An, and Nick Cercone: Comparison of interestingness functions for learning web usage patterns, CIKM, McLean, Virginia, USA (2002)
14. Gemma Casas-Garriga: Statistical strategies for pruning all the uninteresting association rules, European Conference on Artificial Intelligence, Valencia, Spain (2004)
15. Roberto J. Bayardo Jr. and Rakesh Agrawal: Mining the most interesting rules, Proc. Of the fifth ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining, (1999) 145-154
16. Yen-Liang Chen, Shih-Sheng Chen, Ping-Yu Hsu: Mining hybrid sequential patterns and sequential rules, Information Systems 27(2002) 345-362
17. Baumgarten, M, Buchner, A.G., and Hughes, J.G: User-driven ranking for measuring the interestingness of knowledge patterns, IKE 2003. 122-128
18. Robert J. Hilderman and Howard J. Hamilton: Measuring the interestingness of discovered knowledge: a principled approach, Intelligent Data Analysis 7(2003) 347-381
19. Jensen, Finn V. Bayesian networks and decision graphs, Springer-Verlag (2001)

# Mining Multi-dimensional Frequent Patterns Without Data Cube Construction\*

Chuan Li<sup>1</sup>, Changjie Tang<sup>1</sup>, Zhonghua Yu<sup>1</sup>, Yintian Liu<sup>1</sup>, Tianqing Zhang<sup>1</sup>,  
Qihong Liu<sup>1</sup>, Mingfang Zhu<sup>1</sup>, and Yongguang Jiang<sup>2</sup>

<sup>1</sup> The Data Base and Knowledge Engineering Lab (DBKE)

Computer School of Sichuan University

{lichuan, tangchangjie}@cs.scu.edu.cn

<sup>2</sup> Chengdu University of Traditional Chinese Medicine

cdtcm@163.com

**Abstract.** Existing approaches for multi-dimensional frequent patterns mining rely on the construction of data cube. Since the space of a data cube grows explosively as dimensionality or cardinality grows, it is too costly to materialize a full data cube, esp. when dimensionality or cardinality is large. In this paper, an efficient method is proposed to mine multi-dimensional frequent patterns without data cube construction. The main contributions include: (1) formally proposing the concept of multi-dimensional frequent pattern and its pruning strategy based on Extended Apriori Property, (2) proposing a novel structure called Multi-dimensional Index Tree (MDIT) and a MDIT-based multi-dimensional frequent patterns mining method (MDIT-Mining), and (3) conducting extensive experiments which show that the space consuming of MDIT is more than 4 orders of multitudes smaller than that of data cube along with the increasing of dimensionality or cardinality at most cases.

## 1 Introduction

Multi-dimensional frequent patterns mining is the essential step in multi-dimensional association rules mining and has wide applications [1, 2]. Let's see a motivating example. A store manager may ask a question like "what groups of customers would like to buy what groups of items ?" What he wants is the patterns like the following:

{*Pricai Proceedings, DBMiner*}  $\wedge$  {*Occupation('Student')*, *Major('Computer')*} : 2%  
{*Hamburg, Egg, Edible Oil*}  $\wedge$  {*Occupation('House Wife')*, *Income('Middle')*} : 4%

---

\* This work was supported by National Science Foundation of China (60473071), Specialized Research Fund for Doctoral Program by the Ministry of Education (SRFDP 20020610007), the grant from the State Administration of Traditional Chinese Medicine (SATCM 2003JP40) and National Science Foundation of China (90409007). CHUAN LI, TIANQING ZHANG, YINTIAN LIU, QIHONG LIU, and MINGFANG ZHU are Ph. D Candidates at DB&KE Lab, Sichuan University. YU ZHONGHUA is a professor at Sichuan University. JIANG YONGGUANG is a professor at Chengdu University of TCM. And TANG CHANGJIE is the associate author.

These patterns are useful as they are helpful to analyze the structure of the purchasing power and the major interests of different customer groups [1]. Such patterns are called multi-dimensional frequent patterns as they contain not only items but also the multi-dimensional values combination and they are frequent. Regretfully, this concept has never been formally proposed. And it is only deemed as an intermediate step in multi-dimensional association rules mining [2]. Almost all existing approaches for multi-dimensional frequent patterns mining rely on the construction of data cube [2, 3, 7, 8, 9].

Unfortunately, the storage space of data cube grows explosively as dimensionality or cardinality (i.e. the number of distinct values in a dimension) grows. For example, a data cube of 6 dimensions, cardinality of 1000 (i.e. each dimension having 1000 distinct values), may contain as many as  $(1000)^6$  cube cells. Although adoption of partial materialization (such as iceberg cube, condensed cube, etc.) can delay the growth, it does not solve the fundamental problem [7, 8, 9, 10]. Some may suggest the introduction of concept hierarchies and concept generalization techniques such as AOI [11, 12], attribute reduction [12], etc. to simplify the relation and hence alleviate the disaster caused by large cardinality. However, there are applications with large cardinality dimensions, which have no hierarchies and are too important to be deleted, such as the analysis of Traditional Chinese Medicine Prescriptions [13].

In this paper, an efficient method is proposed to mine multi-dimensional frequent patterns without data cube construction. Our main contributions include: (1) formally proposing the concept of multi-dimensional frequent pattern and its pruning strategy based on Extended Apriori Property, (2) proposing a novel structure called Multi-dimensional Index Tree (MDIT) and a MDIT-based multi-dimensional frequent patterns mining method (MDIT-Mining), and (3) conducting extensive experiments, which show the space consuming of MDIT is more than 4 orders of multitudes smaller than that of data cube along with the increasing of dimensionality or cardinality at most cases.

## 2 Formal Definition of Multi-dimensional Frequent Pattern

Suppose there are  $D$  dimensions,  $\{P_1, P_2, \dots, P_D\}$ . Let  $I$  be a set of items, and  $p_i$  be a possible value of the  $i_{th}$  dimension. We have the following formal definitions.

**Definition 2.1 (Multi-dimensional Pattern).** Let  $i, i_1, i_2, \dots, i_k$  be items of  $I$ ,  $l_1, l_2, \dots, l_j$  be distinct integers between 1 and  $D$ .  $i \wedge \{p_{i1}, p_{i2}, \dots, p_{ij}\}$  is called a **Multi-dimensional Item** and  $\{i_1, i_2, \dots, i_k\} \wedge \{p_{i1}, p_{i2}, \dots, p_{ij}\}$  is called a **Multi-dimensional Pattern**, where  $\{p_{i1}, p_{i2}, \dots, p_{ij}\}$  is called a **Multi-dimensional Values Combination**.

**Definition 2.2 (The Universal Set).** Let  $l_1, l_2, \dots, l_j$  be distinct integers between 1 and  $D$ . The union of all distinct multi-dimensional patterns of the form  $I \wedge \{p_{i1}, p_{i2}, \dots, p_{ij}\}$  is called the **Universal Set** of the multi-dimensional space, denoted as **MI**,  

$$MI = \bigcup_{\text{possible } l_1, l_2, \dots, l_j} (I \wedge \{p_{i1}, p_{i2}, \dots, p_{ij}\}).$$

**Definition 2.3 (Sub-pattern Relationship).** Let  $MA = \{i_1, i_2, \dots, i_j\} \wedge \{p_{11}, p_{12}, \dots, p_{ij}\}$ ,  $MB = \{i_1', i_2', \dots, i_k'\} \wedge \{p_{11'}, p_{12'}, \dots, p_{ij'}\}$  be two multi-dimensional patterns.  $MA$  is called a **Sub-pattern of  $MB$** , denoted as  $MA \subseteq MB$ , if the following two criteria hold simultaneously:

- (1)  $\{i_1, i_2, \dots, i_j\} \subseteq \{i_1', i_2', \dots, i_k'\}$ , and
- (2) For any element  $p_{l1} \in \{p_{11}, p_{12}, \dots, p_{ij}\}$ , there exists an element  $p_{l'} \in \{p_{11'}, p_{12'}, \dots, p_{ij'}\}$ <sup>1</sup> such that (a)  $p_l$  and  $p_{l'}$  are values of the same dimension, and (b)  $p_l = p_{l'}$ .

A pattern  $MA = \{i_1, i_2, \dots, i_k\} \wedge \{p_{11}, p_{12}, \dots, p_{ij}\}$  is called a **Sub-pattern of  $MI$** , denoted as  $MA \subseteq MI$ , if one of the following two criteria holds:

- (1)  $MA$  is of the form  $I \wedge \{p_{11}, p_{12}, \dots, p_{ij}\}$  or
- (2) There exists an multi-dimensional pattern,  $MB = I \wedge \{p_{11}, p_{12}, \dots, p_{ij}\}$  satisfying the following two criteria simultaneously:
  - (a)  $MB \subseteq MI$ , and
  - (b)  $MA \subseteq MB$

Based on Definition 2.2 and 2.3 it's clear that for any multi-dimensional pattern  $MA$ ,  $MA \subseteq MI$ .

**Definition 2.4 (Multi-dimensional Transaction Database).** Given the universal set  $MI$ , a set of multi-dimensional patterns  $\{mt_1, mt_2, \dots, mt_n\}$  is called a **Multi-dimensional Transaction Database**, denoted as  $MTDB$ , if for any  $i \in [1..n]$ ,  $mt_i \subseteq MI$ . The support  $\eta$  (or occurrence frequency) of a multi-dimensional values combination  $\{p_{11}, p_{12}, \dots, p_{ij}\}$  is the number of transactions containing  $\{p_{11}, p_{12}, \dots, p_{ij}\}$  in  $MTDB$ . The support  $\eta$  (or occurrence frequency) of a multi-dimensional pattern  $MA$  is the number of transactions containing  $MA$  in  $MTDB$ .

**Definition 2.5 (Multi-dimensional Frequent Pattern).** Let  $\eta$  be the predefined minimum support threshold for the given  $MTDB$ . A multi-dimensional pattern is a **Multi-dimensional Frequent Pattern** if its support is no less than  $\eta$ . The problem of multi-dimensional frequent patterns mining is to generate the complete set of multi-dimensional frequent patterns from the multi-dimensional database  $MTDB$ .

### 3 Pruning Strategy

Based on Definition 2.4 and 2.5, each multi-dimensional pattern should be investigated as to judge whether it is frequent. However, for any given multi-dimensional pattern with  $J$  dimensions and  $K$  items, the possible non-empty sub-patterns are as many as  $(2^J - 1) * (2^K - 1)$ . The search space is too huge to be handled. Thus, optimization is in great need. Inspired by the single dimensional case, we consider extending the *Apriori* property [5] to multi-dimensional landscape and using it to reduce the search space. Then we have the following lemmas.

---

<sup>1</sup> Just having  $\{p_{11}, p_{12}, \dots, p_{ij}\} \subseteq \{p_{11'}, p_{12'}, \dots, p_{ij'}\}$  is incorrect because it may not satisfy (a).

**Lemma 3.1 (Extended Apriori Property).** Let  $MA$ ,  $MB$  be two multi-dimensional patterns,  $MA \subseteq MB$ . If  $MA$  is not a multi-dimensional frequent pattern,  $MB$  is not a multi-dimensional frequent pattern, either.

**Lemma 3.2 (Pruning Strategy).** Multi-dimensional patterns containing infrequent multi-dimensional values combination cannot be frequent.

**Observation 3.1.** The complete set of multi-dimensional frequent patterns could be partitioned into a series of non-overlapped sets such that patterns in each set share the same frequent multi-dimensional values combination.

Based on Lemma 3.2 and Observation 3.1, the multi-dimensional frequent patterns mining process can be fulfilled in the following summarized steps:

- (1) Discover all the multi-dimensional values combinations,  $\{p_{11}, p_{12}, \dots, p_{ij}\}$ , whose support is above the minimum support threshold,  $\eta$ .
- (2) For each multi-dimensional values combination,  $\{p_{11}, p_{12}, \dots, p_{ij}\}$ , track down all the multi-dimensional transactions containing it, which forms a projected database of  $MTDB$ .
- (3) Mine single-dimensional frequent patterns from the projected databases and output the patterns in conjunction with their corresponding multi-dimensional values combinations.

## 4 The Introduction of MDIT: Design, Construction and Analysis

Stemming from the above discussion, we propose a new approach, called MDIT, and two new algorithms: one for computing MDIT, and one for multi-dimensional frequent patterns mining based on MDIT. This new approach will be able to handle databases of reasonable dimensionality and extremely high cardinality. The general idea is to partition the base dataset into a series of projected databases according to different multi-dimensional values combinations. With MDIT, one can search the corresponding sets of transaction IDs ( $TIDs$ ) according to frequent multi-dimension values combinations. After that, one can mine frequent patterns from each projected database (represented as a  $TID$  list). Finally, after assembled with multi-dimensional values combinations, one can get the complete set of multi-dimensional frequent patterns. The whole process is accomplished highly efficient with MDIT.

### 4.1 Multi-dimensional Index Tree (MDIT)

**Definition 4.1.** A Inverted Index Item is a 2-tuple  $R = \{I, P\}$ , where  $I$  denotes a dimension value and  $P$  denotes a pointer leading to a child node.

**Definition 4.2. Multi-dimensional Index Tree (MDIT)** is a tree-like structure defined as follows:

- (1) **MDIT** consists of a root node, a set of sub-trees, a set of transaction buckets and a global  $MTDB$  array.
- (2) The **Root** node contains a set of inverted index items.
- (3) Each node in the **Sub-trees** contains a set of inverted index items.

- (4) A **Transaction Bucket** is a tuple  $B = \{S, TIDs\}$ , where frequency field  $S$  denotes the support of the bucket,  $TIDs$  denotes a  $TID$  list.
- (5) A **Global MTDB Array** is a global array  $A = \{t_1, t_2, \dots, t_m\}$ , where  $t_1, t_2, \dots, t_m$  are the itemset part of transactions of the  $MTDB$ .

**Example 4.1.** We roughly sketch the MDIT construction process for the  $MTDB$  given in Example 4.1. Initially, MDIT is an empty root node and empty global array. We sequentially process all the transactions. For each transaction, we investigate its dimension values one by one, and after that, we store its  $TID$  and itemset part into the global  $MTDB$  array. See transaction 1. The first dimension value is  $A1$ , thus we add  $A1$  into root and generate a path  $Root \rightarrow D \rightarrow E \rightarrow F$  to memorize the inverted index  $\{A1, Any, Any: 1:1\}$ <sup>2</sup>. No further updates are made to MDIT by transaction 1 since the rest dimension values are “Any”. Then we store  $\{1: I001, I002\}$  into the global  $MTDB$  array. The following transactions are processed until transaction 7. The first dimension value is  $A1$ , so we update inverted index  $\{A1, Any, Any: 1:1\}$  into  $\{A1, Any, Any: 2: 1, 7\}$  by modifying transaction bucket  $F$ . The second dimension value is  $B1$ . We add  $B1$  into node  $D$  and generate a path  $B1 \rightarrow A \rightarrow B$  to store inverted index  $\{A1, B1, Any: 1:7\}$ . What's more, we need to update inverted index in transaction bucket  $C$  by registering 7 into its  $TIDs$  field and adding to its frequency field. The third dimension value is “Any”, so we store  $\{7: I023, I025\}$  into the global  $MTDB$  array. In this way, after all 11 transactions are processed, we got the MDIT shown in Figure 1.

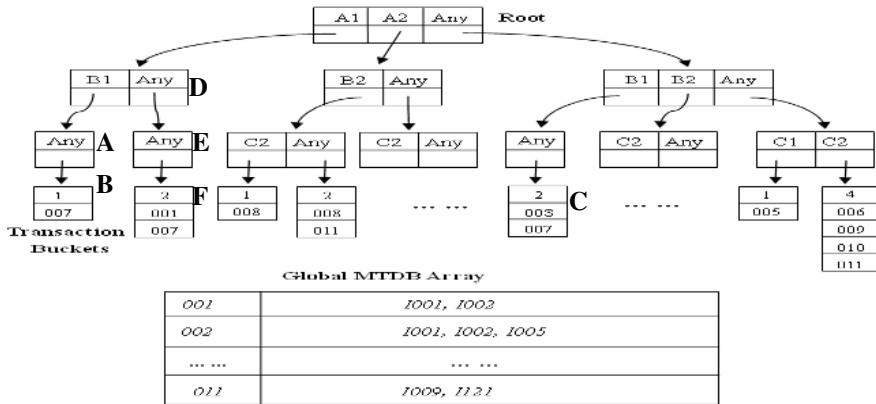


Fig. 1. Multi-dimensional Index Tree (MDIT)

## 4.2 Construction Algorithm of MDIT

Based on above illustration, the MDIT building algorithm can be summarized below.

**Algorithm 1 (Build-MDIT).** Construction of MDIT

**Input:** A multi-dimensional database  $MTDB = \{mt_1, mt_2, \dots, mt_m\}$  with  $n$  dimensions ( $P_1, P_2, \dots, P_n$ ).

<sup>2</sup> The inverted is represented as {multi-dimensional values combination: frequency:TID list}.

**Output:** A MDIT consisting of: (1) A MDIT tree<sup>3</sup> and (2) A global *MTDB* array

**Method:**

- (1) Initialize the MDIT with *root* and an empty global *MTDB* array;
- (2) **for** each  $mt_i \in MTDB$
- (3)   **for** each dimension value  $p_j$  of dimension  $P_j$  in  $mt_i$  sequentially
- (4)     **if** there is no  $p_j$ 's multi-dimensional index item in level  $j$
- (5)       Add  $p_j$ 's multi-dimensional index items into level  $j$ ;
- (6)       Register the *TID* of  $mt_i$  into corresponding set of transaction buckets associated with  $p_j$ ,  $BS = \{B_1, B_2, \dots, B_k\}$ ;
- (7)       **for** each bucket  $B \in BS$ ,  $B \rightarrow S++$  ;
- (8)     **else**
- (9)       Register the *TID* of  $mt_i$  into corresponding set of transaction buckets associated with  $p_j$ ,  $BS' = \{B_1, B_2, \dots, B_k\}$ ;
- (10)      **for** each bucket  $B \in BS'$ ,  $B \rightarrow S++$  ;
- (11)     Add the *TID* and the itemset part of  $mt_i$  into the global *MTDB* array;

## 5 Algorithm MDIT-Mining

### 5.1 Algorithm for Multi-dimensional Frequent Patterns Mining

Given the MDIT, one can mine multi-dimensional frequent patterns directly. Suppose the user wants to find all multi-dimensional frequent patterns from MDIT shown in Figure 1. Assume the minimum support threshold,  $\eta$  to be 2. Just as what we did in case of inverted indices, firstly we search the MDIT to find all the buckets whose support is equal to or above 2. These buckets are  $\{1, 7: 2\}^4$ ,  $\{8, 11: 2\}, \dots, \{6, 9, 10, 11: 4\}$ . Secondly, for each frequent bucket, we get its projected database via its *TID list* and mine frequent patterns wherfrom. Finally, we output them in conjunction with their corresponding multi-dimensional values combinations.

**Example 5.1.** For the transaction bucket  $\{003, 007: 2\}$  (*C* in Figure 1), transaction 3 and 7 constitute a projected database as shown in *Table 1*, and they share the same frequent multi-dimension values combination  $\{Any, B1, Any\}$ .

**Table 1.** Projected database represented by *TID* 3 and 7

<i>TID</i>	<i>Items Sold</i>
003	I023,I334
007	I023,I025

Then we mine frequent patterns from this projected database and get frequent pattern  $\{I023: 2\}$ <sup>5</sup>. Concatenated with its multi-dimensional values combination, we

<sup>3</sup> For convenience, we call the tree part of MDIT (consisting of root, sub-trees, and transaction buckets) a MDIT tree.

<sup>4</sup> The format is  $\{\text{TID list}: \text{frequency}\}$ .

<sup>5</sup> The format of frequent pattern is  $\{\text{Itemset}: \text{frequency}\}$ .

get a multi-dimensional frequent pattern  $\{I023\} \wedge \{\text{Any}, B1, \text{Any}\}$  whose support is 2. In this way, we can get all the multi-dimensional frequent patterns in *Table 2*.

**Table 2.** The resultant multi-dimensional frequent patterns

Multi-dimensional frequent patterns	Support
$\{I023\} \wedge \{\text{Any}, \text{Any}, C2\}$	2
$\{I025\} \wedge \{\text{Any}, \text{Any}, C2\}$	2
$\{I023\} \wedge \{\text{Any}, B1, \text{Any}\}$	2

This leads to our algorithm for multi-dimensional frequent patterns mining as below.

**Algorithm 2 (MDIT-Mining).** Mining multi-dimensional patterns based on MDIT

**Input:** MDIT and the minimum support threshold,  $\eta$ .

**Output:** The complete set of multi-dimensional frequent patterns, where each pattern consists of: (1) the multi-dimensional values combination and (2) a set of items.

**Method:**

- (1) **for** each transaction bucket  $B\{$
- (2)   **if** ( $B->S \geq \eta$ ) {
  - //get the multi-dimensional values combination
  - (3)    $Mdvc$ =the set of dimensional values of the inverted indices from *root* to  $B$ ;
  - //mine the frequent patterns in the projected database
  - (4)   Construct *FP-Tree* with the projected database represented by  $B->TIDs$ ;
  - (5)    $FP=FP\_Growth(FP-Tree, \eta)$ ;
  - (6)   **for** each pattern  $p \square FP$  output  $p$  in conjunction with  $Mdvc$ .
  - (7)   }
  - (8) }

## 6 Performance Study

This section reports the performance study of the two algorithms: Build-MDIT and MDIT-Mining in comparison with data cube method on various datasets. All theses algorithms are implemented in *Microsoft Visual C++ 6.0*. Experiments are performed on a  $1.6GHz$  AMD PC with  $448MB$  main memory, running *Microsoft Windows XP Professional*. All algorithms together with data generator are available through address [14]. Regarding the data cube method, we develop a simulating algorithm to estimate its running costs on basis of paper [2]. We have implemented various simplifications to reduce the estimation of both its time and space consuming including:

- (1) Ignoring the space consumed by all the other cuboids generated except the base one. i.e. For the data cube with dimensionality of  $D$ , among the  $(2^D - 1)$  cuboids, we only consider the base cuboid and ignore all the other  $(2^D - 2)$  cuboids.

- (2) Supposing each cube cell only takes the space of an integer (which should actually store a *TID list*).  
(3) Ignoring the space cost by the multi-dimensional database.  
As a notational convention, the following symbols are adopted.

**Table 3.** Symbols

<i>Symbol</i>	<i>Meaning</i>	<i>Symbol</i>	<i>Meaning</i>
<i>D</i>	<i>Number of dimensions</i>	<i>C</i>	<i>Cardinality of each dimension</i>
<i>T</i>	<i>Number of transactions</i>	<i>M</i>	<i>Max_item in the MTDB</i>
<i>L</i>	<i>Maximum transaction length</i>	<i>S</i>	<i>Minimum support</i>

## 6.1 Space Scalability of MDIT

Figure 2 shows how the storage size of MDIT scales as dimensionality increases in comparison with data cube. The experiments were conducted on dataset (10000-T). The storage space of MDIT only approximately doubles as dimensionality increases by 1. However, the storage space of data cube would increase about  $(C+1)=100$  times as dimensionality increases by 1 (Note that time axis is in logarithmic scale).

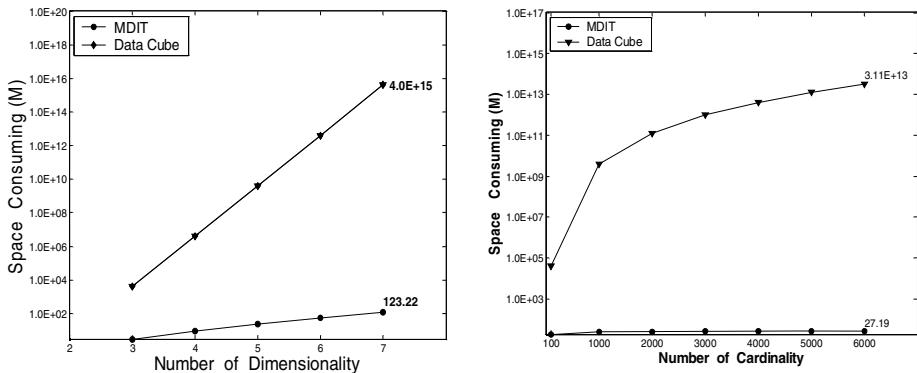
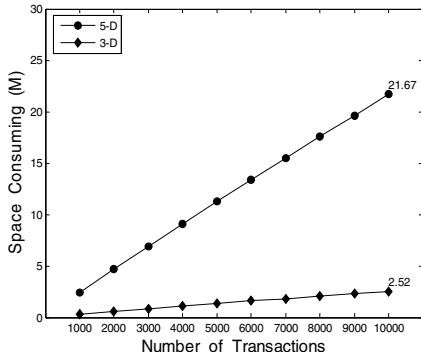
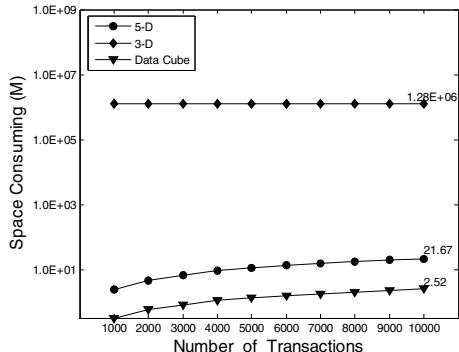
**Fig. 2.** Storage size of MDIT vs. data cube (10000-T)  $T=10000$ ,  $C=1000$ ,  $M=10000$ ,  $L=20$ **Fig. 3.** Storage size of MDIT vs. data cube (5-D)  $D=5$ ,  $T=5000$ ,  $M=5000$ ,  $L=10$ 

Figure 3 demonstrates how the cardinality affects the space growth of MDIT and that of data cube on dataset (5-D). The space of data cube is  $S=O((C+1)^D)$ [1, 2]. Although it does not grow exponentially along with number of cardinality, the growth rate actually is greater and greater as cardinality increases ( $S'=O(D(C+1)^{D-1})$ ), which ultimately leads to even worse space explosion. However, the space of MDIT

almost keeps unchanged with cardinality<sup>6</sup>. When cardinality reaches 6000, the storage of MDIT remains under 30 MB, while the space of data cube has reached 12 orders of multitudes that of MDIT and surpassed the capacity summation of all existing storage devices. Overall, space consuming of MDIT is at least 4 orders of multitudes smaller than that of data cube. In addition, this gap grows wider as dimensionality or cardinality increases.



**Fig. 4.** Storage size of MDIT: (5-D)  $D=5$ ,  $C=200$ ,  $M=10000$ ,  $L=20$ . (3-D)  $D=3$ ,  $C=200$ ,  $M=5000$ ,  $L=10$



**Fig. 5.** Storage size of MDIT vs. Data cube: (MDIT 5-D)  $D=5$ ,  $C=200$ ,  $M=10000$ ,  $L=20$ . (3-D)  $D=3$ ,  $C=200$ ,  $M=5000$ ,  $L=10$ . (Data cube 5-D)  $D=5$ ,  $C=200$

Fig. 4 and Fig. 5 show the space scalability of MDIT along with number of transactions. Our experiments were conducted on two different datasets (5-D) and (3-D). From Figure 4 we can see that storage space grows linearly as number of transactions grows. This is expected because each transaction would at most increase a certain amount of storage,  $(2^D - 1) * (\text{sizeof}(tid) + \text{sizeof}(I) + \text{sizeof}(P)) = 12 * (2^D)$ , as explored in space complexity analysis. In Figure 5, we compare the space consuming of MDIT with that of complete data cube. In the range transactions number varies from 1000 to 10000, the space of MDIT is less than 22MB, while the storage of data cube reaches  $1.28 \times 10^6$  MB, more than  $10^4$  times that of MDIT.

## 7 Conclusion

In this paper, an efficient method is proposed to mine multi-dimensional frequent patterns without data cube construction. Our contributions are (1) proposing the concept of multi-dimensional frequent pattern, (2) proposing a novel MDIT structure and a MDIT-based multi-dimensional frequent patterns mining method (MDIT-Mining), and (3) conducting extensive experiments that show space of MDIT is more than 4 orders of multitudes smaller than data cube along with the increase of dimensionality or cardinality.

<sup>6</sup> There is slight growth because larger cardinality causes lower sharing rate of multi-dimensional index items, and hence more multi-dimensional index items are generated.

## References

1. Data Mining: Concepts and Techniques. Jiawei Han, Micheline Chamber, Morgan Kaufmann, Hardcover, ISBN 1558604898
2. Kamber, J. Han, and J.Y. Chiang. Metarule-guided mining of multi-dimensional association rules using data cubes. KKD'97.
3. R. Baeza-Yates. et. al Modern Information Retrieval. Addison-Wesley, 1999.
4. G. Piatetski-Shapiro. Discovery, analysis, and presentation of strong rules. In Knowledge Discovery in Databases, pages 229-248, 1991.
5. Rakesh Agrawal, et.al Mining association rules between sets of items in large databases. In Proc. of the ACM SIGMOD 1993
6. Han, J., Pei, J., and Yin, Y. Mining Frequent Patterns without Candidate Generation. SIGMOD, 1-12. 2000.
7. D. Barbara and M. Sullivan. Quasi-cubes: Exploiting approximation in multidimensional databases. SIGMOD Record, 26:12-17, 1997.
8. S. Agarwal, et. al. On the computation of multidimensional aggregates. In Proc. 22nd VLDB, pages 506--521, Mumbai, Sept. 1996.
9. Y. Sismanis and N. Roussopoulos. The dwarf data cube eliminates the high dimensionality curse. TR-CS4552, University of Maryland, 2003.
10. Xiaolei Li, Jiawei Han, and Hector Gonzalez. High-Dimensional OLAP: A Minimal Cubing Approach. In VLDB'04.
11. J. Han, Y. Cai, and N. Cercone. Knowledge discovery in databases: An attribute-oriented approach. VLDB 95, pages 547--559.
12. Y.D. Cai, N. Cercone, and J. Han. *Attribute-oriented induction in relational databases*. In Knowledge Discovery in Databases, 1991.
13. Peng Huaiyu. The First Volume of Great Formula Dictionary of TCM. People's Medical Publishing House. December 1993
14. <http://cs.scu.edu.cn/~lichuan/lichuan.rar>

# A New Approach to Symbolic Classification Rule Extraction Based on SVM

Dexian Zhang<sup>1</sup>, Tiejun Yang<sup>1</sup>, Ziqiang Wang<sup>1</sup>, and Yanfeng Fan<sup>2</sup>

<sup>1</sup> School of Information Science and Engineering, Henan University of Technology,  
Zheng Zhou 450052, P.R.C (China)  
[zdx@haut.edu.cn](mailto:zdx@haut.edu.cn)

<sup>2</sup> Computer College, Northwestern Polytechnical University,  
Xi'an 710072, P.R.C (China)

**Abstract.** There still exist two key problems required to be solved in the classification rule extraction, i.e. how to select attributes and discretize continuous attributes effectively. The lack of efficient heuristic information is the fundamental reason that affects the performance of currently used approaches. In this paper, a new measure for determining the importance level of the attributes based on the trained SVM is proposed, which is suitable for both continuous attributes and discrete attributes. Based on this new measure, a new approach for rule extraction from trained SVM and classification problems with continuous attributes is proposed. The performance of the new approach is demonstrated by several computing cases. The experimental results prove that the approach proposed can improve the validity of the extracted rules remarkably compared with other rule extracting approaches, especially for the complicated classification problems.

## 1 Introduction

One of the data mining problems is classification. Classification plays a very important role in many fields of applications. Classification is the process of finding the common properties among different patterns and classifying them into classes. The results are often expressed in the form of symbolic rules—the classification rules. By applying the rules, patterns can be easily classified into different classes they belong to. The classification rule extraction has become an important aspect of data mining. The existing approaches for extracting the classification rules can be roughly classified into two categories, data driven approaches and model driven approaches. The main characteristic of the data driven approaches is to extract the symbolic rules completely based on the treatment with the sample data. The main characteristic of the model driven approaches is to establish a model at first through the sample set, and then extract rules based on the relation between inputs and outputs represented by the model. Theoretically, these rule extraction approaches can overcome the shortcomings of data driven approaches mentioned above. Therefore, the model driven approaches will be the promising ones for rules extraction. The representative approaches are rules

extraction approaches based on neural networks. Existing methods of rules extraction based on neural networks can roughly be divided into two kinds, one kind is the methods based on the analysis of network structure [1-4]. The other kind is the methods based on the relation conversion of inputs and outputs [5-8]. Though these methods have certain effectiveness for rules extraction, there exist still some problems, such as low efficiency and validity, and difficulty in dealing with continuous attributes etc.

There are two key problems required to be solved in the classification rule extraction, i.e. the attribute selection and the discretization to continuous attributes. Attribute selection is to select the best subset of attributes out of original set. The attributes that are important to maintain the concepts in the original data are selected from the entire attributes set. How to determine the importance level of attributes is the key to attribute selection. Mutual information based attribute selection [9-10] is a common method of attribute selection, in which the information content of each attribute is evaluated with regard to class labels and other attributes. By calculating mutual information, the importance levels of attributes are ranked based on their ability to maximize the evaluation formula. Another attribute selection method uses entropy measure to evaluate the relative importance of attributes [11]. The entropy measure is based on the similarities of different instances without considering the class labels. In paper [12], the separability-correlation measure is proposed for determining the importance of the original attributes. The measure includes two parts, the intra-class distance to inter-class distance ratio and an attributes-class correlation measure. Through attributes-class correlation measure, the correlation between the changes in attributes and their corresponding changes in class labels are taken into account when ranking the importance of attributes. The attribute selection methods mentioned above can be classified into the sample driven method. Their performance depends on the numbers and distributions of samples heavily. It is also difficult to use them to deal with continuous attributes. Therefore, it is still required to find more effective heuristic information for the attribute selection and the discretization to continuous attribute in the classification rule extraction.

The position and shape characteristics of the classification hypersurface are effective and direct heuristic information for the attribute selection and the discretization to continuous attribute in the classification rule extraction. In ID3 algorithm, the heuristic information for constructing decision tree is the information content of each attribute for classification, which reflects the proportion changes of classes with the change of attribute values, and also reflects the shape changes of classification hypersurface with the changes of attribute values. But the reflection to the shape changes of classification hypersurface is obviously indirect and excessively dependent on the sample distributions and numbers, which cause many problems such as preferring to select the attributes with many values in the process of decision tree construction. According to the analysis above, the effective heuristic information for the attribute selection and the discretization to continuous attribute in the classification rule extraction can be derived from the position and shape characteristics of classification hypersurface.

SVM (Support vector machine) is a new technique for data classification based on statistic study theory. It can find the optimal separating hypersurface with the maximal margin to divide the sample set. Based on the theory of minimizing the structure risk, it can have better generalization even for small sample sets. In addition to these advantages, it has high training speed contrast with neural networks. Therefore in this paper, we use trained SVM to obtain the position and shape characteristics of the classification hypersurface. Based on the analysis of the relations among the position and shape characteristics of classification hypersurface, the partial derivative distribution of the outputs of trained SVM to its corresponding inputs and the importance level of attributes to classifications, this paper mainly studies on the measure method of the classification power of attributes on the basis of differential information of the trained SVM and develops new approach for the rule extraction.

## 2 Measure for Attribute Importance Ranking

Without losing the universality, next we will discuss the classification problems with two attributes and two class labels.

For a 2-dimension classification problem, assuming the shape of classification hypersurface in the given area  $\Omega$  is as shown in Fig.1, in which the level axis is attribute  $x_1$  and the perpendicular axis is attribute  $x_2$ . In the cases (a) and (b), the classification power of attribute  $x_1$  is stronger, so area should be divided via attribute  $x_1$ . In the case (c), attribute  $x_1$  and attribute  $x_2$  have the equal classification powers. Therefore, for a given attribute value space  $\Omega$ , the importance level of each attribute depends on the mean perpendicular degree between each attribute axis and classification hypersurface in space  $\Omega$  or its adjacent space. The higher is the mean perpendicular degree, the higher is the importance level.

Assuming the attribute vector of classification problems is  $x = [X_1, X_2, \dots, X_n]$ , where  $n$  is the number of attributes, and the corresponding classification label is  $y$  and  $y \in \{1, -1\}$ , then the sample of classification problems can be represented as  $\langle X, y \rangle$ . For the convenience of computing and comparing, quantification and normalization of the attribute values and classification values will be carried out as follows.

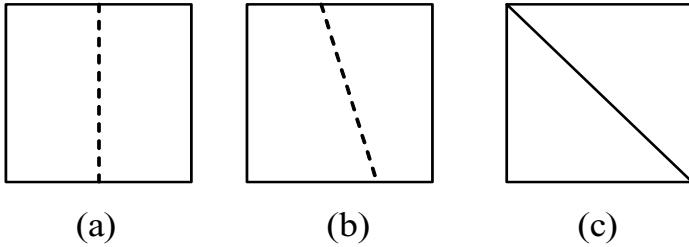
The quantification is performed for the values of discrete attributes. In this paper, values of discrete attributes are quantified as integer in some order, i.e. 0, 1, 2, 3, ...

The normalization is performed to adjust the of SVM input ranges. For a given attribute value space  $\Omega$ , utilizing the following linear transformation to map the attribute value  $X$  of sample to the SVM input  $x$ , making every elements in the  $x$  with the same range of  $[\Delta, -\Delta]$ .

$$x = bX + b_0 \quad (1)$$

where  $b = (b_{ij})$  is a transformation coefficient matrix. Here

$$b_{ij} = \begin{cases} \frac{2\Delta}{MaxX_i - MinX_i} : j = i \\ 0 : otherwise \end{cases} \quad (2)$$



**Fig. 1.** Typical shapes of classification hypersurface

$b_0 = (b_{0i})$  is a transformation vector. Here,

$$b_{0i} = \Delta - a_{ii} \text{Max}X_i \quad (3)$$

The parameter  $\Delta$  affects the generalization of trained SVM, in this paper we usually set  $\Delta = 0.5 \sim 2$ .

After quantification and normalization, we can get a sample set of instance-label pairs  $(x_i, y_i), i = 1, \dots, l$ , where  $x_i \in R^n$  and  $y_i \in \{1, -1\}$ . Then the support vector machine trained by solving the following optimization problem.

$$\min_{w,b,\xi} \frac{1}{2} w^T w + C \sum_{i=1}^l \xi_i \quad (4)$$

subject to

$$y_i(w^T \phi(x_i) + b) \geq 1 - \xi_i \text{ and } \xi_i \geq 0 \quad (5)$$

$$\xi_i \geq 0 \quad (6)$$

Here sample vectors  $x_i$  are mapped into a higher dimensional space by the function  $\phi$ . Then SVM finds a linear separating hyperlane with the maximal margin in this higher dimensional space.  $C > 0$  is the penalty parameter of the error item. In this paper we usually let  $C = 10 \sim 10^5$ . Furthermore,  $K(x_i, x_j) = \phi(x_i)^T \phi(x_j)$  is the kernel function. In this paper, we use the following radial basis function (RBF) as kernel function.

$$K(x_i, x_j) = \exp(-\gamma \|(x_i - x_j)^2\|), \gamma > 0 \quad (7)$$

Here,  $\gamma$  is kernel parameter. In this paper we usually let  $\gamma = (0.1 \sim 1)/n$ ,  $n$  is the number of attributes in training sample set. The formula (2) dual is

$$\min_{\alpha} \frac{1}{2} \alpha^T Q \alpha - e^T \alpha \quad (8)$$

$$0 \leq \alpha_i \leq C, i = 1, \dots, l \quad (9)$$

subject to

$$y^T \alpha = 0 \quad (10)$$

Where  $e$  is the vector of all ones,  $Q$  is an  $l$  by  $l$  positive semidefinite matrix. In this paper, the output function of SVM is

$$Y(x) = \sum_{i=1}^l y_i \alpha_i K(x_i, x) + b \quad (11)$$

During the attribute importance ranking, only the attribute value space covered by the sample set should be taken into account. Obviously, when the kernel function of SVM is the radial basis function shown by formula (7), any order derivatives of network output  $Y(x)$  to each SVM input  $x_k$  exist.

**Definition 1.** For a given sample set, the attribute value space  $\Omega$  is defined as follows.

$$\Omega = \{x | Minx_k \leq x_k \leq Maxx_k, k = 1, \dots, n\} \quad (12)$$

Where  $Minx_k$  and  $Maxx_k$  are the minimal and maximal value of  $k$ -th attribute in the given sample set, respectively.

**Definition 2.** For a given attribute value space  $\Gamma$ ,  $\Gamma \subset \Omega$ , the space near the classification hypersurface  $V_\Gamma$  is defined as follows.

$$V_\Gamma = \{x | x \in \Gamma \wedge Y(x) \leq \delta\} \quad (13)$$

Here  $\delta \leq 1$ .

According to discussion to Fig.1, for a given attribute value space  $\Omega$ , the importance level of each attribute depends on the mean perpendicular degree between each attribute axis and classification hypersurface in space  $\Omega$  or its adjacent space. We can represent the perpendicular degree approximately based on the analysis of the geometrical relation between the gradient vectors of the classification hypersurface and the each attribute axis vectors. Based on the perpendicular degree we can establish the measure of the importance level of attributes.

**Definition 3.** For a given trained SVM and the attribute value  $\Gamma$ ,  $\Gamma \subset \Omega$ , the perpendicular level between classification hypersurface and attribute axis  $x_k$  is defined as follows.

$$P_k(x) = \begin{cases} \frac{|\frac{\partial Y(x)}{\partial x_k}|}{\sqrt{\sum_k [(\frac{\partial Y(x)}{\partial x_k})^2 + 1]}} & : x \in V_\Gamma \\ 0 & : otherwise \end{cases} \quad (14)$$

**Definition 4.** For a given attribute value space  $\Gamma$ ,  $\Gamma \subset \Omega$ , the measurement of classification power of attribute  $x_k$  is defined as follows.

$$JP(x_k) = \begin{cases} \frac{\int_{V_\Gamma} P_k(x) dx}{\int_{V_\Gamma} dx} & : \int_{V_\Gamma} dx \neq 0 \\ 0 & : otherwise \end{cases} \quad (15)$$

The importance level measure  $JP(x_k)$  of the attribute  $x_k$  represents the influence degree of attribute  $x_k$  to classification. So in the process of rules extraction, the

value  $JP(x_k)$  is the important instruction information for selecting attributes and dividing attribute value space. The new measure of the attribute importance level proposed in this paper has many advantages compared with ID3 algorithm based on mutual information. It is suitable for measuring the classification power of continuous attributes and discrete attributes. Because of the good prediction ability of trained SVM, it can also overcome the shortcomings of the measure method of ID3 algorithm such as excess dependency on the number and distribution of samples, too much sensitive to the noise etc.

The typical classification problem of weather type for playing golf is employed to demonstrate the performance of the new measure method. The computing results are shown as table 1. The attributes and their values are as follows: Outlook has the value of sunny, overcast and rain, quantified as 0, 1, 2. Temperature has the value of  $64 \sim 83$ . Humidity has the value of  $65 \sim 96$ . Windy has the value of true, false, quantified as 0, 1. The size of the training sample set is 14.

From table 1, in the whole space the measure value of importance level of attribute outlook is the biggest, therefore attribute outlook should be selected as the root node of the decision tree, and the attribute value space should be divided by its values. While in the subspace of outlook= rain, measure value of attribute windy is the biggest. Thus according to this information, the optimal decision tree and corresponding classification rules can be generated.

**Table 1.** Computing Results of Measurement Value  $JP(x_k)$

Attributes	Whole Area	Outlook=sunny	Outlook=rain
Outlook	<u>0.502</u>	–	–
Temperature	0.216	0.397	0.078
Humidity	0.262	<u>0.517</u>	0.313
Windy	0.191	0.080	<u>0.425</u>

### 3 Rules Extraction Method

The rules extraction process from trained SVM proposed in this paper is quite similar to the process of generating decision trees, which includes data pretreatment, SVM training and rules extraction etc. The algorithm for rule extraction from trained SVM proposed includes the rule extraction algorithm and the main algorithm.

The rule extraction algorithm is described as follows.

- 1) Generate a queue  $R$  for finished rules and a queue  $U$  for unfinished rules.
- 2) Select attribute  $x_k$  with the biggest value of  $JP(x_k)$  computed by formula (15) as the extending attribute out of the present attributes. Divide the attribute  $x_k$  into intervals according to the chosen interval number. Then for each interval, pick attribute  $x_j$  with the biggest  $JP(x_k)$  as the extending attribute for each interval. Merge the pairs of adjacent intervals with the same extending attribute and same class label with the largest proportion in all of the class labels. A rule

is generated for each merged interval. If the class error of the generated rule is less than the predefined value, put it into the queue  $R$ , otherwise put it into the queue  $U$ .

3) If  $U$  is empty, the extraction process terminates, otherwise go to 4).

4) Pick an unfinished rule from the queue  $U$  by a certain order, and perform division and mergence. A rule is generated for each merged interval. If the class error of the generated rule is less than the predefined value, then put it into the queue  $R$ , otherwise put it into the queue  $U$ . Go to 3).

The main algorithm is described as follows.

1) Initializing. a) Divide the given sample set into two parts, the training sample set and the test set. By the training sample set, generate the attribute value space  $\Omega$  by formula (12).b) Set the interval number of attributes and the predefined value of error rate.

2) Rule generating. For the class problems with two class labels, train the SVM using the training sample set and generate the rules by the rule extraction algorithm. Otherwise, for each class label, make a copy of the original sample set and set the class labels equal to 1 except the selected class label, generate the rules using the same method as the class problem with two class labels, respectively.

3) Rule Processing. Check the rule number of each class label. Let the rules of the class label with the most rules be default rules.

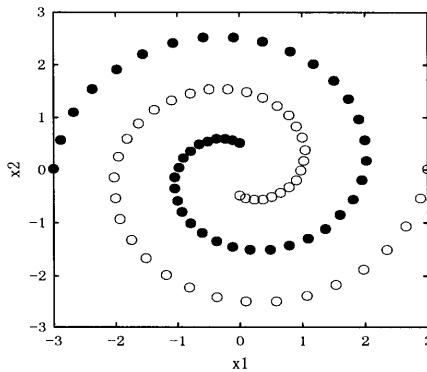
## 4 Experiment and Analysis

The spiral problem [13] and balance scale (balance for short), congressional voting records(voting for short), hepatitis, iris plant(iris for short), statlog australian credit approval(credit-a for short ) in UCI data sets [14] are employed as computing cases, shown in table 2. The attribute value distribution of spiral problem is shown as Fig.2, in which solid points are of Class C0, empty points are of Class C1.

Since no other approaches extracting rules from SVM are available, we include a popular rule learning approach i.e.C4.5R for comparison. The experimental results are tabulated in Table 3. For the spiral problem and the Iris plant problem, the rules set extracted by the new approach are shown in Table 4 and Table 5, respectively. Table 3 shows that the rules extraction results of the new approach

**Table 2.** Computing Cases

	Spiral	Balance	Voting	Hepatitis	Iris	Credit-A
Total Samples	168	625	232	80	150	690
Training Samples	84	157	78	53	50	173
Testing Samples	84	468	154	27	100	517
Classification Numbers	2	3	2	2	3	2
Total Attributes	2	4	16	19	4	15
Discrete Attributes	0	0	16	13	0	9
Continuous Attributes	2	4	0	6	4	6

**Fig. 2.** Samples distribution of spiral problem**Table 3.** Experimental Results Comparison between New Approach(NA) and C4.5R

	#Rules(NA: C4.5R)	Err.Train(NA: C4.5R)	Err.Test(NA: C4.5R)
Spiral	10: 3	0: 38.1%	0: 40.5%
Balance	17: 20	16.5%: 10.8%	15.6%: 18.2%
Voting	3: 4	2.5%: 2.6%	2.5%: 3.2%
Hepatitis	4: 5	3.7%: 3.8%	11.1%: 29.6%
Iris	4: 4	0%: 0%	4%: 10%
Credit-A	3: 3	13.8%: 13.9%	13.9%: 14.9%

**Table 4.** Rules Set of Spiral Problem Generated by the Algorithm Proposed

R1	$x_1 < -1.66 \rightarrow C1$
R2	$x_1[-1.66, -0.71] \wedge x_0 < -1.53 \rightarrow C1$
R3	$x_1[-1.66, -0.71] \wedge x_0 \geq 2.29 \rightarrow C1$
R4	$x_1[-0.71, 0.23] \wedge x_0[-2.5, -1.39] \rightarrow C1$
R5	$x_1[-0.71, 0.23] \wedge x_0[-0.83, 1.39] \rightarrow C1$
R6	$x_1[-0.71, 0.23] \wedge x_0 \geq 2.5 \rightarrow C1$
R7	$x_1[0.23, 0.71] \wedge x_0[-2.48, -1.12] \rightarrow C1$
R8	$x_1[0.23, 0.71] \wedge x_0[0.68, 1.14] \rightarrow C1$
R9	$x_1[0.71, 1.66] \wedge x_0[-2.29, 1.11] \rightarrow C1$
R10	<i>Default</i> $\rightarrow C0$

are obviously better than that of C4.5R, especially for spiral problem. For the case of spiral problem, C4.5R is difficult to extract effective rules, but the new approach has so impressive results that are beyond our anticipation. This means that the new approach proposed can improve the validity of the extracted rules for complicated classification problems remarkably. Moreover, the generalization ability of those rules extracted by the new approach is also better than that of rules extracted by the C4.5R.

**Table 5.** Rules Set for the Iris Plant Problem Generated by the Algorithm Proposed

R1	$petallength < 4.1 \wedge petalwidth < 0.61 \longrightarrow Iris - setosa$
R2	$petalwidth \geq 1.76 \longrightarrow Iris - virginica$
R3	$petalwidth[1.34, 1.76] \wedge petallength \geq 5.38 \longrightarrow Iris - virginica$
R4	$Default \longrightarrow Iris - versicolor$

## 5 Conclusions

In this paper, based on the analysis of the relations among the characteristics of position and shape of classification hypersurface, the partial derivative distribution of the trained SVM output to the corresponding inputs, a new measure for determining the importance level of the attributes based on the differential information of trained SVM is proposed, which is suitable for both continuous attributes and discrete attributes, and can overcome the shortcomings of the measure method based on information entropy. On the basis of this new measure, a new approach for rules extraction from trained SVM is presented, which is also suitable for classification problems with continuous attributes. The performance of the new approach is demonstrated by several typical examples, the computing results prove that the new approach can improve the validity of the extracted rules remarkably compared with other rule extracting approaches, especially for complicated classification problems.

## References

1. Fu,L.M.: Rule generation from neural networks. IEEE Transactions on Systems, Man and Cybernetics. 8(1994)1114-1124
2. Towell,G.,Shavlik,J.A.:The extraction of refined rules from knowledge-based neural networks.Machine Learning.1(1993):71-101
3. Lu,H.J.,Setiono,R.,Liu.H.: NeuroRule: a connectionist approach to data mining. In: Proceedings of 21th International Conference on Very Large Data Bases,Zurich, Switzerland (1995)81-106
4. Zhou, Z.H.,Jiang,Y.,Chen,S.F.:Extracting symbolic rules from trained neural network ensembles. AI Communications.6(2003):3-15
5. Sestito,S.,Dillon T.:Knowledge acquisition of conjunctive rules using multilayered neural networks. International Journal of Intelligent Systems.7(1993)779-805
6. Craven,M.W.,Shavlik,J.W.:Using sampling and queries to extract rules from trained neural networks. In: Proceedings of the 11th International Conference on Machine Learning, New Brunswick, NJ,USA(1994)37-45
7. Maire,F.:Rule-extraction by backpropagation of polyhedra. Neural Networks.12(1999)717-725
8. Setiono,R.,Leow,W.K.: On mapping decision trees and neural networks.Knowledge Based Systems.12(1999)95-99
9. Battiti R.A.: Using mutual information for selecting featuring in supervised net neural learning. IEEE Trans on Neural Networks. 5(1994) 537-550

10. Bollacker K.D., Ghosh J.C.: Mutual information feature extractors for neural classifiers. In:Proceedings of 1996 IEEE international Conference on Neural Networks,Washington (1996)1528–1533
11. Dash M., Liu H., Yao J.C.: Dimensionality reduction of unsupervised data. In:Proceedings of the 9th International Conference on Tools with Artificial Intelligence,Newport Beach (1997)532-539
12. Fu, X.J., Wang, L.P.: Data dimensionality reduction with application to simplifying RBF network structure and improving classification performance. IEEE Transactions on Systems, Man and Cybernetics,Part B - Cybernetics.33(2003)399-409
13. Kamarthi,S.V., Pittner,S.: Accelerating neural network training using weight extrapolation. Neural Networks. 12(1999)1285-1299
14. Blake,C., Keogh, E.,Merz,C. J.: UCI repository of machine learning databases [<http://www.ics.uci.edu/mlearn/ML Repository.htm>], Department of Information and Computer Science, University of California,Irvine,CA. (1998)

# Feature Selection for Bagging of Support Vector Machines

Guo-Zheng Li<sup>1,2,\*</sup> and Tian-Yu Liu<sup>1</sup>

<sup>1</sup> School of Computer Engineering and Science, Shanghai University,  
Shanghai 200072, China

<sup>2</sup> State Key Laboratory for Novel Software Technology, Nanjing University,  
Nanjing 210093, China  
[gzli@staff.shu.edu.cn](mailto:gzli@staff.shu.edu.cn)

**Abstract.** Feature selection for the individuals of bagging is studied in this paper. Ensemble learning like bagging can effectively improve the performance of single learning machines, and so can feature selection, but few has studied whether feature selection could improve bagging of single learning machines. Therefore, two typical feature selection approaches namely the embedded feature selection model with the prediction risk criteria and the filter model with the mutual information criteria are used for the bagging of support vector machines respectively. Experiments performed on the UCI data sets show the effectiveness of feature selection for the bagging of support vector machines.

## 1 Introduction

Ensemble learning and feature selection are two hot topics in the machine learning community [1], which have been widely used to improve the generalization performance of single learning machines. For ensemble learning, a good ensemble is one whose individuals are both accurate and make their errors on different parts of the input space [1]. The most popular methods for ensembles creation are Bagging and Boosting [1,2,3]. The effectiveness of such methods comes primarily from the diversity caused by re-sampling the training set. For feature selection, different methods can be categorized into the filter model, the wrapper model and the embedded model [4,5]. The filter model is independent of the learning machine, while both the embedded model and the wrapper model are depending on the learning machine, but the embedded model has lower computational complexity than the wrapper model does and has been widely studied in recent years especially by using support vector machines [5,6].

Both feature selection and bagging can improve the generalization performance of single learning machines, but what about the generalization ability of bagging if it is combined with feature selection, or can feature selection further improve the generalization ability of bagging? Fortunately, Zhou et al. proposed a selective ensemble learning method, GASEN, where genetic algorithm was employed to select the individuals of bagging, and obtained satisfactory results [7].

---

\* Corresponding author.

GASEN is proposed from the view of selecting optimal individual subset for bagging, yet, few people has studied feature selection for bagging from the view of selecting optimal feature subset for the individuals of bagging. Therefore, here we employ feature selection approaches for individual learning machines of bagging to test if feature selection can improve the generalization performance of bagging.

Valentini and Dietterich have studied on bagging of support vector machines (SVMs) and concluded that improving the accuracy of SVMs would improve the accuracy of their bagging [8]. Recently, we have finished one work which used an embedded feature selection method to improve the accuracy of SVMs and obtained satisfactory results [6]. Therefore, we furthermore employ the embedded feature selection model to improve the generalization performance of bagging of SVMs. At the same time, the filter model of feature selection is used widely and frequently [9]. In this paper, both the embedded feature selection model and the filter model will be employed to select optimal feature subset for the individual learning machines of bagging.

The rest of this paper are arranged as follows: In Section 2, The embedded feature selection model with the prediction risk criteria and the filter model with the mutual information criteria for bagging are described in detail. Then, experiments on UCI data sets are performed in Section 3. At last, conclusions are given in Section 4.

## 2 Feature Selection for Bagging of SVMs

Ensemble learning methods like bagging can effectively improve the generalization accuracy of single learning machines, this is approved and validated in many works [1,2,3,7]. Especially for support vector machines (SVMs), Valentini and Dietterich concluded that reducing the error of single SVMs would reduce the error of bagging of SVMs [8]. At the same time, feature selection can also effectively improve the generalization ability of single learning machines, where the filter model and the embedded model are two popular methods [4,5]. Therefore, employing feature selection to further improve the generalization performance of bagging of single learning machines is an interesting issue. Motivated by these works, we propose to use the embedded feature selection method with the prediction risk criteria and an filter model based method for bagging of SVMs to test if feature selection can effectively improve the accuracy of bagging methods.

### 2.1 The Prediction Risk Based Feature Selection Method

Since the embedded model was just employed on SVMs and obtained satisfactory results [6], we firstly employ this model to ensemble learning, where the prediction risk criteria is used to rank the features. The prediction risk criteria was proposed by Moody and Utans [10] which evaluates each feature through estimating prediction error of the data sets when the values of all examples of this feature are replaced by their mean value:

$$S_i = \text{ERR}(\bar{x}^i) - \text{ERR} \quad (1)$$

where  $ERR$  is the training error, and  $ERR(\bar{x}^i)$  is the test error on the training data set with the mean value of  $i^{th}$  feature and defined as

$$ERR(\bar{x}^i) = \frac{1}{\ell} \sum_{j=1}^{\ell} (\tilde{y}(x_j^1, \dots, \bar{x}^i, \dots, x_j^D) \neq y_j)$$

where  $\ell$ ,  $D$  is the number of examples  $S_i$  and features,  $\bar{x}^i$  is the mean value of the  $i^{th}$  feature.  $\tilde{y}()$  is the prediction value of the  $j^{th}$  example after the value of the  $i^{th}$  feature is replaced by its mean value. Finally, the feature corresponding with the smallest will be deleted, because this feature causes the smallest error and is the least important one.

In this paper, the embedded feature selection model with the prediction risk criteria is employed to select relevant features for the individuals of bagging of SVMs, which is named as PRIFEB (Prediction Risk based Feature sElection for Bagging). The basic steps of PRIFEB are described as follows.

### **Algorithm PRIFEB**

Suppose  $T_r(x^1, x^2, \dots, x^D, C)$  is the training set and  $p$  is the number of individuals of ensemble.

$T_r$  and  $p$  are input into the procedure and ensemble model  $L$  is the output.

- Step 1.** Generate a training subset  $T_{rk}$  from  $T_r$  by using Bootstrap sampling algorithm [2], the size of  $T_{rk}$  is three quarters of the size of  $T_r$ .
- Step 2.** Train an individual model  $L_k$  on the training subset  $T_{rk}$  by using support vector machines algorithm and calculate the training error  $ERR$ .
- Step 3.** Compute the prediction risk value  $S_i$  using Equation (1). If  $S_i$  is greater than 0, the  $i^{th}$  feature is selected as one of optimal features.
- Step 4.** Repeat Step 3 until all the features in  $T_{rk}$  are computed.
- Step 5.** Generate the optimal training subset  $T_{rk-optimal}$  from  $T_{rk}$  according to the optimal features obtained in Step 3.
- Step 6.** Re-train the individual model  $L_k$  on the optimal training subset  $T_{rk-optimal}$  by using support vector machines.
- Step 7.** Repeat from Step 2 to Step 6 until  $p$  models are set up,  $p$  is 20 in this paper.
- Step 8.** Ensemble the obtained models  $L$  by the way of majority voting method for classification problems.

## 2.2 The Mutual Information Based Feature Selection Method

The above embedded feature selection model is dependent on the learning machine, while the filter model is independent and used frequently, where the mutual information criteria is studied widely [9]. The mutual information describes the statistical dependence of two random features or the amount of information that one feature contains about the other and it is a widely used information theoretic measure for the stochastic dependency of discrete random features.

The mutual information between two features  $R$  and  $S$  is defined in terms of the probability distribution of intensities as follows:

$$I(R : S) = \sum_{r \in R} \sum_{s \in S} p\{r, s\} \lg \frac{p\{r, s\}}{p\{r\}p\{s\}}. \quad (2)$$

where  $p\{r, s\}$  is the combined probability distribution of intensities of two features  $R$  and  $S$ .  $p\{r\}$  and  $p\{s\}$  respectively are the individual probability distribution of intensities of  $R$  and  $S$ .

The mutual information criteria has been widely used in the filter feature selection model [9], therefore, we employ this method to bagging and propose a method, MIFEB (Mutual Information based Feature sElection for Bagging), and compare it with PRIFEB to test which feature selection model is more effective. The basic steps of MIFEB are described as follows.

### Algorithm MIFEB

$T_r$  and  $p$  are input into the procedure and an ensemble model  $L$  is the output.

- Step 1.** Generate a training subset  $T_{rk}$  from  $T_r$  by using Bootstrap sampling algorithm, the size of  $T_{rk}$  is three quarters of the size of  $T_r$ .
- Step 2.** Employ the mutual information formula (2) on the training subset  $T_{rk}$  and the target set of  $T_{rk}$  to obtain the value vector  $MI$ .
- Step 3.** Rank the vector  $MI$  in descending order and sum  $MI$  to obtain  $SUM(MI)$ .
- Step 4.** Select all of the first features as the optimal subset, whose total values should be greater than  $RMI * SUM(MI)$ .  $RMI$  is a pre-defined ratio which is greater than 0 but less than 1.
- Step 5.** Generate the optimal training subset  $T_{rk-optimal}$  from  $T_{rk}$  according to the optimal features obtained in Step 4.
- Step 6.** Train the individual model  $L_k$  on the optimal training subset  $T_{rk-optimal}$  by using support vector machines.
- Step 7.** Repeat from Step 2 to Step 6 until  $p$  models are set up,  $p$  is 20 in this paper.
- Step 8.** Ensemble the obtained models  $L$  by the way of majority voting method for classification problems.

As for time complexity, comparing with the general case of bagging, the times of SVMs training in PRIFEB are double of that of bagging, and there are additional  $p * D$  times of testing SVMs on the training data set in PRIFEB. While in MIFEB, there are additional  $p * D$  times of computation of mutual information between the feature vector and the label vector.

## 3 Numerical Experiments

Since the proposed algorithms PRIFEB and MIFEB employ feature selection approaches to improve the generalization performance of bagging of SVMs, here

PRIFEB and MIFEB are only compared with the general case of bagging. Thirteen data sets listed in Table 1 are selected from the UCI machine learning repository [11]. The number of instances in these data sets ranges from hundreds to thousands and the number of features ranges from 9 to 70. The nominal values are changed to be numerical in all data sets. Then, all the attributes are transformed into the interval of [-1, 1] by an affine function.

**Table 1.** The properties of the UCI data sets for comparison

Data set	Number of classes	Number of features	Number of instances
all-bp	3	29	3772
all-hyper	5	29	3772
all-hypo	4	29	3772
backup	19	35	683
audio	24	70	226
proc-C	5	13	303
proc-H	2	13	294
soybean-l	19	34	307
statlog-h	2	13	270
glass	6	9	214
voting-records	2	16	435
Ionosphere	2	34	351
breast-cancer-W	2	9	699

The hold out method is used to validate the results. Experiments are repeated fifty times on each data set. The same pair of parameters for SVMs,  $C = 100$ ,  $\sigma = 10$ , is used and the number of individuals for bagging is 20.  $RMI$  for mutual information algorithm is 0.9 in our experiments. Experimental results of accuracy obtained by the different bagging methods are shown in Table 2.

It can be seen from Table 2 that the mean accuracy obtained by the bagging methods with feature selection are improved in various degree on all the data sets which range from 0% to 6%, the mean value improved on all the data sets is 2.35 percent for the PRIFEB method, and 2.16 percent for the MIFEB method.

It can also be seen that PRIFEB performs slightly better than MIFEB on eight out of thirteen data sets, and worse on five data sets. Yet, in fact, the deference between the results obtained by PRIFEB and MIFEB is so slight (less than 1% on all the data sets) that we consider these two feature selection approaches perform equally well.

Numbers of features in the optimal feature subsets obtained by different bagging methods on different data sets are shown in Table 3, where R\_PRIFEB means the ratio of the number of features in the optimal subsets obtained by PRIFEB to that of the total, and R\_MIFEB means the ratio of the number of features in the optimal subsets obtained by MIFEB to that of the total. From Table 3, it can be seen that all the data sets need to remove some features in some degree, and MIFEB obtains slightly smaller feature subsets than PRIFEB does.

**Table 2.** Statistical prediction accuracy by the bagging methods with feature selection and without feature selection on the UCI data sets

Data set	PRIFEB	MIFEB	Bagging
all-bp	97.07 ± 0.45	96.86 ± 0.51	95.95 ± 0.13
all-hyper	97.87 ± 0.21	97.93 ± 0.16	96.76 ± 0.43
all-hypo	97.42 ± 0.43	97.08 ± 0.16	96.66 ± 0.61
backup	91.99 ± 1.41	91.47 ± 1.62	89.90 ± 2.06
audio	75.19 ± 3.46	75.66 ± 3.72	74.67 ± 4.23
proc-C	53.71 ± 3.67	54.16 ± 2.40	49.93 ± 3.65
proc-H	79.84 ± 2.55	79.70 ± 2.71	73.89 ± 3.52
soybean-l	85.54 ± 3.64	84.30 ± 3.38	83.25 ± 3.57
statlog-h	78.62 ± 3.30	78.98 ± 2.87	74.88 ± 3.87
glass	64.95 ± 3.74	65.78 ± 4.01	61.75 ± 5.12
voting-records	95.59 ± 1.08	94.30 ± 4.88	94.24 ± 1.40
Ionosphere	88.70 ± 2.16	88.21 ± 2.74	87.28 ± 3.07
breast-cancer-W	94.38 ± 1.02	94.01 ± 0.92	91.23 ± 1.71
Average	84.68 ± 2.08	84.49 ± 2.31	82.33 ± 2.57

**Table 3.** Average number of the optimal features obtained by PRIFEB and MIFEB

Data set	PRIFEB	R_PRIFEB	MIFEB	R_MIFEB
all-bp	23.86 ± 0.77	82.27%	25.58 ± 0.24	88.21%
all-hyper	24.81 ± 0.98	85.55%	25.60 ± 0.27	88.28%
all-hypo	18.70 ± 0.13	64.48%	25.50 ± 0.31	87.93%
backup	31.76 ± 0.26	90.74%	25.81 ± 0.73	73.74%
audio	65.71 ± 0.16	93.87%	66.14 ± 0.38	94.49%
proc-C	11.97 ± 0.03	92.07%	9.86 ± 0.24	75.85%
proc-H	11.26 ± 0.12	86.62%	10.94 ± 0.27	84.15%
soybean-l	31.05 ± 0.07	91.32%	27.57 ± 0.74	81.09%
statlog-h	11.61 ± 0.18	85.85%	10.42 ± 0.28	80.15%
glass	7.96 ± 0.04	88.44%	7.06 ± 0.16	78.44%
voting-records	13.60 ± 0.31	85.00%	11.55 ± 0.34	72.19%
Ionosphere	30.81 ± 0.17	90.62%	30.07 ± 0.21	88.47%
breast-cancer-W	7.67 ± 0.13	85.22%	7.33 ± 0.09	81.44%
Average	22.37 ± 0.26	86.31%	21.80 ± 0.32	82.65%

## 4 Conclusions

Feature selection for the bagging of support vector machines (SVMs) has been studied. From the experimental results on UCI data sets, it can be seen that both the filter model with the mutual information criteria and the embedded model with the prediction risk criteria have improved the generalization performance of bagging of SVMs. Feature selection can reduce the irrelevant features and even redundant features to improve prediction accuracy of single individuals, which has also been proved by the previous study. At the same time, feature selection reduces different features for different individuals and helps to increase

the diversity among the individuals of bagging. According to the theory [1], improving the accuracy of every individuals and increasing the diversity among them will effectively improve the accuracy of an ensemble method. The above two aspects caused by feature selection make it true that feature selection effectively improve the generalization ability of bagging.

This work has showed the effectiveness of feature selection for bagging, the future work will be to extend our work to other ensemble methods like boosting.

## Acknowledgement

This work was supported by NSF of China (20503015) and SHMEC NSF (05AZ67). Thank the anonymous reviewers for their valuable remarks.

## References

1. Dietterich, T.: Machine-learning research: Four current directions. *The AI Magazine* **18** (1998) 97–136
2. Breiman, L.: Bagging predictors. *Machine learning* **24** (1996) 123–140
3. Bauer, E., Kohavi, R.: An empirical comparison of voting classification algorithms: Bagging, boosting, and variants. *Machine learning* **36** (1999) 105–139
4. Yu, L., Liu, H.: Efficient feature selection via analysis of relevance and redundancy. *Journal of Machine Learning Research* **5** (Oct) (2004) 1205–1224
5. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. *Journal of machine learning research* **3** (2003) 1157–1182
6. Li, G.Z., Yang, J., Liu, G.P., Xue, L.: Feature selection for multi-class problems using support vector machines. In: *Lecture Notes on Artificial Intelligence* 3173, Auckland, New Zealand, Springer (2004) 292–300
7. Zhou, Z.H., Wu, J.X., Tang, W.: Ensembling neural networks: many could be better than all. *Artificial Intelligence* **137** (2002) 239–263
8. Valentini, G., Dietterich, T.: Bias-variance analysis of support vector machines for the development of svm-based ensemble methods. *Journal of Machine Learning Research* **5** (2004) 725–775
9. Liu, H., Yu, L.: Toward integrating feature selection algorithms for classification and clustering. *IEEE Trans. on Knowledge and Data Engineering* **17(3)** (2005) 1–12
10. Moody, J., Utans, J.: Principled architecture selection for neural networks: Application to corporate bond rating prediction. In *NIPS 4.*, Morgan Kaufmann Publishers, Inc. (1992) 683–690
11. Blake, C., Keogh, E., Merz, C.J.: UCI repository of machine learning databases. Technical report, Department of Information and Computer Science, University of California, Irvine, CA (1998).

# Neural Classification of Lung Sounds Using Wavelet Packet Coefficients Energy\*

Yi Liu<sup>1</sup>, Caiming Zhang<sup>1</sup>, and Yuhua Peng<sup>2</sup>

<sup>1</sup> School of Computer Science and Technology, Shandong Univ., Jingshi Avenue 73,  
250061, Jinan, China

{liuyi, czhang}@Sdu.edu.cn

<sup>2</sup> School of Computer Sci. and Eng., Shandong Univ., Shanda SourthRoad 27,  
250100, Jinan, China  
pyuhua@Sdu.edu.cn

**Abstract.** A novel method for recognition two kinds of lung sounds is presented. The proposed scheme is based on the analysis of a wavelet packet decomposition(WPD). Normal and abnormal lung sounds data were sampled from various subjects. Each signal is segmented to inspiration and expiration. From their high dimension WPD coefficients, we build the compact and meaningful energy feature vectors, then use them as the input vectors of the artificial neural network(ANN) to classify the lung sound types. Extensive experimental results show that this feature extraction method has convincing recognition efficiency although not yet good enough for clinical use.

## 1 Introduction

Over recent years, lung sounds analysis has become a hot theme in medicine field. Computerized methods for the recording and analysis of respiratory sounds have overcome many limitations of simple auscultation. It can make permanent records of the measurements and produce graphical representations that help the diagnosis of chest diseases. Normally, lung sounds analysis methods include time-expanded waveform, classical spectral analysis, parametric spectral analysis, wavelet transform et al. However, because of the randomicity of the signals, the diversity of analysis methods[1-5]and a lack of guidelines for data acquisition has made it difficult to compare results with different laboratories. With the development of computer and signal processing techniques, lung sounds recognition has become a new research issue for the investigation of respiratory sounds.

Feature extraction is the crucial work for lung sounds pattern recognition. The extracting process is to throw off the useless information and look for the most efficient signal feature to form a lower-dimensional pattern feature vectors for classification. For a given lung sounds processing system, its transfer function is stable and the corresponding amplitude-frequency and phase-frequency characteristics are also not changeable. So the sampled data may be able to reflect the signals' physical

---

\* Supported by Shandong Province Nature Science Foundation(Y2005G01).

characteristic. Lung sound signals are composed of multi-frequency components and take on obvious periodic waveform. Normal and abnormal signals' energy distribution are not the same, i.e. the differences of different frequency components may reflect the characteristics of different signals. But it is hard to extract this characteristic with conventional methods, so a novel feature extraction method of lung sounds based on WPD is presented.

In this paper, our work consists of three stages. First, we do WPT and search the best basis; Second, a new feature extraction strategy is proposed based on the distribution of signal energy by using WPT coefficients. Finally, these energy feature is used as the input vector of the BackPropagation(BP) network to classify the given signals(normal and abnormal). The experimental results are compared with the previously published methods.

## 2 WPD and Energy Representation

In this section, we'll provide a more detailed WPD algorithm based on filter banks and give out the energy representation fashion.

### 2.1 WPD Algorithm

According to the multi-resolution analysis(MRA) theory[6], the signal is divided into subbands by successive lowpass filtering operator  $H$  and a highpass filtering operator  $G$ . This corresponds respectively to presenting the signal using orthonormal set of basis functions, consisting of scaling function  $\phi(t)$  for  $G$  and the associated mother wavelet function  $\psi(t)$  for  $H$ . In the wavelet packet case, the lowpass and the highpass filtered parts at each step are split, so that the bandwidth of each subband is equal. Now  $\phi(t)$  and  $\psi(t)$  are expressed as  $u_0(t)$  and  $u_1(t)$  respectively, and the associated basis functions  $u_n(t)$  are now defined[7] recursively as:

$$u_{2n}(t) = \sqrt{2} \sum_{k \in Z} h(k) u_n(2t - k), \quad u_{2n+1}(t) = \sqrt{2} \sum_{k \in Z} g(k) u_n(2t - k) \quad (1)$$

where  $\{h(k)\}$  and  $\{g(k)\}$  are the filter coefficients and satisfy  $g(k)=(-1)^{1-k} h(1-k)$ , they are known as quadrature mirror filter banks. The shifted series  $\{u_n(t)\}(n \in Z^+)$  are called orthogonal wavelet packet.

If the finite energy signal  $x(t)$  is the function in  $L^2(R)$  space, its sampling series is  $\{x(p)\}_{p=1,2,\dots,N}$ , then  $x(t)$  can be decomposed over this orthogonal basis, the WPD coefficients  $\{C_p^{j,n} \mid p \in Z\}$  are given by

$$C_p^{j,n} = \int_{-\infty}^{+\infty} x(t) 2^{j/2} u_n(2^j t - p) dt \quad (2)$$

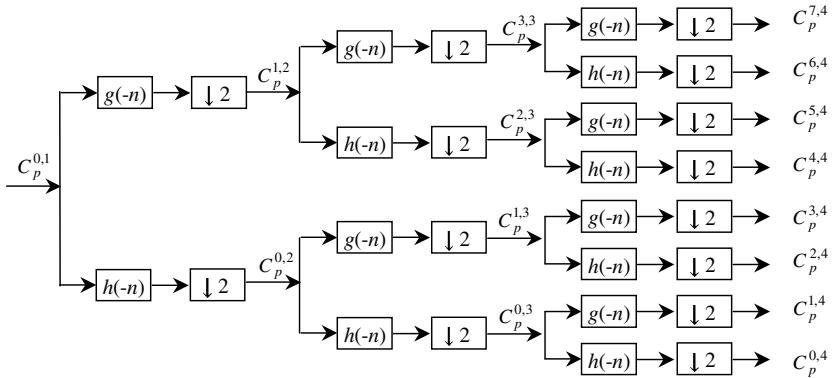
where  $j, p, n$  is the scale, position and surge parameter, respectively. The coefficients  $\{C_p^{j-1,2n} \mid p \in Z\}$  and  $\{C_p^{j-1,2n+1} \mid p \in Z\}$  are given by formula (3). Formula (3) is known as the WPD algorithm.

$$C_p^{j-1,2n} = \sum_k h_{k-2p} C_k^{j,n}, \quad C_p^{j-1,2n+1} = \sum_k g_{k-2p} C_k^{j,n} \quad (3)$$

The WPD algorithm is hierarchical. It consists with the MRA theory. Thus we can use filter banks to represent the WPD algorithm. Moreover, a fast discrete WT is implemented by cascading these conjugate mirror filters. The equivalence between this continuous time wavelet theory and discrete filter banks led to a new fruitful interface between digital signal processing and harmonic analysis, and this also create a culture shock that is not totally resolved today.

In the case of WPD, the lowpass filter output from the initial or first level filtering of the signal is applied to the same lowpass and highpass filters. The output of the lowpass filter is again applied to the same lowpass and highpass filters, and so on, until all samples in the block of length  $N$  have been exhausted through the successive filtering. Downsampling by two occurs after each operation, as we will see in Fig.1.

Filters inputs and outputs are vectors. The blocks are all linear operators based on filter definitions[8]. Since the analysis filters are orthonormal, the synthetical filters are simply the transpose of the analysis filters. The downward arrow followed by the number 2 indicates downsampling by a factor of two.



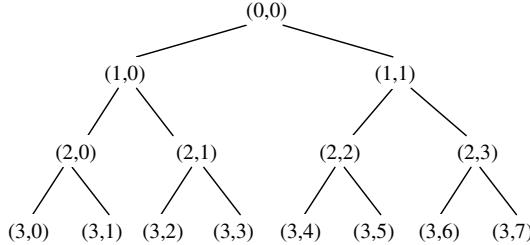
**Fig. 1.** Two channel WPD orthogonal filter banks structure-depth 3 analysis banks

The downsampling of the filter outputs is an important part of the filter banks operation, resulting in what is called a *critically sampled* transform. Without downsampling, each filtering stage would double the number of samples at its inputs. Also by downsampling, we can apply the identical filters at each stage and have them operated at successively larger scales. Downsampling by a factor of two specifically because we use exactly two filters in our hierarchical subbands splitting that the so-called *dyadic* spacing is produced.

The hierarchical filtering scheme above is in fact a representation of the *pyramid algorithm*[6]. When it is implemented in matrix form it is called the *fast wavelet transform*. It is so efficient because at each stage other than the first, we are filtering the output of a previous filtering operation. Such a recursive approach results in a maximum of  $2LN$  multiplications for a length  $N$  transform when filters of length  $L$  are used.

Because of the recursive nature of this approach, a tree-like structure emerges when we construct either a time or frequency domain representation of it.

Fig.2 shows the tree diagram associated with a depth 3 WPD. It reflects the structure of its corresponding hierarchical filter banks shown in Fig.1.



**Fig. 2.** Tree Diagram of Wavelet packet Transform – depth 3

Moving from top to bottom in the diagram of Fig.2, frequencies are divided into ever smaller segments. Each line that emanates down and to the left of a node represents a lowpass filtering operation ( $h$ ), and to the right a highpass filtering operation ( $g$ ). The nodes that have no further nodes emanating down from them are referred to as terminal nodes, leaves, or subbands. We refer to the other nodes as non-terminal, or internal nodes. The first line represents the original signal bandwidth. The other nodes are computed from their father by one application of either lowpass or highpass quadrature mirror filters. The bandwidth is narrower half with each filtering operation. In the bottom line, each subband is eighth of the original signal bandwidth. Thus multiresolution is achieved and the tree nodes labeling scheme provides a simple mechanism for indicating the nodes in the tree that we can work with.

Then analogous to the wavelet transform case, the WPD does not require the explicit definition of the wavelet. The filter definitions are enough, and we sometimes refer to this filter bank based implementation as the fast WPT (FWPT).

To see what the equivalent wavelet packets would be at a given decomposition level, we can do a recursion operation at each node moving down the diagram, to get the wavelets at the next level. Using WPD, we can gain more abundant local time frequency information. The analysis and detection using wavelet packets are more fit for mutation and non-stationary signals.

## 2.2 Energy Representation

In this section, we'll give the energy representation fashion associated with the wavelet packet coefficients.

**Define 1.** The square norm of signal  $x(t)$  in  $L^2(R)$  is defined[7] as

$$\|x\|_2^2 = \int_R |x(t)|^2 dt \quad (4)$$

This means the signal energy in time domain is equals to its square norm in  $L^2(R)$ .

**Theorem 1.** For WPD in best basis, its orthonormal WP basis function is  $U_p^{j,n}$ , then

$$\sum_{j \in Z} \sum_{n \in Z^+} \sum_{p \in Z} \left| \langle x, U_p^{j,n} \rangle \right|^2 = \|x\|_2^2 \quad (5)$$

This indicates that the serial  $\{x\}$  can be decomposed and its energy is equal to the energy sum of each WPD coefficients. Shown in Fig.1, four levels is calculated in  $V_j$  space, and two scale is used in  $W_j$  space, then

$$\sum_{n=0}^7 \sum_{p \in Z} \left| \langle x, U_p^{4,n} \rangle \right|^2 + \sum_{n=2}^3 \sum_{p \in Z} \left| \langle x, U_p^{2,n} \rangle \right|^2 = \|x\|_2^2 \quad (6)$$

Thus the equivalence relation in time domain between wavelet coefficients and signal energy is found.

Let  $E_{j,n}$  denote the corresponding energy of WPT coefficient  $S_{j,n}$ , then

$$E_{j,n} = \sum_{p=1}^N |C_p^{j,n}|^2, \quad n = 0, 1, 2, \dots, 2^j - 1, j \in Z \quad (7)$$

where  $N$  is the number of discrete sampling points,  $C_p^{j,n}$  is the high dimension coefficients matrix of the WPD and determined by (3) formula.

The analysis above show that different lung sounds have different energy in different frequency bands. So we define the energy as signal feature. In the next section, we want to extract feature corresponding to the best basis.

### 3 Best Basis Selection and Feature Extraction

In our work, we have generally created the *complete tree* basis first, and then give a best basis algorithm[8] to find a basis that minimizes a given cost function.

The scaled and translated versions of  $\{u_n(t)\}(n \in Z^+)$  form a library of basis functions. These wavelet packets collectively make up the complete family of possible bases, and many potential bases can be constructed from them. If only the lowpass filter is used, the result is the wavelet basis. If two filters are iterated, then results in the complete tree basis(shown in Fig.2). Between these two extremes lie a large number of possible bases and associated subtrees. Nodes can be merged or split based on whatever requirements are at hand. In all cases, the leaves of each connected subtrees of the complete wavelet packet tree form a basis of the initial space; they span the space in a linearly independent fashion.

Once the WPD of the signal is achieved, the best possible representation among the library of bases would be chosen using the best basis search algorithm[9]. Because from Fig.2, the WPD is very *complexity*, there are roughly  $2^N$  possible bases associated with a transform of block length  $N$ . For example, a 16 samples block, which incidentally can support a maximum depth-4 transform, would have over 65,536 possible bases derivable from the initial WP basis. Thus the fairly efficient transform that produces the full WP tree in  $O(N \log N)$  operations.

In general, the different WP basis conduce to different analysis results. So two issue must be solved: one is how to evaluate a basis using a “suitability criterion”, and another is how to find the wanted best basis from the given library.

Then the best basis searching algorithm and the feature extraction procedure consists of several steps are as follows:

*Step 1:* Segment the signal into inspiration and expiration, and calculate the WPD coefficients of each segment at given scales.

Here, the maximum scale of WPD is four(See *Step 3*). Different wavelet families, which Daubechies wavelets of order 5 and Symlet wavelets of order 8 are tried to get the best results. The comparison of classification results indicate that Sym8 is more suitable for our case. This is because the Sym8 has much better local peculiarity.

*Step 2:* Search for the best tree (basis), which can efficiently represent the original signal, based on the best basis algorithm[8]. In this algorithm, the information cost function  $M$  should reflect the convergence of the WPD coefficients. Therefore, if the distribution of the signal’s energy is uniformity relatively, the value of  $M$  should be higher and the corresponding basis does not respond to the user’s wishes. If the distribution is mainly concentrated on several coefficients and the other value can be omitted, this  $M$  should be lower and the associated basis does fit for our purpose.

The most common cost function is the additive entropy which measures both the energy concentration and the dynamic ranging of the coefficients. They are the Shannon entropy and the  $l^p$ -criterion, which are defined as:

**Define 2.** If the serie  $u = \{u_j\}$  is the associated WPD coefficients,  $P_j = |u_j|^2 / \|u\|^2$  and  $P_j \log P_j = 0$  when  $P=0$ , then the Shannon information entropy of  $u$  is

$$M(u) = - \sum_j P_j \log P_j \quad (8)$$

where  $M(0) = 0$ ,  $M(\{u_j\}) = \sum_{j \in Z} M(u_j)$ .

**Define 3.**  $l^p$ -criterion is defined as:

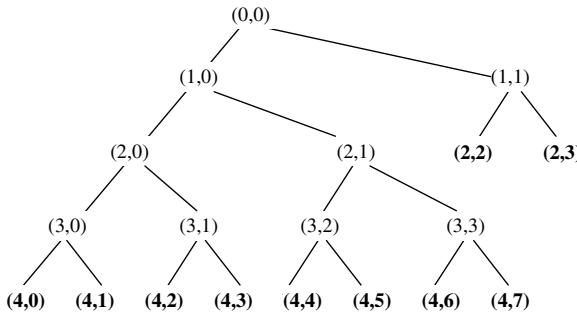
$$l^p(u) = - \sum_j |u_j|^p, \quad 0 < p < 2 \quad (9)$$

where  $u_j$  is a sequence of coefficients corresponding to a subband.

The experiments of Shannon entropy and  $l^p$ -criterion with  $p = 1.0, 1.1, 1.2, 1.3, 1.5$  indicate that  $p = 1.2$  could give the relatively better results.

Starting from the terminal nodes of the tree, each pair of subbands’ cost is compared to its father’s and a decision is made whether to keep the representations of the sons, or to combine them into the father subbands. If the parents’ cost is less, then it is retained and the children nodes are pruned. Otherwise the children nodes are kept and the parents’ cost is replaced with the sum of the children. Move recursively up to the top level. Then we get the best tree, which has least costs comparing with other choices of the tree structure. This searching for the minimum cost pruned tree would take very long if no structure existed. Since we assume an orthogonal WPD, each

node with its children can be treated individually, efficient basis selection is possible. The nodes with lowest value of the cost function comprise of a disjoint cover in  $L^2(R)$  space, and then the best basis are the terminal nodes  $(4,0)$ ,  $(4,1)$ ,  $(4,2)$ ,  $(4,3)$ ,  $(4,4)$ ,  $(4,5)$ ,  $(4,6)$ ,  $(4,7)$ ,  $(2,2)$  and  $(2,3)$  (marked with larger bold font) from left to right along the bottom row, shown in Fig.3.



**Fig. 3.** The best tree of a normal lung sounds recording-depth 4

*Step 3:* Subdivide the signals frequency components into ten disjoint portions.

In this study, usually the lung sounds frequencies are less than 1000 Hz conventionally according to the analysis results of Burg spectrum method. Then downsampling frequency of 2750Hz is employed since our sampling frequency is 22,050Hz. According to the Naquist Sampling Theorem, the maximum signal frequency is 1375Hz. Thus the best tree, which roughly correspond to ten unequal widths frequencies bands are obtained. They are 0-86Hz, 86-172Hz, 172-258Hz, 258-343Hz, 343-429Hz, 429-515Hz, 515-601Hz, 601-687Hz, 687-1030Hz and 1030-1375Hz, respectively. This result in the maximum scale of WPD is four.

*Step 4:* Calculate the ten corresponding subbands coefficients energy according to equation (7). Before they are used as input vectors of the classification neural network, they should be normalized in order to give the similar distances between feature vectors.

*Step 5:* Code the coefficients energy from lower to higher in these subbands. The feature vectors, which include these 10 elements, can be constructed as

$$[E_{40}, E_{41}, E_{42}, E_{43}, E_{44}, E_{45}, E_{46}, E_{47}, E_{22}, E_{23}]$$

*Step 6:* Repeat steps 1 to 5 for each lung sounds segment, then obtain all the feature vectors needed in the classification ANN algorithm.

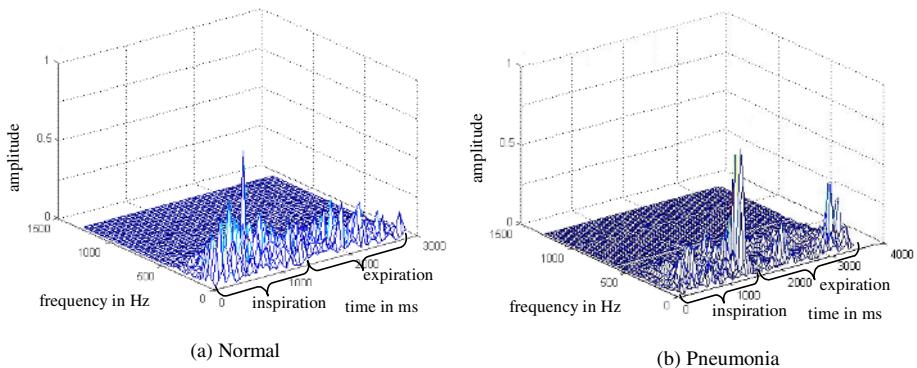
The extracted WP coefficients provide a compact representation that shows the energy distribution of the signal in frequency domain. After 4 times decomposition, each subband has different frequency components. Thus we can assume that feature vectors abstracted from different lung sounds are different enough to enable classification using ANN. The experiment results bellow indicate that our assumption is reasonable and practicable.

*Remark.* This approach does not afford any user because there are no parameters to set. You can modify this algorithm and replace them with another Entropy such as Threshold Entropy, Logarithm Energy Entropy, and so on.

## 4 Experimental Results

To verify the performance of the feature extraction algorithm, we create the needed lung sounds library because there are no universal public ones. Lung sounds are sampled from the subjects' back in relaxed respiration and sitting position condition under the supervision of senior physician in hospital. Recording and filtering the representative normal and abnormal(include bronchus, pneumonia and asthma) signals. The sampling rate is 22050Hz, recording time is 60s and the A/D resolution is 16 bits. This experiment is fulfilled with  $2 \times 100$  lung sounds signals.

In working, the expiration and inspiration are detached and the feature is extracted respectively. So we can find the best character associated with each lung sounds. The time-frequency-amplitude distributions of two signals in one respiratory period are shown in Fig.4.



**Fig. 4.** Time-frequency-amplitude distributions of two lung sounds in one respiratory periods

Db5 and Sym8 are used. The scale parameter is 4. The WPD coefficients and their energy are calculated according to the formula (3) and (7), respectively. Two individuals energy feature values using Sym8 are shown in Table 1.

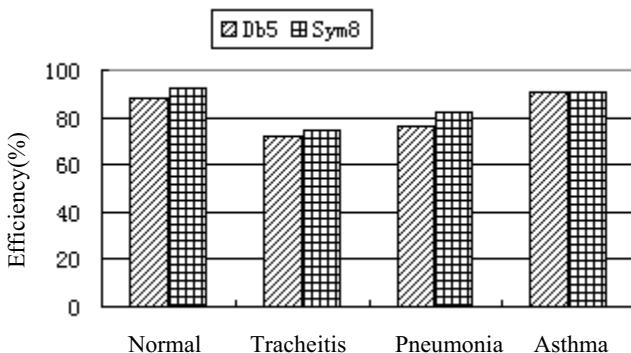
The twenty energy feature values of one signal are used as one group of input vectors for BP network(20-45-4 structure) to classify the lung sounds types. The classifier consists of feed forward (FF) neural network using BP learning method to train the network. The data set is randomly divided into two subsets, 40(10 of which are normal sounds) for training and the remaining 60(normal 15, tracheitis 15, pneumonia 15 and Asthma 15, respectively) for validation to check the classification performance. So the average correct classification efficiency is 81.7% for Db5 and 85% for Sym8.

The correct classification results of various types of lung sounds are shown in Fig.5. It indicates that Sym8 is slightly good.

Note: The wrong classification may be brought by the BP network itself, the improper feature selection, or the shortage of typical samples.

**Table 1.** Energy feature vectors extracted from two lung sounds signals

Energy	Normal		Pneumonia	
	Inspiration	Expiration	Inspiration	Expiration
$E_{40}$	2.766	1.416	0.148	2.416
$E_{41}$	5.175	1.442	0.222	2.127
$E_{42}$	0.522	0.251	0.058	0.758
$E_{43}$	2.804	0.702	0.101	0.708
$E_{44}$	0.002	0.001	0.001	0.064
$E_{45}$	0.025	0.011	0.004	0.191
$E_{46}$	0.361	0.146	0.082	1.487
$E_{47}$	0.138	0.055	0.057	1.503
$E_{22}$	0.003	0.003	0.001	0.015
$E_{23}$	0.012	0.010	0.006	0.256



**Fig. 5.** The comparison of classification results using two kinds wavelets

The comparison with those of the references methods is presented in Table 2. Kan-daswamy[5] who using statistical features of wavelet coefficients(21 normal, 105 for abnormal) and Pesu[10] who using Learning Vector Quantization(9 abnormal). Although our method is not the best, the feature extraction is very easy and simple, and our method is based on more large data library. This means our approach has more persuasion than that in smaller database.

**Table 2.** The Comparison of several experimental results

Methods	Dataset	Efficiency(%)
Kandaswamy <sup>[5]</sup>	6×21	91.67
Pesu <sup>[10]</sup>	9	71.2
Db5 based	2×100	81.7
Sym8 based	2×100	85.0

## 5 Conclusion

Nowadays wavelet transform based algorithm is an important research field of digital signal processing. The recognition method proposed in this paper makes the classification ANN simplify and has high identifying efficiency relatively. The experimental results show that the WPD based analysis method fits for non-stationary signal processing. But the classification efficiency of lung sounds depends on the recorded data library, the selected training samples, the used type of wavelet packet, the ANN structure and its ability of fault tolerance. Our future work in this area would include testing a wider range of lung sounds types, looking for more comprehensive feature extraction algorithm and classification method that might incorporate the characteristics of the human audio system. Especially, we would try the SVM(Support Vector Machine) classifier since it is generally said that SVM could exhibit better performance than ANN's.

## References

1. Ono M., Arakawa K., Mori M.: Separation of Fine Crackles from Vesicular Sounds by a Nonlinear Digital Filter. IEEE Transactions Biomed. Eng., Vol. 36, (1989) 286-295
2. Fenton T. R., Pasterkamp H., Tal A.: Automated Spectral Characterization of Wheezing in Asthmatic Children. IEEE Transactions Biomed. Eng., Vol. 32, (1985) 50-55
3. Stefan P., Sagar V.: Feature Extraction From Wavelet Coefficients for Pattern Recognition Tasks. IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 21 (1999) 83-87
4. Broersen, P.M.T., Waele S: Feature Extraction with Time Series Models: Application to Lung Sounds. Instrumentation and Measurement Technology Conference, Proceedings of the 16th IEEE, Vol. 1 (1999) 370-375
5. Kandaswamy A., Kumar C., Jayaraman S.: Neural Classification of Lung Sounds Using wavelet Coefficients. Computers in Biology and Medicine, Vol. 34 (2004) 523-537
6. Mallat S.: A Theory for Multiresolution Signal Decomposition: The Wavelet Representation. IEEE Trans. On Pattern Analysis and Machine Intelligence, Vol. 11 (1989) 674-693
7. Stephane Mallat: A Wavelet Tour of Signal Processing. Washington: Academic Press(1999)
8. Peng Yuhua. Wavelet Transform and Engineering Application. Beijing: Science Press(1999)
9. Antonini, Orlandi: Wavelet Packet-based EMI Signal Processing and Source Identification. IEEE Trans on Electromagnetic Compatibility, Vol. 43 (2001) 140-148
10. Pesu L., Helisto P., Ademovic E.: Classification of Respiratory Sounds Based on Wavelet Packet Decomposition and Learning Vector Quantization. Technology and Health Care, Vol.6 (1998) 65-74

# Wireless Communication Quality Monitoring with Artificial Neural Networks

Dauren F. Akhmetov<sup>1</sup> and Minoru Kotaki<sup>2</sup>

<sup>1</sup> NTT DoCoMo Hokkaido, Inc., Sapporo, 060-0001, Japan

dauren\_akhmetov@nttdocomo-h.co.jp

<sup>2</sup> School of Eng., Hokkaido Tokai University, Sapporo, 005-8601, Japan

kotaki@de.htokai.ac.jp

**Abstract.** Quality and reliability of wireless communication is an actual issue for design of modern high-efficiency information systems in the wide area of human activities. In the paper, the problem of wireless communication reliability and methods of its evaluation are studied. The quality of communication at actual spot is estimated with the method proposed by the authors. It is based on the usage of a prediction mathematical model presenting the time series for receiving signal level data. Different model classes are considered for the data description including neural network models. Special model training procedure based on the Aggregative Learning Method (ALM) is applied along with expert approach for the data classification. The validity and the efficiency of the proposed approach have been tested through its application for different cases including “open-air” and “in-building” environments. Cellular phone communication network of DoCoMo Inc. is used as a test bed for the proposed method. Classification abilities of the method are shown reliable for estimation of the communication quality. Characterized with high computational efficiency and simple decision making procedure, the derived method can be useful for design of simple and reliable real-time systems for communication quality monitoring.

## 1 Introduction

At present, technological progress made possible the penetration of IT achievements into many spheres of human activity due to reasonable expenses and easy-to-use features of novel devices and systems. Wireless communication is a powerful method to provide user with the prompt information from remote or difficult-to-access sites. However, there is a problem of reliability of communication channel and its estimation.

On the other hand, modern information processing theory and practice provide effective methods for real data representation in compressed and laconic form of mathematical models. Identification systems using artificial neural networks (ANN) are widely spread now, due to their well-known ability of process approximation to arbitrary accuracy [1, 2].

In this paper, the time series analysis is applied for estimation of communication conditions. First, some issues of mathematical presentation of real process are discussed and so-called aggregative learning method (ALM), proposed by the authors, is explained [3-5]. Next, ALM-based Backpropagation Neural Network (BPNN) is

presented and its application for the stated problem solving is proposed. After that, an experiment on data acquisition over the mobile network of DoCoMo, Inc. is described. Then, the application of ALM for estimation of communication conditions is considered. Main results are formulated in the conclusion.

## 2 Aggregative Learning Method

Learning is a reaction of an intelligent system (IS) to the environment changes for the sake of the best environment mapping achievement. Any Intelligent System is characterized by some number of degrees of freedom (NDF) to be changed during and by the interaction with environment (learning procedure). Reduced (optimized) NDF intelligence system (Optimized Degree of Freedom System - ODFS) is characterized with the highest learning speed and the best approximation ability inside the class which it belongs to.

Further improvement of the ODFS learning characteristics is possible by flexible reduction of the intelligent system NDF relatively ODFS during some periods of the learning process. Usage of simplified structure models (reduced NDF systems - RDFS) and modified learning algorithms may provide better results, especially for unsufficient *a priori* information cases during earlier learning stage.

The principle of usage the RDFS instead of ODFS, at least during some learning phases, is called the Aggregative Learning Method. It provides a universal basis for various type model design and analysis. In this paper, the ALM is applied for training of multilayer neural networks, namely, the backpropagation neural networks (BPNN). Derived BPNN-model is used then for analysis of communication quality accordingly to a novel method proposed by the authors.

### 2.1 ALM for System Identification

**Vector General Parameter Approach.** Realization of the ALM for parameter identification problem is so-called General Parameter (GP) method, which has been considered in the preceding papers [3-5]. Recurrent procedures of identification are based on adjusting model schemes.

Assume that the plant is nonlinear on input variables:

$$x(t) = \tilde{a}^T \tilde{f}(t) + \eta(t), \quad (1)$$

where  $\tilde{a}$  : plant parameter vector of dimensionality N;  $\tilde{f}(t) = [\tilde{f}_1(t), \dots, \tilde{f}_m(t)]^T$  : vector function of input variables;  $\tilde{f}_i(t) = [f_{i1}(\tilde{u}), \dots, f_{ip_i}(\tilde{u})]^T$  : subvector function of input variables;  $\tilde{u}(t)$  : input vector;  $t = 0, 1, 2, \dots$ : discrete time;  $N \sum_{i=1}^m p_i = N$  : plant's dimensionality. GP-model of the plant (1) is of the form

$$\hat{x}(t) = (\hat{a}(0) + Q\tilde{\beta}(t-1))^T \tilde{f}(t), \quad (2)$$

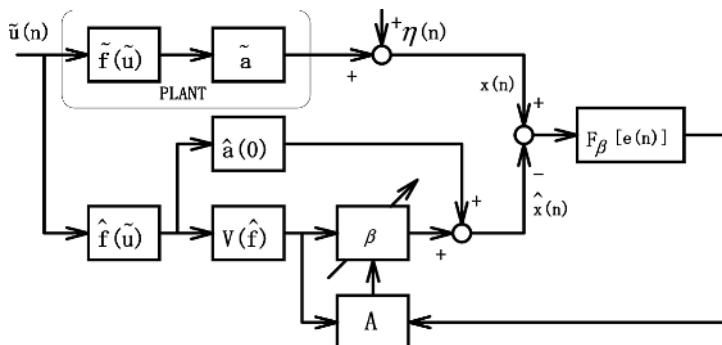
where  $\hat{x}(t)$  : model output;  $\hat{a}(0) = [\hat{a}_1(0), \dots, \hat{a}_m(0)]^T$  : initial model parameter vector;  $\hat{a}_i(0) = [\hat{a}_{i1}(0), \dots, \hat{a}_{ip_i}(0)]^T$  : i-th subvector of the initial model parameter vector;  $Q = \text{diag}(q_1 \tilde{I}_1, \dots, q_m \tilde{I}_m)$  :  $(N \times m)$ -matrix;  $\tilde{I}_i = [1 \dots 1]^T$  :  $(p_i \times 1)$ -vector of units;  $q_i$  : weight coefficients;  $\tilde{\beta}(t) = [\beta_1(t), \dots, \beta_m(t)]^T$  general parameter vector adjusted with algorithm

$$\tilde{\beta}(t) = \tilde{\beta}(t-1) + (x(t) - \hat{x}(t))\Gamma(t)Q^T \tilde{f}(t), \quad (3)$$

where  $\Gamma(t)$  :  $(m \times m)$ -matrix of gains,  $m$  : dimensionality of the GP-vector ( $1 \leq m < N$ ) (Fig.1).

It was shown that, for fixed values of signal/noise ratio, the quadratic form  $S(t) = E\{\Delta\tilde{\beta}^T(t)Q^T Q\Delta\tilde{\beta}(t)\}$  which is an expectation of the general parameter error vector  $\Delta\tilde{\beta}(t) = \tilde{\beta}(t) - C$ , where  $C$  is general parameter vector expectation at steady state (asymptotical value), may be considered as an identification accuracy measure. Besides that, the convergence rate (the rate of the value  $S(t)/S(0)$  decrease) depends on general parameter vector dimensionality  $m$ , not on model dimensionality  $N$ . Moreover, the convergence speed of the algorithm (3) is superior to conventional adjusting procedures. Therefore, the GP-model is a type of Reduced Number of Freedom System (RDFS) providing yet superior learning abilities.

**General Parameter Backpropagation Neural Networks (GP BPNN).** The GP method described in the preceding section was also extended to neural network models [4-6]. There is trade-off between network approximation ability and learning procedure convergence rate [1]. In this paper so-called Back-propagation neural networks (BPNN) are considered. They have been intensively used for data modeling, control and other tasks, which deal with high-dimensional, substantially nonlinear mapping. Let consider a one-layer sigmoid neural network described by equations



**Fig. 1.** Block Diagram of General Parameter Identification System

$$\begin{cases} x_i = \sum_{j=1}^p w_{ij} u_j \\ y_i = f(x_i) \\ f(x_i) = \frac{1}{1 + e^{-x_i}} \\ i = \overline{1, q} \end{cases}, \quad (4)$$

where  $u_j$  : input variables,  $j = \overline{1, p}$ ;  $p$  : input space dimensionality;  $y_i$  : output variables,  $i = \overline{1, q}$ ;  $q$  : the output space dimensionality or neuron units number;  $x_i$  : internal variables;  $w_{ij}$  : weights of network;  $f(x_i)$  : activation function. The standard backpropagation learning procedure for network (4) includes the following relations:

$$\begin{cases} e_i = y_i^* - y_i \\ w_{ij} = w_{ij} + \gamma_{ij} e_i f(x_i) [1 - f(x_i)] u_j \end{cases}, \quad (5)$$

where  $y_i^*$  : desired or sample value of network output;  $e_i$  : error signal of  $i$ -th output;  $\gamma_{ij}$  : weight adjusting algorithm gain.

The training procedure (4), (5) requires a long time period caused by large dimensionality of an adjusting parameter vector  $N = p \times q$ . It limits practical usage of such kind of networks, particularly for real time problems. One of the ways to cope with mentioned difficulty is the reduction of adjusting parameter space dimensionality at least during some of learning time periods by the general parameter approach implementation [4,5].

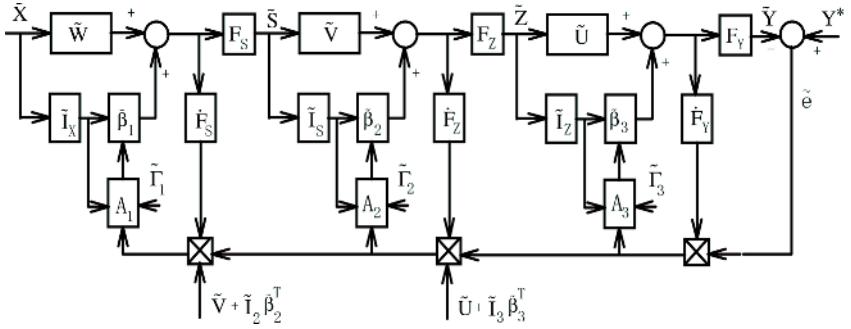
GP-version of the one-layer sigmoid neural network (4) is described by (6).

$$\begin{cases} \hat{x}_i = \sum_{j=1}^p (\hat{w}_{ij}^0 + \beta_i) u_j \\ y_i = f(\hat{x}_i) \\ i = \overline{1, q} \end{cases}, \quad (6)$$

where  $\hat{w}_{ij}^0$  : fixed initial values of network weights;  $\tilde{\beta} = [\beta_1 \beta_2 \dots \beta_q]^T$  : general parameter vector, estimated by algorithm (7) with constant gain  $\gamma_i$ .

$$\beta_i = \beta_i + \gamma_i e_i f(x_i) \times [1 - f(x_i)] \sum_{j=1}^p u_j, i = \overline{1, q}. \quad (7)$$

The developed approach may be implemented also with the multilayer sigmoid network structure as represented in Fig.2. The following designations are accepted:

**Fig. 2.** Three layer GP-BPNN

$\tilde{X}, \tilde{Y}$ : input and output vectors;  $\hat{S}, \tilde{S}, \hat{Z}, \tilde{Z}, \hat{Y}$ : internal variable vectors of the NN;  $\tilde{W}, \tilde{V}, \tilde{U}$ : weight matrixes;  $\tilde{I}_i = [1 \dots I]^T$ : units array of i-th layer input space dimensionality;  $\tilde{\beta}_i$ : general parameter vector of i-th layer;  $A_i$ : adjusting algorithms;  $F_s, F_z, F_y$ : nonlinear operators with  $F_s = \text{diag}\{f_i^s(\cdot)\}$ , where  $f_i(\cdot)$ : sigmoidal function from (4).

Recurrent procedure of general parameter adjusting in matrix form is

$$\begin{cases} \tilde{\beta}_1 = \tilde{\beta}_1 - \tilde{\Gamma}_1 \dot{F}_S (\tilde{V} + \tilde{I}_2 \tilde{\beta}_2^T) \dot{F}_Z (\tilde{U} + \tilde{I}_3 \tilde{\beta}_3^T) \dot{F}_Y \tilde{e} \tilde{X}^T \tilde{I}_X \\ \tilde{\beta}_2 = \tilde{\beta}_2 - \tilde{\Gamma}_2 \dot{F}_Z (\tilde{U} + \tilde{I}_3 \tilde{\beta}_3^T) \dot{F}_Y \tilde{e} \tilde{S}^T \tilde{I}_S \\ \tilde{\beta}_3 = \tilde{\beta}_3 - \tilde{\Gamma}_3 \dot{F}_Y \tilde{e} \tilde{Z}^T \tilde{I}_Z \end{cases}, \quad (8)$$

where  $\tilde{\beta}_i = [\beta_{i1} \beta_{i2} \dots \beta_{ij}]^T$ : i-th layer vector general parameter with dimensionality  $j$  equal to i-th layer unit number;  $\tilde{\Gamma}_i = \text{diag}\{\gamma_{ij}\}$ : gain matrix;  $\dot{F}_S = \text{diag}\{\dot{f}_j(S)\}$ : nonlinear matrix operator with elements corresponding to activation function derivatives;  $\tilde{e} = \tilde{Y}^* - \tilde{Y} = [e_1 e_2 \dots e_q]^T$ : output error vector;  $\tilde{I}_x, \tilde{I}_s, \tilde{I}_z, \tilde{I}_1, \tilde{I}_2, \tilde{I}_3$ : unit arrays of appropriate dimensionality.

It was shown in [4] that GP-BPNN features superior training and data classification abilities than conventional BPNN. In the paper the GP-BPNN was chosen for presentation of signal level data for the mobile communication network.

### 3 Monitoring of Communication Conditions

The developed method and algorithms have been applied for practical monitoring of the communication conditions during series of experiments carried out in 2002-2005. Conventional practical method for on-site monitoring of the network communication conditions is a measurement of the electromagnetic field level with specialized monitoring device. For stable communication it is required the field level to be higher than some fixed predetermined value. However, it is well-known fact that even a

strong signal from the base station in some cases couldn't guarantee the required stability because of specifics of radio wave propagation, environment influence, interference, etc. Therefore, conventional methods sometimes provide unreliable results or require high additional expenses. In such situation, ALM-based method for monitoring of communication conditions is proposed by the authors. It is briefly described below.

### 3.1 ALM Application for Time Series Analysis

Aggregative learning method with GP-BPNN is used for approximation of electromagnetic field level data from monitoring device and for estimation of communication quality. It is assumed that signal level data may be presented to desired accuracy by nonlinear time series (9).

$$x(n) = F[x(n-1), \dots, x(n-N)] + \eta(n), \quad (9)$$

where  $x(n)$ : signal level data time series;  $\eta(n)$ : noise;  $F[\cdot]$  is a plant operator;  $n=0,1,2,\dots$ : discrete time;  $N$ : plant dimensionality. The ALM-model of the process (10) is presented with GP-BPNN shown in Fig.2.

The proposed algorithm consists of the two following stages.

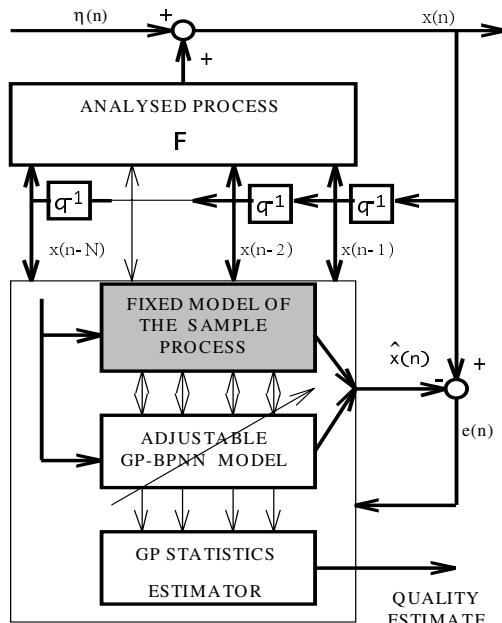
1) *Identification of a mathematical model corresponding to “ideal” communication conditions.* For this purpose, the field level data are registered in the spot where no interruptions occur during communication session. The data are used then for the model teaching procedure according to ALM. There are several parameters of the BPNN to be defined through learning. They include the number of the network units, and the values of the network weights. Along with these, optimal values of the learning algorithm parameters are to be determined, for example, an optimal value of the adjusting gains in the expression (8). On the identification stage, the general parameter expectation  $E_\beta$  and variance  $D_\beta$  (for scalar GP method) or the general parameter vector expectation and quadratic form  $S$  value (for vector GP method) are indicators of the current accuracy of the process model. The object of this stage is to construct a model of desired approximation accuracy. It corresponds to minimum values of  $E_\beta, D_\beta, S$ .

2) *Estimation of communication quality with the model identified during the first stage.* The developed model is used for calculation of predicted value of the electromagnetic field level. Analysis of the residual between the model (predicted) value and measured level data makes possible the estimation of communication quality. This phase is based on the inverse filtering procedure usage. The residual is used here for adjusting of network parameters, namely general parameters. It is important that other network parameters (number of units, weights) are fixed equal to those determined during the network training with the “ideal” sample data. Exclusively general parameters are adjusted using residual signal, and variations of general parameters and their statistics  $E_\beta, D_\beta, S$  are used as a measure for communication quality evaluation according to the expression (10), which is an

experimentally derived relationship between the values of  $E_\beta, D_\beta, S$  and the quality parameter  $Q$  (for example, the probability of stable communication).

$$Q = Q(E_\beta, D_\beta, S) \quad (10)$$

Block diagram of the communication quality estimation procedure with ALM is shown in Fig.3.

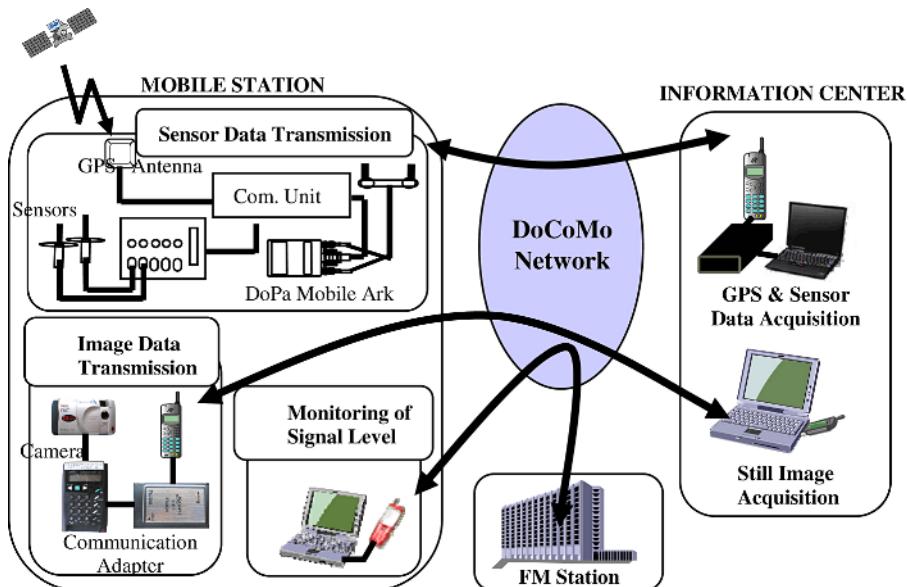


**Fig. 3.** ALM Approach for Communication Quality Estimation

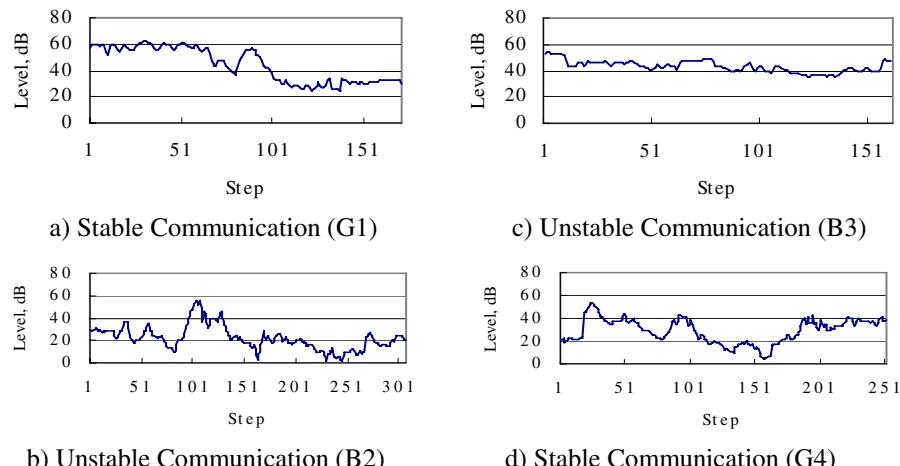
### 3.2 Outline of Experiment

Experimental setup is presented in Fig.4. There are three autonomous parts: 1) the operator-assisted system for measurement and registration of signal level data from a base station and direct audio monitoring of the voice communication quality, which used the signal from FM broadcasting station, 2) the monitoring system for packet communication (transmission of GPS data and sensor data), 3) the system for transmission of still images from digital camera in order to get visual confirmation information from the experimental site. All above systems are functioning simultaneously in order to create the database of related real-time data describing the state of the mobile network.

The quality data for voice communication and that for data transmission are registered in the binary form (1 – “success”, 0 – “failure”). Expert knowledge (operator skill) is used for the judgment. The gained data are used then as teaching data in the ALM procedure.

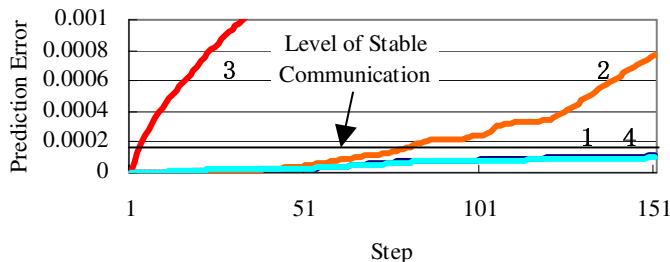


**Fig. 4.** Experiment Setup for Estimation of the Mobile Communication Quality



**Fig. 5.** Samples of Monitored Signal Level Data

In the paper, the monitoring problem of mobile network for voice communication is considered. Therefore, the signal level data are considered as input for ALM-learning system, and the “success-failure” data for packet communication are applied for ALM-model design.



**Fig. 6.** Results of Monitoring Data Analysis with ALM ("open-area" case) (2, 3 – unstable communication; 1, 4 – stable communication)

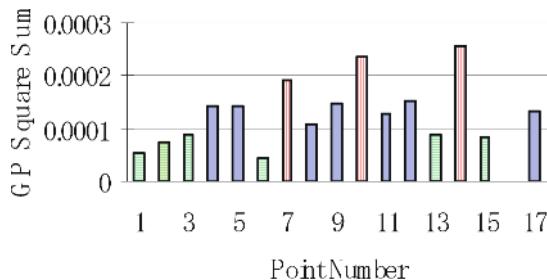
Let consider the learning procedure briefly. First, the signal level data are classified into groups depending on the communication quality, i.e. the communication stability (reliability). Second, "good" data are used for the ALM-learning, when the model structure and parameters should be defined. After the learning is finished, the model quality is verified using sample "good" and "bad" data. The trained and verified model is used then for real-time monitoring of communication conditions as follows.

Signal level data are measured on-line and fed into the model, a predicted value of the level is calculated and compared with real amount, which is measured one the next monitoring step. Based on the prediction error and GP statistics (variance, expectation), the judgment is made on the communication quality and reliability. Lastly, the set of calculated values of the quality index is plotted on the area map, which used for practical needs.

Two series of experiments have been held under different conditions. First of them was carried out at several "open-area" sites, and the second one aimed at "in-building" communication condition analysis. Typical waveforms for "open-area" measurements of signal level are shown in Fig. 5.

The curves in Fig.6 show the time change of ALM statistics, namely, the prediction error of ALM-model calculated for "normal" (1, 4) and "abnormal" (2, 3) data corresponding to stable (Fig.5a, 5d) and unstable (Fig.5b, 5c) data transmission respectively. It is seen that the behavior of the curves significantly different and the ALM prediction error value contents the information about communication quality. Moreover, in opposite to the conventional method of simple monitoring of the signal level, ALM provided clear discrimination of communication quality.

In Fig.7 the results of communication quality evaluation with ALM for "in-building" environment are shown. The signal level of the mobile network by DoCoMo, Inc. has been measured at several points of the 3<sup>rd</sup> floor of the Hokkaido Tokai University. The data from the point with the most stable communication, which characterized with minimum value of the quality criterion (No.16 in Fig.7), have been used for the GP-BP neural network training. The communication quality at other points was estimated both with the ALM and practically by direct usage of mobile phones. Comparison of the two sets of results approved a high reliability of the



**Fig. 7.** Results of Monitoring Data Analysis with ALM (“in-building” case)

proposed approach. In other words, ALM statistics may be considered as a criterion for real-time discrimination of communication conditions.

## 4 Conclusion

- 1) Combination of intelligent information processing approaches, such as the artificial neural networks, with advanced metrological methods and modern communication technology results in design of simple, efficient and economically reasonable systems.
- 2) The proposed Aggregative Learning Method could be useful for simple and reliable real-time surveillance and monitoring system design.

## References

1. Moody, T.J. and Darken, C.J.: Fast learning in networks of locally tuned processing units, *Neural Computation*, Vol.1, (1989) pp.151-160.
2. Castillo, O., Melin, P.: Hybrid intelligent systems for time series prediction using neural networks, fuzzy logic, and fractal theory. *IEEE Trans. Neural Networks*, 13(2002) 1395-1408.
3. Syzdykov, D. Zh., and Ashimov, A.A.: Identification of High Dimensional System by the General Parameter Method, *Prep .of the YIII IFAC Congress*, Kyoto, Japan, Vol.6, (1981) pp.32-37.
4. Akhmetov, D.F. et al.: System identification by the general parameter neural networks with fuzzy self-organization, in *Prep. of the 11th IFAC Symposium on System Identification (SYSID'97)*, Kitakyushu, Japan, Vol. 2, (1997) 829-834.
5. Akhmetov, D.F., and Dote, Y.: Fuzzy system identification with general parameter radial basis function neural network, In: *Analytical Issues in Fuzzy Control: Synthesis and Analysis*, S.Farinwata, D.Filev, and R.Langari, (Eds.), John Wiley & Sons, Ltd, United Kingdom, (2000) 73-92.
6. Akhmetov, D. et al: Data Monitoring and Analysis with Aggregative Learning Method, *Proc. of the Int. Conference on Computing, Communications and Control Technologies (CCCT 2004)*, Austin, Texas, USA, Vol.6, (2004) 234-239.

# Prediction of MPEG Video Source Traffic Using BiLinear Recurrent Neural Networks

Dong-Chul Park<sup>1</sup>, Chung Nguyen Tran<sup>1</sup>, Young-Soo Song<sup>1</sup>, and Yunsik Lee<sup>2</sup>

<sup>1</sup> ICRL, Dept. of Information Engineering, Myong Ji University, Korea  
`{parkd, tnchung, twobasehit}@mj.ac.kr`

<sup>2</sup> SoC Research Center, Korea Electronics Tech. Inst., Seongnam, Korea  
`leeys@keti.re.kr`

**Abstract.** A prediction scheme for the MPEG video traffic in ATM networks using a BiLinear Recurrent Neural Network (BLRNN) is proposed in this paper. Since the BLRNN is based on the bilinear polynomial, it has been successfully used in modeling highly nonlinear systems with time-series characteristics, and the BLRNN can be a natural choice in predicting the MPEG video traffic with a bursty nature in the ATM networks. The proposed BLRNN-based predictor is applied to MPEG-1 and MPEG-4 video traffic data. The performance of the proposed BLRNN-based predictor is evaluated and compared with the conventional MultiLayer Perceptron Type Neural Network (MLPNN)-based predictor. When compared with the MLPNN-based predictor, the proposed BLRNN-based predictor shows 27%-51% improvement in terms of the Relative Mean Square Error (RMSE) criterion.

## 1 Introduction

In recent years, multimedia services such as video conferencing, video on demand, and audio through dynamic wireless ATM networks have been developing rapidly. One of the major problems in wireless ATM networks is the limited resources and quality of service (QoS). Therefore, the demand for bandwidth allocation and QoS control are expected to increase. However, the network bandwidth allocation and management, as well as the traffic control have become more complicated than ever before.

To tackle the bandwidth allocation problem, two approaches have been proposed. The first approach uses a fixed or static bandwidth allocation. At the beginning of a request, the admission control is performed based on the traffic descriptors declared by the user and the QoS requests. After the request is accepted, the user is allocated with a fixed bandwidth for the duration of the request process. However, this approach is only suitable for prerecorded video applications. Therefore, a second approach using a dynamic bandwidth allocation has been proposed for real-time applications. One of the solutions used for dynamic bandwidth allocation and QoS is to predict the near future traffic according to the previously observed network traffic and take appropriate actions

such as controlling buffer sizes, to optimize the bandwidth allocation. Accurate traffic prediction is, therefore, critical in order to utilize the resources efficiently.

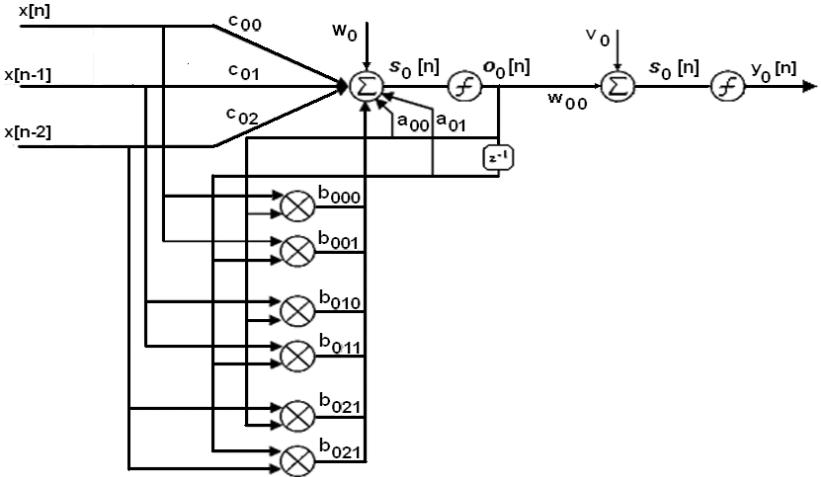
Several models have been proposed to predict the video source traffic. Nomura *et al.* [1] proposed a model based on the autoregressive (AR) processes. This model simulates the characteristics of a single video source that can capture the autocorrelation properties of the video signal. The AR model has been shown to be very effective in predicting video signals with high correlations. In [2], Adas proposed an adaptive linear predictor for video source traffic prediction that also considered traffic variation prediction errors. From the Adas' work, Yoo proposed a new adaptive linear predictor that utilized the analysis of scene changes[3]. These two approaches provided a methodology to predict the sizes of the I-frames and P-frames. However, these models based on a linear analysis may not be suitable for MPEG video traffic with bursty characteristics.

Various approaches based on neural networks have been proposed for MPEG video traffic prediction [4,5,6]. Chang *et al.* proposed a Pipelined Recurrent Neural Network (PRNN) predictor for MPEG video traffic in ATM networks [4]. The pipelined architecture was employed to overcome the simplicity of the nonlinear autoregressive moving average (NARMA) model and to guarantee fast learning. However, its architecture is very complicated and the performance of the NARMA model itself is not suitable for practical problems. The approach presented by Doulamis *et al.* [5] proposed an adaptable neural network architecture that included online and offline traffic modeling to predict the MPEG-2 video source traffic. Bhattacharya *et al.* [6] proposed a recurrent neural network based predictor that showed a better performance than the previous mentioned predictors. In this paper, a nonlinear scheme using a recurrent neural network based on a bilinear polynomial called BiLinear Recurrent Neural Network (BLRNN)-based predictor is proposed. The bilinear polynomial model is used widely in modeling time-series data. The neural network based on the bilinear polynomial model BLRNN, introduced by Park and Zhu [7], is applied to predict MPEG video traffic.

The remainder of this paper is organized as follows: Section 2 gives a brief review of the BLRNN and its training algorithm. The application of the BLRNN to predict video source traffic is presented in Section 3. Section 4 presents some experiments and results including a performance comparison of the proposed BLRNN-based predictor and the conventional MLPNN-based predictor. The conclusion and some further remarks are given in Section 5.

## 2 BiLinear Recurrent Neural Networks

Artificial neural networks have been used for nonlinear system identification for adaptive control like time series. Among various neural network architectures, the high order neural network has advantages over linear first order feed-forward type neural network. It has better computational, storage, pattern recognition, and learning properties than the conventional neural networks. However, to generate a complicated nonlinear decision surface from the first



**Fig. 1.** Simple BLRNN with structure 3-1-1 and 2 recursion lines

order neural network, it requires complicated structure while the higher order neural network can implement nonlinear decision surface with fewer hidden neurons. But, one of the disadvantages for the higher order neural network is that the complexity of the model can be very high even for moderate values of dimension [7,9,10]. However, a recurrent neural network that utilizes linear input and first order recurrent term only fails to utilize the high order terms of inputs while the high order neural network fails to utilize the history of recurrent terms. It encourages us to find a new hidden neuron expression which can utilize all three kinds of information. The BLRNN that adopts the bilinear polynomial and recurrent terms is an alternative to the high order neural networks [7].

Fig. 1 shows an example of a BLRNN with one hidden layer. The output value of a bilinear recurrent neuron is computed by the following equations:

$$\begin{aligned}
 s_p[n] &= w_p + \sum_{k_2=0}^{N_f-1} a_{pk_2} o_p[n - k_2] \\
 &+ \sum_{k_1=0}^{N_i-1} \sum_{k_2=0}^{N_f-1} b_{pk_1k_2} x[n - k_1] o_p[n - k_2] \\
 &+ \sum_{k_1=0}^{N_i-1} c_{pk_1} x[n - k_2] \\
 &= w_p + A_p^T Z_p[n] + Z_p[n] B_p^T X[n] + C_p^T X[n]
 \end{aligned} \tag{1}$$

where  $w_p$  is the weight of the bias neuron for the  $p - th$  hidden neuron,  $p = 1, 2, \dots, N_h$ .  $N_h$ ,  $N_i$ , and  $N_f$  are the number of hidden neurons, input neurons, and feedback lines, respectively.  $A_p$  is the weight vector for recurrent portion,

$B_p$  is the weight matrix for the bilinear recurrent portion, and  $C_p$  is the weight vector for the feed-forward portion. T represents the transpose of a matrix.

The weight adaption can be summarized as follows:

$$w_{ij}^{new} = w_{ij}^{old} + \eta \sum_{n=1}^{N_{data}} \delta_l[n] o_p[n] \quad (2)$$

$$C_p^{new} = C_p^{old} + \eta \sum_{n=1}^{N_{data}} \delta_p[n] X[n] \quad (3)$$

$$A_p^{new} = A_p^{old} + \eta \sum_{n=1}^{N_{data}} \delta_p[n] Z[n] \quad (4)$$

$$B_p^{new} = B_p^{old} + \eta \sum_{n=1}^{N_{data}} \delta_p[n] Z[n] X[n]^T \quad (5)$$

where  $\delta_l[n] = (d_l[n] - o_l[n])\phi'(s_l[n])$  and  $\delta_p[n] = \sum_{l=0}^{N_o} \delta_l[n] w_{lp} \phi'(s_p[n])$ .

More detailed information on the BLRNN can be found in [7,10,11].

### 3 Video Source Traffic Prediction

#### 3.1 MPEG Source Characteristics

In MPEG-1, three main types of frames are defined. Intra frames (I-frames) are the frames that are coded without any reference to other frames. Predictive frames (P-frames) are frames that are coded with reference to previous I-frames or P-frames. Bidirectionally predictive frames (B-frames) are frames that are coded with reference to previous I-frames or P-frames, as well as future I-frames or P-frames. With three different types of frames, the coding algorithm gives a highly efficient compression. An I-frame is typically used as a reference for creating P-frames and B-frames. A complete video clip is represented by a group of frames or a group of pictures (GOP). An MPEG-1 bit stream consists of a repeating GOP structure. A typical GOP structure is defined by two parameters,  $M$  and  $N$ , where  $M$  is the distance between two I-frames, while  $N$  is the distance between two P-frames. One example of a GOP sequence is *IBBPBBPBBPB*. In this example,  $M = 12$  and  $N = 3$ .

In 1998, the MPEG-4 International Standard was first introduced. It was a new coding algorithm to encode video and audio using a much lower bit rate than MPEG-1 and MPEG-2. In MPEG-4, an image or frame is represented by a visual object plane (VOP). The term GOP in MPEG-1 and MPEG-2 was replaced by a group of video object plane (GOV).

#### 3.2 Traffic Prediction Models

Since each frame type has its own traffic characteristics, it is suggested that each frame type (I, P, or B) should be modeled individually. Bhattacharya *et. al.*

[6] analyzed the input parameters and suggested to add some additional input parameters such as the difference in frame sizes between consecutive frames, to improve the prediction accuracy. In this paper, we propose three kinds of predictors with different input parameters: one each for I-frame, P-frame, and B-frame.

**I-frame Predictor.** The inputs for the I-frame predictor are the frame sizes of the four previous I-frames. To capture the variations of the I-frame sizes, the difference in frame size between two consecutive I-frames and the second derivation values of the consecutive I-frames are included in the I-frame predictor as additional inputs. The difference in frame sizes between two consecutive I-frames is defined as:

$$\Delta I(i) = x(i) - x(i-1) \quad (6)$$

The second derivation of the I-frame sizes is defined as:

$$\Delta^2 I(i) = \Delta x(i) - \Delta x(i-1) = x(i) - 2x(i-1) + x(i-2) \quad (7)$$

where  $i$  is the index of the I-frames in the sequences. From the four previous I-frame sizes, we can define the three differences in frame sizes and the three second derivations. Therefore, the total number of neurons for the input layer of the I-frame predictor used in our experiments is 10.

**P-frame Predictor.** Similar to the I-frame predictor, the inputs of the P-frame predictor are the frame sizes of the four previous P-frames. In the case of a P-frame subsequence, at the point of the scene change, the generated bit rate increases rapidly due to low temporal correlations; whereas inside the scene, the generated traffic is also smooth. In order to obtain an accurate prediction, the scene change should be involved in the input parameters. To overcome this problem, the difference in frame sizes between two consecutive P-frames or a consecutive P-frame and I-frame must be added to the inputs of the P-frame predictor. The difference in frame sizes between the two consecutive P-frames or a consecutive P-frame and I-frame is defined as:

$$\Delta P(i) = x(i) - x(i-1) \quad (8)$$

where  $i$  is the index of the I-frames or P-frames in sequence. Therefore, with four previous P-frame sizes and four differences in frame sizes between two consecutive P-frames or a consecutive P-frame and I-frame, the total number of neurons in the input layer of the P-frame predictor used in our experiments is 8.

**B-frame Predictor.** The main inputs of the B-frame predictor are similar to those used in the previous I-frame and P-frame predictors. They are the frame sizes of the four previous B-frames. As mentioned above, B-frames use a bidirectional predictive coding scheme, therefore, it is logical to include the I-frame sizes and P-frame sizes to predict the B-frame sizes. In order to capture the variations of the B-frame sizes, not only the difference in frame sizes between two

consecutive B-frames themselves, but also the difference in frame sizes between two consecutive I-frames and P-frames should be included in the inputs of the B-frame predictor.

$$\Delta B(i) = x(i) - x(i-1) \quad (9)$$

where  $i$  is the index of the I-frames, P-frames, or B-frames in the sequence. Therefore, with the four previous B-frame sizes, three previous I-frame sizes, three previous P-frame sizes, and three differences in frame sizes between two consecutive B-frames, the total number of neurons in the input layer of the B-frame predictor used in our experiments is 13.

From the inputs described above, three different predictors, one each for the I-frames, P-frames, and B-frames were created. To simplify the model, we chose a network with one hidden layer. The number of neurons in the hidden layer was selected through experiments. All predictors had one neuron in the output layer. The number of recursion lines was only 2 or 3 to reduce the computational costs.

## 4 Experiments and Results

For experiments, three predictors of video source traffic for I-frames, P-frames, and B-frames were implemented. For all experiments, we used an I-frame predictor with a structure of 10-6-2-1, a P-frame predictor with a structure of 8-6-2-1, and a B-frame predictor with structure of 13-7-3-1; the first index is the number of input neurons, the second index is the number of hidden neurons, the third index is the number of recursive lines, and the last index is the number of output neurons. In order to measure the performance of prediction algorithms, the Relative Mean Square Error (RMSE) was employed.

### 4.1 Experiments on MPEG-1 Video Traces

In this experiment, the proposed BLRNN-based predictor was evaluated and compared with the conventional MLPNN-based predictor using MPEG-1 video traffic data. All implemented predictors were trained and tested on MPEG video traces obtained from [12]. These MPEG-1 video traces were provided by the Institute of Computer Science III, the University of Wuerzburg, Wuerzburg, Germany. For more details on these traces, documents can be found at [12]. In this experiment, we chose the following traces for training and testing the implemented predictors: *Star Wars*, *Talk Show*, *Silence of the Lambs*, *Mr. Bean*, *Terminator*, *MTV* and *Simpsons*.

In order to evaluate the performance and test the robustness of our proposed BLRNN-based predictor, we only used segments of *Star Wars* from these sequences for training and validation. After training, the proposed BLRNN-based predictor was evaluated and compared with the conventional MLPNN-based predictor in the remaining segments of *Star Wars* and the remaining trace sequences.

**Table 1.** Comparative results of the proposed BLRNN-based predictor with the conventional MLPNN-based predictor in terms of RMSE on MPEG-1 data

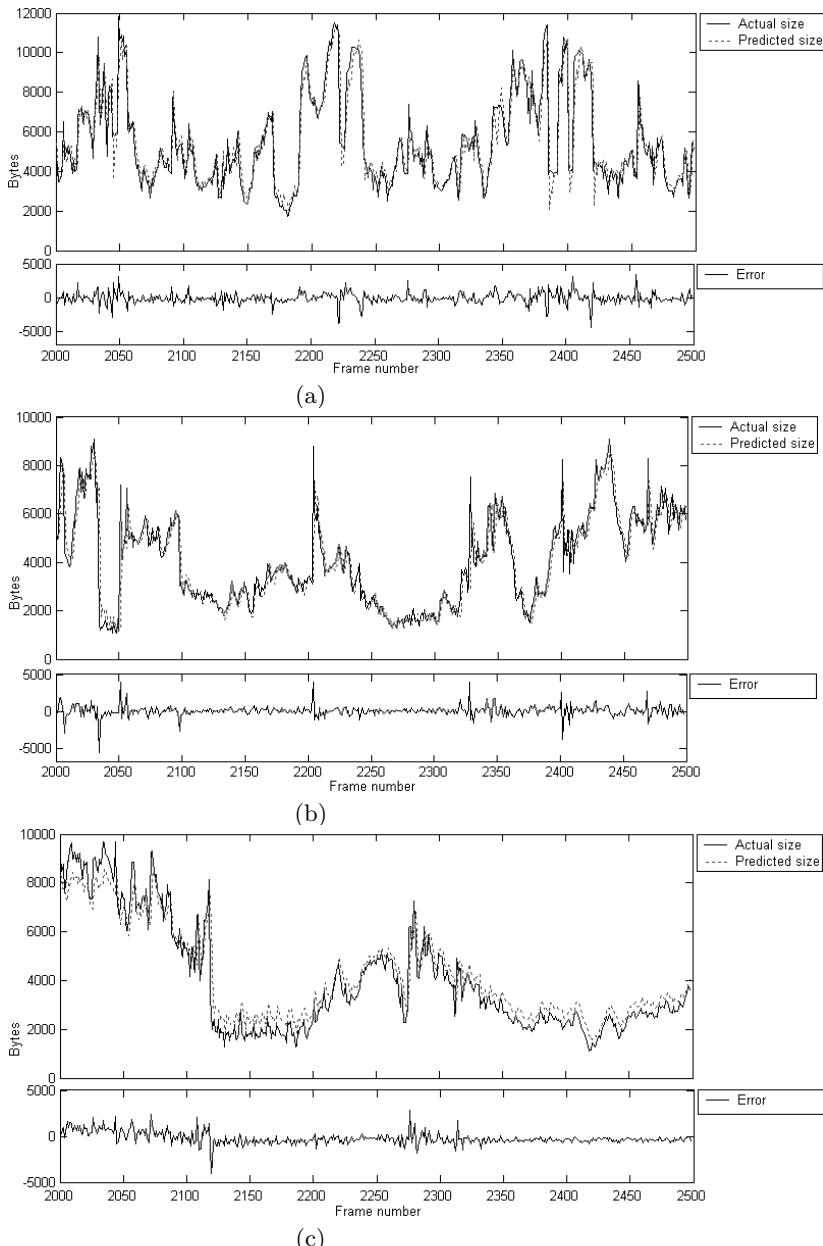
Trace	Type	MLPNN	BLRNN
<i>Star Wars</i>	I	0.024	0.015
	P	0.277	0.220
	B	0.295	0.134
<i>Talk Show</i>	I	0.023	0.007
	P	0.136	0.116
	B	0.050	0.035
<i>Silence of the Lambs</i>	I	0.025	0.021
	P	0.281	0.228
	B	0.565	0.132
<i>Mr. Bean</i>	I	0.031	0.011
	P	0.152	0.106
	B	0.101	0.051
<i>Terminator</i>	I	0.025	0.023
	P	0.255	0.132
	B	0.164	0.139
<i>MTV</i>	I	0.054	0.051
	P	0.229	0.131
	B	0.155	0.143
<i>Simpsons</i>	I	0.038	0.021
	P	0.238	0.220
	B	0.181	0.172
Average	I	0.031	0.021
	P	0.224	0.164
	B	0.215	0.115

The results in Table 1 show that the performance of the proposed BLRNN-based predictor improves significantly when compared with the conventional MLPNN-based predictor. The average improvements for the I-frame, P-frame, and B-frame predictors were 32%, 27%, and 47%, respectively.

## 4.2 Experiments on MPEG-4 Video Traces

In this experiment, the performance of the proposed BLRNN-based predictor was evaluated and compared with the conventional MLPNN-based predictor using MPEG-4 video traffic data. All the implemented predictors were trained and tested on the video traces obtained from [13]. These are the MPEG-4 video traces used for network performance evaluations. These video traces were provided by the Telecommunication Networks Group of the Technical University of Berlin, Berlin, Germany. The technical report available at [14] gives the details of the procedures used in generating the video traffic traces. In this experiment, we chose the following sequences for training and testing all the predictors:

*Mr. Bean, Jurassic Park I, Star Wars, Die Hard III, Lecture Room, Silence of the Lambs, and Skiing.*



**Fig. 2.** Actual and predicted video traffic using the proposed BLRNN-based predictor of 500 frames of *Skiing* sequence for: (a) I-frame., (b) P-frame, and (c) B-frame

From these trace sequences, we only used segments of the *Skiing* for training and validation. After training, all predictors were tested on the remaining segments of the *Skiing* sequence and the remaining sequences from above.

**Table 2.** Comparative results of the proposed BLRNN-based predictor with the conventional MLPNN-based predictor in terms of RMSE on MPEG-4 data

Type	Avg. RMSE	
	MLPNN	BLRNN
I	0.026	0.014
P	0.123	0.073
B	0.106	0.052

Fig. 2-(a) shows the prediction results for the I-frames. The figure consists of 500 samples from the frame number 2,000 to the frame number 2,500. The continuous curve is the actual I-frame sizes while the dashed curve is the predicted sizes of the I-frames. Similarly, Figs. 2-(b) and (c) show the prediction results for P-frames and B-frames, respectively.

Table 2 summarizes the comparative results of the proposed BLRNN-based predictor and the conventional MLPNN-based predictor in terms of the RMSE. The results show that the proposed BLRNN-based predictor significantly outperforms the conventional MLPNN-based predictor. The average improvement for the I-frame, P-frame, and B-frame were 46%, 41%, and 51%, respectively.

## 5 Conclusions

In this paper, a BLRNN-based predictor is proposed for predicting MPEG real-time video source traffic. The main feature of the BLRNN is that it has both the ability to approximate arbitrary nonlinear mapping just like the MLPNN and to simulate the highly bursty characteristics of video bit streams. This research provides an efficient method for dynamic bandwidth management and QoS. The proposed predictor was evaluated and compared with the conventional MLPNN-based predictor with both MPEG-1 and MPEG-4 data. The simulation results showed that the proposed BLRNN-based predictor provides accurate real-time predictions for MPEG video traffic and significant improvements when compared with the conventional MLPNN-based predictor.

## References

1. Nomura, N., FujiiT., Ohta, N.: Basic Characteristics of Variable Rate Video Coding in ATM Environment. IEEE J. Select. Areas Commun., Vol. 7 (1989) 752-760
2. Adas, A.M.: Using Adaptive Linear Prediction to Support Real-Time VBR Video under RCBR Network Service Model. IEEE/ACM Trans. on Networking 6(5) (1998) 635-644
3. Yoo, S.J.: Efficient Traffic Prediction Scheme for Real-Time VBR MPEG Video Transmission over High-Speed Networks. IEEE Trans. on Broadcasting 48(1) (2002) 10-18
4. Chang, P.R., Hu, J.T.: Optimal Nonlinear Adaptive Prediction and Modeling of MPEG Video in ATM Networks using Pipelined Recurrent Neural Networks. IEEE J. Selected Areas Commun., Vol. 15 (1997) 1087-1100

5. Doulamis, A.D., Doulamis, N.D., Kollias, S.D.: An Adaptable Neural Network Model for Recursive Nonlinear Traffic Prediction and Modeling of MPEG Video Sources. *IEEE Trans. on Neural Networks* 14(1) (2003) 150-166
6. Bhattacharya, A., Parlors, A.G., Atiya, A.F.: Prediction of MPEG-Coded Video Source Traffic using Recurrent Neural Networks. *IEEE Trans. on Acoustics, Speech, and Signal Processing* 51(8) (2003) 2177-2190
7. Park, D.C., Zhu, Y.: Bilinear Recurrent Neural Network. *IEEE ICNN*, Vol. 3 (1994) 1459-1464
8. Xian, Z., Le-Ngoc, T.: Polynomial Perceptrons and Their Applications to Fading Channel Equalization and Co-Channel Interference Suppression. *IEEE Trans. on Signal Processing* 42(9) (1994) 2470-2479
9. Connor, J., Martin, R., Altas, L.: Recurrent Neural Networks and Robust Time Series Prediction. *IEEE Trans. on Neural Networks* 5(2) (1995) 240-254
10. Park, D.C., Park, T., Choi, S.: Short-Term Electric Load Forecasting using Recurrent Neural Network. *Proc. of ISAP'97*, (1997) 367-371
11. Park, D.C., Jeong, T.K.: Complex Bilinear Recurrent Neural Network for Equalization of a Satellite Channel. *IEEE Trans. on Neural Networks* 13(3) (2002) 711-725
12. Rose, O.: Index of /MPEG. Univ. Wurzburg, Wurzburg, Germany (1995). Available:<http://www3.informatik.uni-wuerzburg.de/MPEG/>
13. Fitzek, F.H.P., Reisslein,M.: MPEG-4 and H.263 Video Traces for Network Performance Evaluation (2001). Available: <http://www-tkn.ee.tu-berlin.de/research/trace/trace.html>.
14. Fitzek, F.H.P., Reisslein,M.: MPEG-4 and H.263 Video Traces for Network Performance Evaluation. Telecommunication Networks Group, Technical University of Berlin, Berlin, Germany, TKN Tech. Rep., (2000)

# Dynamic Neural Network-Based Fault Diagnosis for Attitude Control Subsystem of a Satellite

Z.Q. Li<sup>1</sup>, L. Ma<sup>1,2</sup>, and K. Khorasani<sup>1</sup>

<sup>1</sup> Department of Electrical and Computer Engineering  
Concordia University,

Montreal, Quebec H3G 1M8 Canada

<sup>2</sup> Department of Applied Computer Science  
Tokyo Polytechnic University  
Atsugi, Kanagawa, Japan  
kash@ece.concordia.ca

**Abstract.** The objective of this paper is to develop a dynamic neural network scheme for fault detection and isolation (FDI) in the reaction wheels of a satellite. The goal is to decide whether a bus voltage fault, a current loss fault or a temperature fault has occurred in one of the three reaction wheels and further to localize which wheel is faulty. In order to achieve these objectives, three dynamic neural networks are introduced to model the dynamics of the wheels on all three axes independently. Due to the dynamic property of the wheel, the architecture utilized is the Elman recurrent network with backpropagation learning algorithm. The effectiveness of this neural network-based FDI scheme is investigated and a comparative study is conducted with the performance of a generalized observer-based scheme. The simulation results have demonstrated the advantages of the proposed neural network-based method.

## 1 Introduction

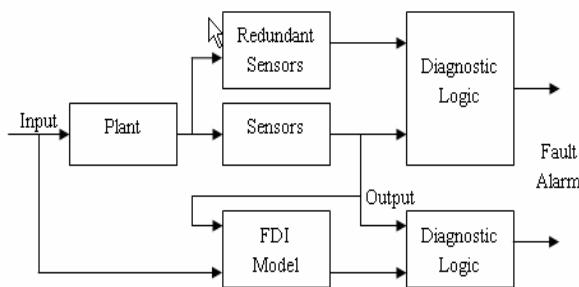
The attitude control of spacecraft has been widely studied since the late 1950s [1-2]. The attitude control subsystem stabilizes the spacecraft and orients it in the desired set point position in short time despite the presence of external disturbance torques. The control torques could be formed from combination of momentum wheels, reaction wheels, control moment gyros, thrusters or magnetic torquers. Normally, there are four reaction wheels (3 active and 1 redundant) on the spacecraft to be controlled. Each active reaction wheel is aligned with one of the body axis of spacecraft, it can rotate in either direction and provide reaction torque for the related axis control. The redundant one will be engaged in case any of the three wheels fails.

Any fault that occurs in the wheels should be detected and isolated as early as possible to avoid serious damage to the attitude control subsystem of the satellite. Normally, there are three types of faults in a wheel that deserve special attention. The first is the bus voltage fault. The bus voltage should be sufficiently high to avoid elimination of the voltage headroom. Low bus voltage will result in reduced torque capacity and will seriously cause instability of the attitude of the satellite. The same

effect will happen when the motor current loss occurs in the wheel. This will result in loss of power, and consequently the wheel cannot supply enough reaction torque to achieve proper set point change of the attitude. Finally, the temperature change is the third source of fault. The temperature is highly related to the viscous friction, which is the main friction factor for the wheel. The temperature fault will cause the wheel to operate in abnormal operating condition. A high fidelity mathematical model of a reaction wheel [3] is discussed briefly in Section 2.

Generally, to detect and isolate faults in a system, some sort of redundancy is required. A traditional approach to accomplish the fault diagnosis and isolation (FDI) problem is based on hardware or physical redundancy that uses multiple sensors, actuators, components to measure and control a particular variable. The redundancy is used to make consistency checks between related variables. Typically, a voting technique is applied to hardware redundancy system to decide whether a fault has occurred or not. This kind of method is very reliable and widely used in many practical industries. But the main disadvantages of hardware redundancy are the requirements of extra equipments, maintenance cost and the additional space to accommodate the redundant equipments [4].

Recently, much research has been made on the development of analytical or functional redundancy methods. In analytical redundancy system, one utilizes more (not necessarily identical) ways to determine a variable, where one way use a mathematical process model in analytical form. Based on analytical redundancy, there exist a large number of methods known as model-based fault diagnosis and isolation (FDI). Figure 1.1 illustrates the concepts of hardware and analytical redundancy.

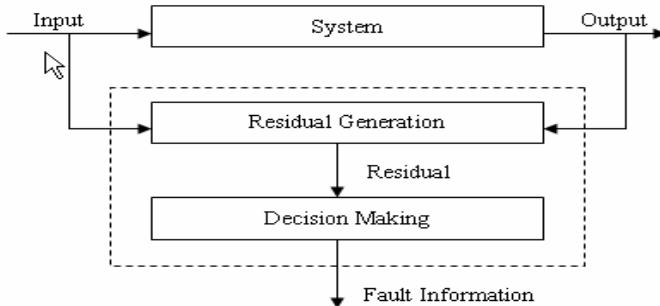


**Fig. 1.1.** Comparison between hardware and analytical redundancy schemes

Model-based FDI methods can be divided into three main subcategories according to the literature [7-9,13,14]. They are observer-based approaches, parity vector (relation) methods and parameter estimation methods. All these model-based FDI methods share some common features in that they usually comprise of two main stages as illustrated in Figure 1.2.

This two-stage structure was first suggested by Chow and Willsky (1980). The first stage is the residual generation. Its purpose is to use the available input and output information of the monitored system to generate the indicated fault signal, termed residual. In ideal system healthy cases, the residual should be normally zero or close to zero, and should be distinguishably different from zero when a fault occurs in the

system. This stage of residual generation identifies representative fault symptoms from the system and these symptoms are carried by the residual. After the residual is generated, it will be processed by the second stage (decision-making stage) for fault likelihood examination and finally a determination about the fault occurrence is made.



**Fig. 1.2.** Conceptual structure of model-based FDI

Other than hardware redundancy and model-based methods, there are a number of methods based on intelligent and learning-based strategies [5]. These methods make use of a large amount of process history data. Neural networks and fuzzy logic techniques have been investigated as powerful modeling and decision making tools. These methods have the potential to “learn” the plant model from input-output data or “learn” fault knowledge from past experiences, as they can be used as function approximators to construct the analytical model for residual generation, or as supervisory schemes to make the fault analysis decisions [6]. The nonlinear modeling ability of neural networks has already been utilized for nonlinear fault diagnosis [7-8].

In this paper, neural network is employed for reaction wheel of each axis so as to observe the estimated reaction torques from each wheel. Through these estimated signals, one will be able to detect the existence of a fault in the system as well as the wheel that is faulty. With the help of neural networks employed with the reaction wheel of each axis, we can observe the estimated reaction torque from each wheel. Figure 1.3 shows the stages performed.

As shown in Figure 1.3, the algorithm developed consists of three stages:

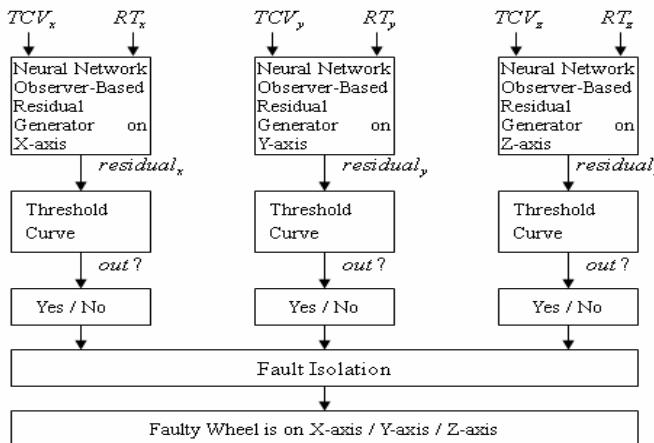
1. Residual Signals Generation: We design three Elman networks [12] to model the dynamics of the reaction wheel on three axes separately. The inputs to each network are the torque command voltage ( $TCV$ ) and the one step delay of the actual reaction torque ( $RT$ ) of the corresponding wheel or the estimated reaction torque from the output of network. The output of the network is the estimated reaction torque. Three residual signals are generated in this stage.
2. Threshold Testing and Fault Detection: The residual signals generated in the first stage will be processed through the corresponding threshold curves for testing. The threshold curves are generated through analyzing a large number of residual curves collected in fault free cases. If the residual curve exceeds the threshold curve for a considerable time, then we consider that there is a fault in the concerned wheel.

- Fault Isolation: With consideration on the threshold testing results from all the three axes, one can also localize which wheel is faulty.

The outline of the remainder of this paper is as follows. In Section 2, a brief review of the attitude control system and model of the reaction wheel will be given. Section 3 presents briefly the generalized Luenberger observer-based scheme used for fault detection and isolation. This approach will serve as benchmark for comparison with the neural network FDI scheme. In Section 4, a dynamic neural network observer-based FDI scheme will be developed. The suitable choice of the network parameters will be discussed. A comparative study between neural network and observer-based FDI schemes will be conducted in Section 5. These comparative results will demonstrate the advantages of neural network observer-based scheme developed in this paper.

## 2 Reaction Wheel Model

A hypothetical satellite is introduced in this section. The satellite is launched into a 700 km circular Low Earth Orbit (LEO), sun-synchronous ( $98.2^\circ$ ) orbit. With a velocity of approximately 7.5 km/s, the orbit has a period of 98.8 minutes. Orbit selection is driven by science requirements, orbit lifetime, ground station coverage, and radiation concerns. We introduce three axis control techniques for attitude control. Specifically, we achieve pointing accuracy for each axis through the reaction wheel aligned with each axis separately. The accuracy requirement of Earth-pointing attitude is within  $0.2^\circ$  in all the three axes. To achieve this, three separate PID control loops are used to control the wheels for each satellite axis.



**Fig. 1.3.** The FDI Methodology by Using Neural Networks

The standard satellite attitude control loop contains four main components: sensor, controller, actuator and body dynamics. Noises are added on the sensors and external

disturbances are imposed on the satellite. The actuators we use for attitude control consist of 3 active reaction wheels on the three axes of the satellite and one additional redundant wheel. Reaction wheels are momentum exchange devices which provide reaction torque to a satellite and store angular momentum [1-3]. Mathematical model of the reaction wheel can be derived merely from an application of Newton's laws, but additional terms are included herein to assess performance beyond the normal speed range, and as a function of temperature and bus voltage. Furthermore, disturbance and noise terms cannot be ignored in reality. A reaction wheel consists of a rotating flywheel, typically suspended on ball bearings, and driven by an inertial brushless DC motor. A detailed block diagram of a typical reaction wheel is shown in Figure 2.1. This diagram provides the fundamental relationships for a high fidelity mathematical model of a reaction wheel system [3].

Operating in space a satellite experiences many types of external environmental disturbance torques. Four main disturbances we consider here are: gravitation torque, solar pressure torque, magnetic torque and aerodynamic torque. It is assumed that the maximum external disturbance torque is the sum of these four maximum torques:

$$DIS = DIS_{gg} + DIS_{sp} + DIS_{mf} + DIS_{ad} = 5.68 \times 10^{-5} N\cdot m$$

We assume the external disturbance torque is a normally distributed random signal with zero mean and variance  $DIS^2 = (5.68 \times 10^{-5})^2$ .

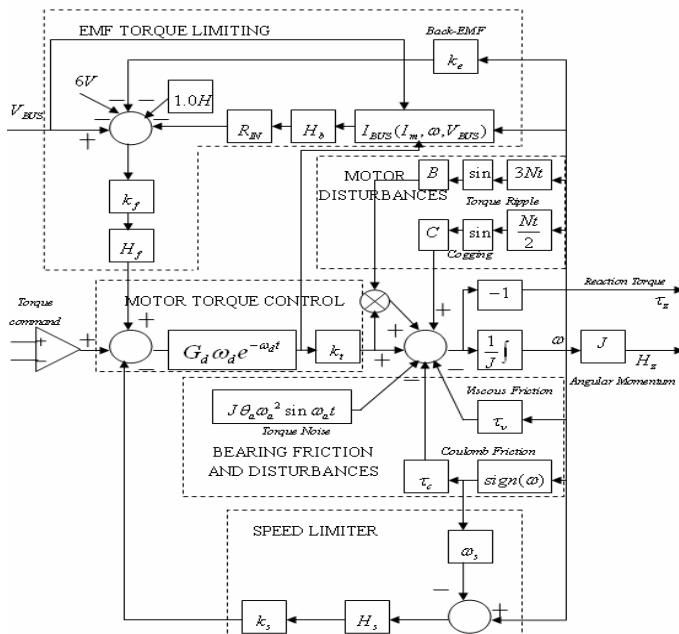


Fig. 2.1. Detailed reaction wheel block diagram

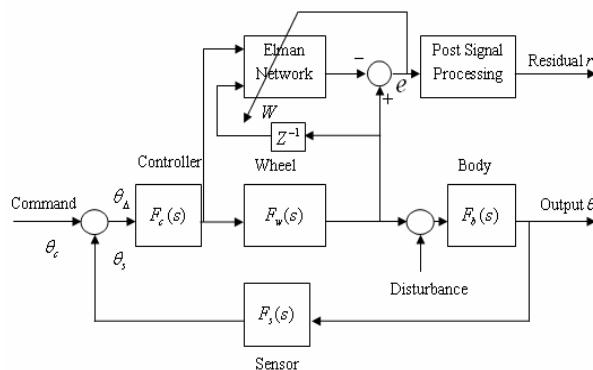
### 3 Observer-Based Fault Detection and Isolation

When the normal operating value of the bus voltage that is  $8V$  drops, the motor torque may be limited at high speeds due to the increasing back-EMF of the motor. This eventually results in reduced torque capacity of the wheel. When this value becomes too low (e.g.  $4V$ ), the attitude control system will malfunction and the attitude of the spacecraft becomes *unstable*. Similarly, since the motor torque is directly related to the motor current through a constant parameter  $k_t$ , when some kind of motor current loss occurs in the reaction wheel, the motor torque will decrease and drop accordingly. Therefore, the wheel can no longer supply enough motor torque for the attitude control system. When the current loss becomes significant, the controlled attitude *will* become unstable.

It is well-known that viscous friction is present in the bearings due to the bearing lubricant. Through this friction, one can estimate the working condition of the bearings. When the bearings are damaged seriously, this viscous friction becomes much larger than that in normal conditions. Since the temperature of the wheel is strongly related to the viscous friction in the wheel, this suggests that an estimate of the working condition of the bearings is possible through monitoring the temperature of the wheel. Observers are designed to estimate the output of the actuators, that is, the reaction torques generated by the wheels. Three independent observers are designed one for each wheel of the three axes. In this way, one can observe the values of reaction torques generated by each wheel at the same time.

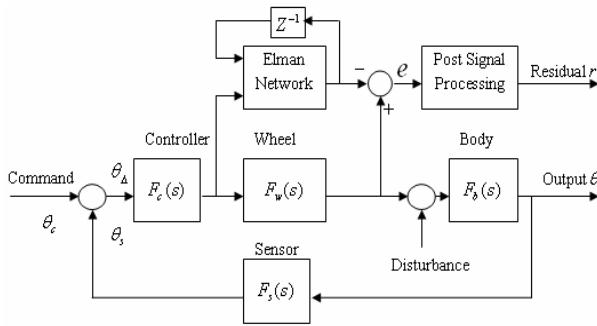
### 4 Neural Network Observer-Based FDI

In this section, we are going to develop a neural network observer-based scheme for fault detection and isolation in reaction wheels. Similar to the linear observer-based method in Section 3, three independent diagnosers are designed. The proposed block diagram is shown in Figure 4.1. It should be noted that for the purpose of recall of the network a different configuration is used as shown in Figure 4.2. The main difference



**Fig. 4.1.** Neural network scheme during the training phase

between these two schemes is that when the network is being trained to model the dynamics of the reaction wheel, the delayed value introduced into the network input is the actual reaction torque. One uses the difference  $e$  between the actual torque and the estimated torque from the network to tune the network synaptic weights  $W$  to minimize this difference. When the network is trained, the network inputs become the delayed value of the estimated torque from the network. Consequently, when the wheel becomes faulty, the neural network still represents the wheel dynamics in the normal and fault free case and the network yields the estimated torque in the normal case. This leads to substantial difference between the actual and the estimated torque for FDI.



**Fig. 4.2.** Neural network scheme for FDI during the recall/testing phase

#### 4.1 Network Architecture/Parameter Selection

In order to model the dynamics of the reaction wheel, a recurrent network known as Elman network [12] is utilized. This network has a feedback from the output of the hidden layer to the input of the hidden layer. This recurrent connection allows the Elman network to detect and generate time-varying patterns or model dynamic functions. The delay in the connection stores values from the previous time step, which can be used in the current time step. Obviously, more hidden neurons or layers are needed if the function being modeled is highly nonlinear and complex.

In this paper, a three-layer Elman network  $NN_{2 \times 25 \times 1}$  is constructed to model the reaction wheel dynamics. That is, the Elman network contains 2 input nodes, 25 hidden neurons and 1 output. One of the input nodes is the torque command voltage generated by the PID controller. The other input node is the one time step delay value of the actual reaction torque signal when the network is trained according to the scheme shown in Figure 4.1 or from the network output when the network has been trained and put into the fault detection and isolation phase as in Figure 4.2.

#### 4.2 Procedure for the Neural Network Observer-Based FDI Scheme

For neural network fault detection and isolation scheme in the reaction wheels, three separate networks for the wheels on the three axes independently are designed. The

procedures for this scheme for each axis are very similar. There are four main steps one needs to follow for each axis of the wheel for FDI:

### 1. Elman Back-Propagation Network Training

The data needed for the network training are the torque command voltage signal and the corresponding reaction torque signal. The torque command voltage and one time delay of the corresponding reaction torque serve as the network inputs and the corresponding reaction output serves as the desired network output. First, we let the satellite operate normally where one would collect the information about the torque command voltage and reaction torque from the reaction wheel.

### 2. Residual Generation

Based on the block diagram shown in Figure 4.1, we used the trained Elman network to estimate the output of the reaction wheel. The difference or the error between the actual and the estimated value of the reaction torque is put into a post-processing block to generate the residual for the purpose of FDI. First, we calculate the magnitude of the error. Second, it is processed through a moving average filter to generate a smooth signal. Finally, this moving average (selected as 40 as the “optimal” size) signal serves as a time varying residual threshold in our scheme.

### 3. Threshold Curve Determination for FDI

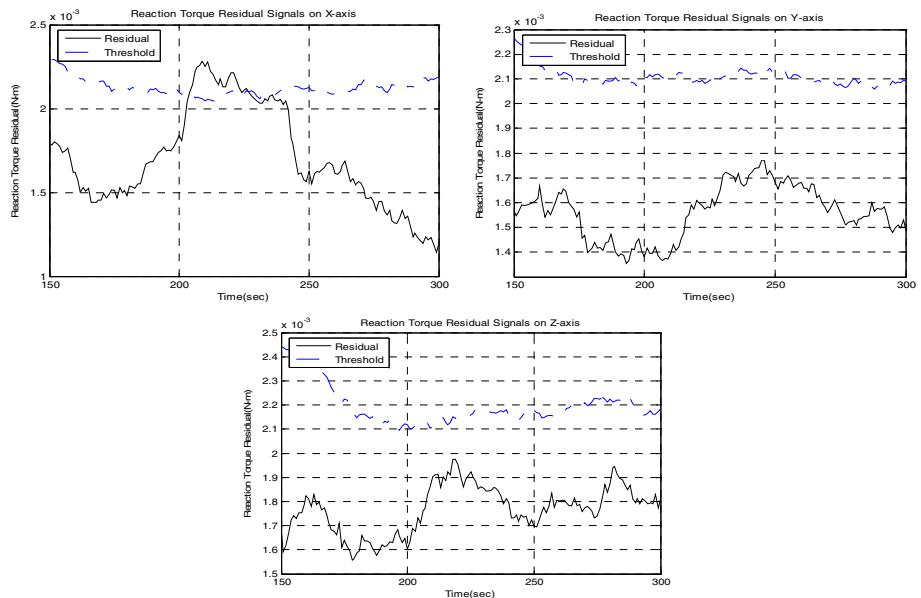
For the FDI purpose, we need a threshold curve for threshold checking. This kind of threshold curve should represent the upper bound of the residual signal in all cases corresponding to the normal (fault free) operating condition. Towards this end, we collect 100 residual curves generated in normal operation. The mean  $\bar{X}_{\text{residual}}$  and the standard deviation  $\sigma$  of these 100 residual curves are calculated. Thus, the threshold curve is defined according to  $X_{\text{threshold}} = \bar{X}_{\text{residual}} + 3 \times \sigma$ .

### 4. Fault Detection and Isolation by Using the Threshold Curves

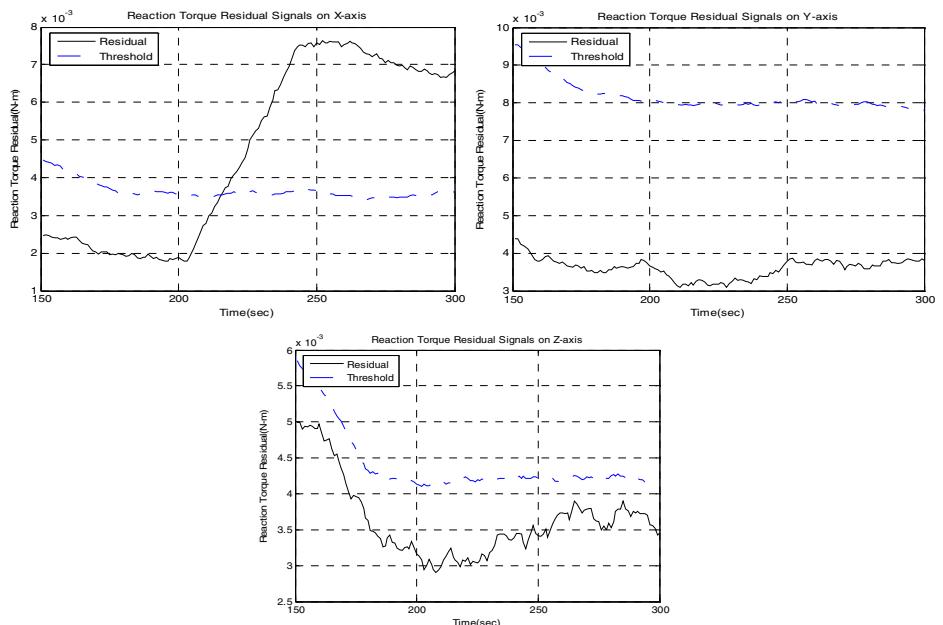
If any of the three residual curves exceeds its corresponding threshold curve, it indicates the reaction wheel on that axis may probably to be faulty. Since these three neural networks are designed separately and independently, the fault happening in one wheel should not affect the residual curves on the other two axes. This implies that we can detect and isolate the fault in the reaction wheels simultaneously and at the same time by the threshold checking.

## 5 Comparison of Linear and Neural Network Observer-Based FDI Schemes

The previously designed FDI schemes success highly depend on the modeling accuracy of the reaction wheel. As seen in Figure 2.1, the reaction wheel is a highly nonlinear system. For the linear observer-based FDI, all the nonlinearities and disturbances effects are ignored. This simplification in representation is clearly not quite accurate. Thus, an FDI scheme may be insensitive to small faults, but not robust to disturbances and noises. Whereas the neural network observer-based FDI scheme developed in Section 4 uses a well trained recurrent network to represent the dynamics of the reaction wheel. These are demonstrated through many simulation results below.

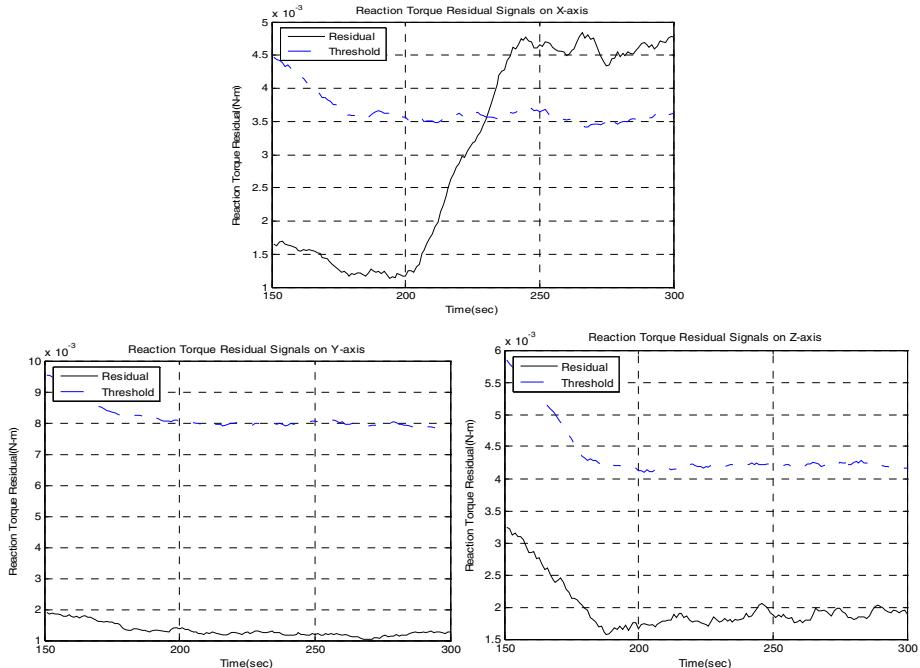


**Fig. 5.1.** Linear observer-based performance for small bus voltage fault



**Fig. 5.2.** Neural network observer-based performance for small bus voltage fault

Let us consider the case of bus voltage fault detection and isolation. By comparing the results shown in Figures 5.1 and 5.2, the residual curve of the X axis generated by the linear observer-based scheme exceeded its corresponding threshold curve shortly after the fault has occurred, however it falls beneath the residual curve again after a short time period. This false negative diagnosis is quite an undesirable outcome for the FDI algorithm. On the other hand, the residual curve generated by the neural network scheme exceeded its corresponding threshold curves and remained above the threshold curves thereafter, as shown in Figure 5.2.



**Fig. 5.3.** Neural network observer-based performance for small current loss fault

In another set of simulations, a case for the current loss fault detection and isolation is shown below. A 40% loss of the motor current corresponding to the reaction wheel aligned on the X axis has occurred at 200 second after a set point change command was applied. From the simulation results one finds that although the residual curve of the X-axis generated by the linear observer-based scheme (not shown due to space limitations) does increase for a few seconds after the fault has happened, but it never passes through the threshold curve. That is, one cannot detect this fault using this method. On the other hand, as shown in Figure 5.3, by using the neural network scheme, the residual exceeded its corresponding threshold and remains above it thereafter, so that the fault is correctly detected and isolated.

## 6 Conclusions

This paper has developed and presented a neural network observer-based scheme for fault detection and isolation in reaction wheels of a satellite. The faults considered are bus voltage fault, motor current loss fault and temperature fault. First, we investigated a linear diagnosis scheme for FDI. It is concluded that this method has a capability for fault detection and isolation in some cases, but it is not reliable and useful for small fault detection as it is highly sensitive to disturbances and noise. Subsequently, a dynamic neural network scheme is introduced to achieve better FDI performance for the reaction wheels. Through a comparative study between an observer-based and the dynamic neural network scheme, it is demonstrated that the latter performance is superior to the former for FDI. The latter is less sensitive to small faults and it has less probability of making false negative detection in noisy environments.

## References

- [1] P.C. Hughes, Spacecraft Attitude Dynamics, John Wiley & Sons, 1986.
- [2] J. R. Wertz (editor), Spacecraft Attitude Determination and Control, Kluwer Academic Publishers, 1995.
- [3] Bill Bialke, High Fidelity Mathematical Modeling of Reaction Wheel Performance, 21<sup>st</sup> Annual American Astronautical Society Guidance and Control Conference, 1998.
- [4] R. Isermann, Fault Detection and Fault Diagnosis Methods: An Introduction, 1997.
- [5] R. J. Patton, F. J. Uppal, J. C. Lopez-Toribio, Soft Computing Approaches to Fault Diagnosis for Dynamic System: A Survey, 2000.
- [6] P. J. Antsaklis and K. M. Passino, Eds., An Introduction to Intelligent and Autonomous Control, 1993.
- [7] R. F. Stengel, Toward Intelligent Flight Control, IEEE Trans. Syst., Man, Cybern., Vol. 23, No.6, pp. 1699-1717, 1993.
- [8] T. Sorsa, H. N. Koivo, and H. Koivisto, Neural Networks in Process Fault Diagnosis, IEEE Trans. Syst., Man, Cybern., Vol. 21, No. 4, pp. 815-825, 1991.
- [9] P. M. Frank, Enhancement of Robustness on Observer-Based Fault Detection, International Journal of Control, 59 (4): 955-983.
- [10] P. M. Frank, X. Ding, Survey of Robust Residual Generation and Evaluation Methods in Observer-Based Fault Detection System, Journal of Process Control, 7 (6): 403-424.
- [11] J. Chen , R. J. Patton, and H. Y. Zhang, Design of Unknown Input Observer and Robust Fault Detection Filters, Int. J. Control, 63 (1): 85-105.
- [12] J. L. Elman, Finding Structure in Time, Cognitive Science, vol. 14, pp. 179-211, 1990.
- [13] Simon Haykin, Neural Network: A Comprehensive Foundation, Prentice-Hall, Inc. Simon and Schuster, Upper Saddle River, New Jersey, USA, 1999.
- [14] B. C. Williams and P. P. Nayak, A Model-Based Approach to Reactive Self-Configuring System, Proc. 13th National Conference on Artificial Intelligence and 8th Innovative Applications of Artificial Intelligence Conference, pp.971-978, 1996.
- [15] R. Fujimaki, T. Yairi, and K. Machida, An Approach to Spacecraft Anomaly Detection Problem Using Kernel Feature Space, Proc. 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-05), pp.401-410, 2005.

# Gauss Chaotic Neural Networks

Yao-qun Xu<sup>1</sup>, Ming Sun<sup>1</sup>, and Ji-hong Shen<sup>2</sup>

<sup>1</sup> Institute of Computer and Information Engineering, Harbin University of Commerce,  
150028, Harbin, China

xuyq@hrbcu.edu.cn, Snogisun@tom.com

<sup>2</sup> Dept. of Mathematics, Harbin Engineering University, 150001, Harbin, China  
shenjihong@hrbeu.edu.cn

**Abstract.** We retrospect Chen's chaotic neural network and then propose a new chaotic neural network model whose activation function is composed of Gauss and Sigmoid function. And the time evolution figures of the largest Lyapunov exponents of chaotic single neural units are plotted. Based on the new model, the model with different parameters is applied to combinational optimization problems. 10-city traveling salesman problem (TSP) is given to make a comparison between Chen's and the new model with different parameters. Finally on the simulation results we conclude that the novel chaotic neural network model we proposed is more effective.

## 1 Introduction

Many combinatorial optimization problems arising from science and technology are often difficult to solve entirely. Hopfield and Tank first applied the continuous-time, continuous-output Hopfield neural network (HNN) to solve TSP<sup>[1]</sup>, thereby initiating a new approach to optimization problems<sup>[2, 3]</sup>. Since Hopfield's seminal work, there has been a strong interest in applying neural networks to solve the combinatorial problems. The Hopfield neural network, one of the well-known models of this type, converges to a stable equilibrium point due to its gradient decent dynamics; however, it causes sever local-minimum problems whenever it is applied to optimization problems.

Chaotic neural networks have been proved to be powerful tools for escaping from local minima. There were considerable literatures where presented and analyzed chaotic neural networks<sup>[8-11]</sup>. M-SCNN has a more powerful performance than Chen's in solving combinatorial optimization problems, especially in searching global minima of continuous nonlinear function and traveling salesman problems<sup>[4]</sup>.

In this paper, we first review the Chen's chaotic neural network. Second, we propose a novel chaotic neural network model. Third, the time evolution figures of their largest Lyapunov exponents of the neural units are given. At last, the two networks are applied to 10-city traveling salesman problem (TSP) in order to make a comparison. Finally it is conclude the novel chaotic neural network model we proposed is more valid.

## 2 Chaotic Neural Network Models

In this section, two chaotic neural network models are given. And the first is proposed by Chen, the second proposed by ourselves.

## 2.1 Chaotic Simulated Annealing with Decaying Self-coupling

Chen and Aihara's transiently chaotic neural network<sup>[5]</sup> is described as follows:

$$x_i(t) = f(y_i(t)) = \frac{1}{1 + e^{-y_i(t)/\epsilon}} \quad (1)$$

$$y_i(t+1) = ky_i(t) + \alpha \left[ \sum_j W_{ij} x_j + I_i \right] - z_i(t)(x_i(t) - I_0) \quad (2)$$

$$z_i(t+1) = (1 - \beta)z_i(t) \quad (3)$$

where  $x_i(t)$  is output of neuron  $i$ ;  $y_i(t)$  denotes internal state of neuron  $i$ ;  $W_{ij}$  describes connection weight from neuron  $j$  to neuron  $i$ ,  $W_{ij} = W_{ji}$ ;  $I_i$  is input bias of neuron  $i$ ,  $\alpha$  a positive scaling parameter for neural inputs,  $k$  damping factor of nerve membrane,  $0 \leq k \leq 1$ ,  $z_i(t)$  self-feedback connection weight (refractory strength)  $\geq 0$ ,  $\beta$  damping factor of  $z_i(t)$ ,  $0 < \beta < 1$ ,  $I_0$  a positive parameter,  $\epsilon$  steepness parameter of the output function,  $\epsilon > 0$ .

## 2.2 Gauss-Sigmoid Chaotic Neural Network Model (G-SCNN)

Gauss-Sigmoid chaotic neural network is a novel model proposed by ourselves, described as follows:

$$y_i(t+1) = ky_i(t) + \alpha \left[ \sum_j W_{ij} x_j + I_i \right] - z_i(t)(x_i(t) - I_0) \quad (4)$$

$$z_i(t+1) = \frac{z_i(t)}{\ln(e + \lambda(1 - z_i(t)))} \quad (5)$$

$$x_i(t) = f(y_i(t)) \quad (6)$$

$$f(y_i(t)) = (A - \gamma_i(t))G((1 + r_i(t))y_i(t)) + S((1 + r_i(t))y_i(t)) \quad (7)$$

$$\gamma_i(t+1) = (1 - \beta_1)\gamma_i(t) \quad (8)$$

$$r_i(t+1) = \beta_2 r_i(t) \quad (9)$$

$$G(u) = e^{-h(u-a)^2} \quad (10)$$

$$S(u) = \frac{1}{1 + e^{-c(u-b)}} \quad (11)$$

where  $x_i(t)$ ,  $y_i(t)$ ,  $W_{ij}$ ,  $\alpha$ ,  $k$ ,  $I_i$ ,  $z_i(t)$ ,  $I_0$  are the same with the above. And  $\lambda$  is a positive parameter which controls the speed of this annealing process;  $\gamma_i(0)$  and  $r_i(0)$  are important parameters of activation function which should be varied with kinds of special optimization problems,  $0 < \beta_1 \leq 1$ ,  $0 < \beta_2 < 1$ .  $A$  is a positive parameter.  $h, a, b, c$  are important parameters of *Gauss* and *Sigmoid* function.

### 3 Research on Lyapunov Exponent of Neural Units

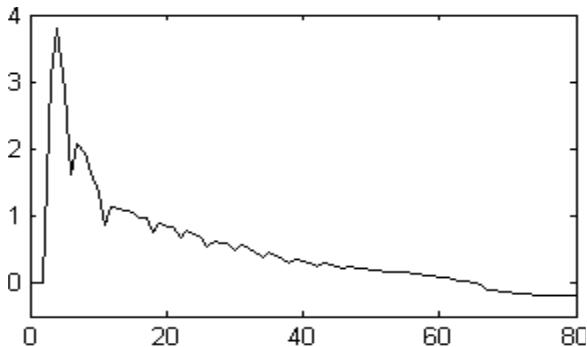
In this section, the largest Lyapunov exponents' figures of the neural units ( $\alpha = 0$ ) of Chen's and M-SCNN in the same parameter of  $\beta = 0.008$  are plotted.

#### 3.1 Chen's Chaotic Neural Unit

The parameters are set as follows:

$$k = 0.6, I_0 = 0.1, \varepsilon = 1/250, z(0) = 0.1, y(0) = 0.283.$$

The time evolution figure of the largest Lyapunov exponent is shown as Fig.1:



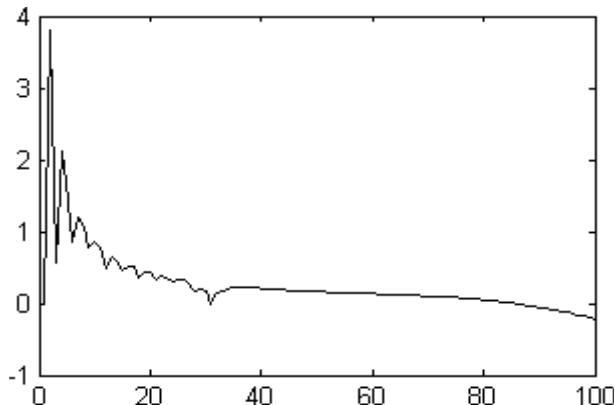
**Fig. 1.** Lyapunov exponent time evolution figure

#### 3.2 Gauss-Sigmoid Chaotic Neural Unit

The parameters are set as follows:

$$k = 1, I_0 = 0.65, y(0) = 0.2, z(0) = 0.2, \lambda = 0.007, \beta_1 = 0.001, \beta_2 = 0.98, \gamma(0) = 0.1, A = 1, r(0) = 32.5, h = 0.2, a = -21, b = 5.0, c = 5.0.$$

The time evolution figure of the most positive Lyapunov exponent is shown as Fig.2.



**Fig. 2.** Lyapunov exponent time evolution figure

Seen from the above figures, one can conclude that the chaotic motions exist in the neurons of both Chen's and the proposed by us.

In order to testify the performance of our novel model, the proposed network is applied to the traveling salesman problem (TSP).

## 4 Application to Traveling Salesman Problem

A solution of TSP with  $N$  cities is represented by  $N \times N$ -permutation matrix, where each entry corresponds to output of a neuron in a network with  $N \times N$  lattice structure. Assume  $v_{xi}$  to be the neuron output which represents city  $x$  in visiting order  $i$ . A computational energy function which is to minimize the total tour length while simultaneously satisfying all constraints takes the follow form [6]:

$$E = \frac{A}{2} \left( \sum_{x=1}^N \sum_{i=1}^N (v_{xi} - 1)^2 + \sum_{i=1}^N \left( \sum_{x=1}^N v_{xi} - N \right)^2 \right) + \frac{B}{2} \sum_{x=1}^N \sum_{y=1}^N \sum_{i=1}^N d_{xy} v_{xi} v_{y,i+1} \quad (12)$$

where  $v_{i0} = v_{in}$  and  $v_{i,n+1} = v_{i1}$ .  $A$  and  $B$  are the coupling parameters corresponding to the constraint function and the cost function of the tour length, respectively.  $d_{xy}$  is the distance between city  $x$  and city  $y$ .

The coordinates of 10-city is as follows:

(0.4, 0.4439), (0.2439, 0.1463), (0.1707, 0.2293), (0.2293, 0.716), (0.5171, 0.9414),  
(0.8732, 0.6536), (0.6878, 0.5219), (0.8488, 0.3609), (0.6683, 0.2536), (0.6195, 0.2634).

The shortest distance of the 10-city is 2.6776.

Here are the results of the test about Chen's and G-SCNN.

The coupling parameters corresponding to the constraint function and the cost function of the tour length we adopt are set as follows:  $A=2.5$ ,  $B=1$ .

(1)The parameters of Chen's are set as follows :

$$\alpha=0.2, k=1, I_0=0.5, \varepsilon=1/20, z(0)=[0.5, 0.5].$$

**Table1.** Simulation Result of Chen's Chaotic Neural Network

Model	VN	GN	VP	GP
Chen's	188	188	94%	94%
	185	185	92.5%	92.5%
	183	183	91.5%	91.5%
	184	184	92%	92%
	181	180	90.5%	90%
	175	175	87.5%	87.5%
	180	179	90%	89.5%
	189	189	94.5%	94.5%
	187	186	93.5%	93%
	178	178	89%	89%
average	183	182.7	91.5%	91.35%

200 different initial conditions are generated randomly in the region [0, 1] when  $\beta = 0.008$ , as are shown in table1 (the following Tables are the same as this one). (VN= valid number; GN= global number; VP= valid percent; GP=global percent.)

In order to gain insight into the performance of the proposed model, we make tests with different  $A$ ,  $\beta_1$  and  $\gamma(0)$ .

(2)The parameters of G-SCNN are set as follows:

$$\begin{aligned} k &= 1, I_0 = 0.5, z(0) = 0.1, \lambda = 0.007, \beta_1 = 0.001, \beta_2 = 0.9, \gamma(0) = 0.1, A = 1, h = 0.2, \\ r(0) &= 200.3, a = -2.1, b = 5.0, c = 5.0. \end{aligned}$$

**Table 2.** Simulation Result of G-SCNN

Model	VN	GN	VP	GP
G-SCNN ( $A = 1$ , $\beta_1 = 0.001$ , $\gamma(0) = 0.1$ )	191	187	95.5%	93.5%
	189	187	94.5%	93.5%
	188	187	94%	93.5%
	192	187	96%	93.5%
	196	196	98%	98%
	189	186	94.5%	93%
	189	189	94.5%	94.5%
	186	185	93%	92.5%
	191	190	95.5%	95%
	191	190	95.5%	95%
average	190.2	188.4	95.1%	94.2%

(3)The parameters of G-SCNN are set as follows:

$$\begin{aligned} k &= 1, I_0 = 0.5, z(0) = 0.1, \lambda = 0.007, \beta_1 = 0.08, \beta_2 = 0.9, \gamma(0) = 0.1, A = 1, r(0) = 200.3, \\ h &= 0.2, a = -2.1, b = 5.0, c = 5.0. \end{aligned}$$

**Table 3.** Simulation Result of G-SCNN

Model	VN	GN	VP	GP
G-SCNN ( $A = 1$ , $\beta_1 = 0.08$ , $\gamma(0) = 0.1$ )	187	185	93.5%	92.5%
	196	194	98%	97%
	189	188	94.5%	94%
	191	191	95.5%	95.5%
	187	186	93.5%	93%
	182	182	91%	91%
	195	194	92.5%	92%
	192	192	96%	96%
	190	189	95%	94.5%
	188	186	94%	93%
average	189.7	188.7	94.85%	94.35%

(4)The parameters of G-SCNN are set as follows:

$$k=1, I_0=0.5, z(0)=0.1, \lambda=0.007, \beta_1=0.99, \beta_2=0.9, \gamma(0)=0.2, A=1, r(0)=200.3, h=0.2, a=-2.1, b=5.0, c=5.0.$$

**Table 4.** Simulation Result of G-SCNN

Model	VN	GN	VP	GP
G-SCNN ( $A=1$ , $\beta_1=0.99$ , $\gamma(0)=0.2$ )	195	195	97.5%	97.5%
	197	197	98.5%	98.5%
	188	187	94%	93.5%
	184	182	92%	91%
	196	195	98%	97.5%
	194	194	97%	97%
	187	186	93.5%	93%
	184	183	92%	92.5%
	190	189	95%	94.5%
	193	191	96.5%	95.5%
average	190.8	189.9	95.4%	94.95%

(5)The parameters of G-SCNN are set as follows:

$$k=1, I_0=0.5, z(0)=0.1, \lambda=0.007, \beta_1=1, \beta_2=0.9, \gamma(0)=0, A=1, r(0)=200.3, h=0.2, a=-2.1, b=5.0, c=5.0.$$

**Table 5.** Simulation Result of G-SCNN

Model	VN	GN	VP	GP
G-SCNN ( $A=1$ , $\beta_1=1$ , $\gamma(0)=0$ )	189	187	94.5%	93.5%
	194	192	97%	96%
	195	194	97.5%	97%
	188	187	94%	93.5%
	193	193	96.5%	96.5%
	188	185	94%	92.5%
	195	193	97.5%	96.5%
	184	181	94%	90.5%
	191	189	95.5%	94.5%
	194	191	97%	95.5%
average	191.1	189.2	95.55%	94.6%

(6)The parameters of G-SCNN are set as follows:

$$k=1, I_0=0.5, z(0)=0.1, \lambda=0.007, \beta_1=0.99, \beta_2=0.9, \gamma(0)=0.1, A=2, r(0)=200.3, h=0.2, a=-2.1, b=5.0, c=5.0.$$

**Table 6.** Simulation Result of G-SCNN

Model	VN	GN	VP	GP
G-SCNN ( $A = 2$ , $\beta_1 = 0.99$ , $\gamma(0) = 0.1$ )	191	190	95.5%	95%
	184	184	92%	92%
	189	188	94.5%	94%
	187	187	93.5%	93.5%
	191	191	95.5%	95.5%
	191	191	95.5%	95.5%
	189	188	94.5%	94%
	193	193	96.5%	96.5%
	190	190	95%	95%
	194	194	97%	97%
	average	189.9	94.95%	94.8%

(7)The parameters of G-SCNN are set as follows:

$$k=1, I_0=0.5, z(0)=0.1, \lambda=0.007, \beta_1=0.99, \beta_2=0.9, \gamma(0)=0.1, A=3, r(0)=200.3, h=0.2, a=-2.1, b=5.0, c=5.0.$$

**Table 7.** Simulation Result of G-SCNN

Model	VN	GN	VP	GP
G-SCNN ( $A = 3$ , $\beta_1 = 0.99$ , $\gamma(0) = 0.1$ )	184	184	92%	92%
	192	192	96%	96%
	191	191	95.5%	95.5%
	190	190	95%	95%
	192	191	96%	95.5%
	190	190	95%	95%
	185	185	92.5%	92.5%
	187	187	93.5%	93.5%
	191	189	95.5%	94.5%
	194	194	97%	97%
	average	189.6	94.8%	94.65%

(8)The parameters of G-SCNN are set as follows:

$$k=1, I_0=0.5, z(0)=0.1, \lambda=0.007, \beta_1=0.99, \beta_2=0.9, \gamma(0)=0.1, A=10, h=0.2, r(0)=200.3, a=-2.1, b=5.0, c=5.0.$$

**Table 8.** Simulation Result of G-SCNN

Model	VN	GN	VP	GP
G-SCNN ( $A = 1$ , $\beta_1 = 1$ , $\gamma(0) = 0$ )	188	188	94%	94%
	183	181	91.5%	90.5%
	193	191	96.5%	95.5%
	188	188	94%	94%

**Table 8.** (*continue*)

	192	192	96%	96%
	183	182	91.5%	91%
	189	186	94.5%	93%
	191	191	95.5%	95.5%
	192	191	96%	95.5%
	191	191	95.5%	95.5%
average	189	188.1	94.5%	94.05%

(9)The parameters of G-SCNN are set as follows:

$$\begin{aligned} k=1, I_0=0.5, z(0)=0.1, \lambda=0.007, \beta_1=0.99, \beta_2=0.9, \gamma(0)=0.1, A=0, h=0.2, \\ r(0)=200.3, a=-2.1, b=5.0, c=5.0. \end{aligned}$$

**Table 9.** Simulation Result of G-SCNN

Model	VN	GN	VP	GP
G-SCNN	193	181	96.5%	90.5%
	187	181	93.5%	90.5%
	178	173	89%	86.5%
	186	182	93%	91%
	(A = 0,	187	93.5%	90.5%
	$\beta_1=0.99,$	184	92%	90%
	$\gamma(0)=0.1)$	181	90.5%	86%
	185	178	92.5%	89%
	184	179	92%	89.5%
	185	179	92.5%	89.5%
average	185	178.6	92.5%	89.3%

An examination of Table2~9 yields the following observations:

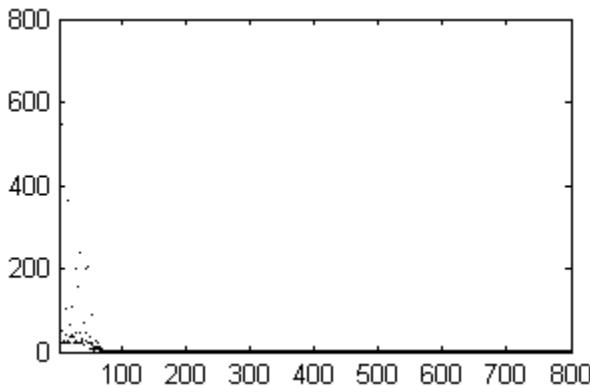
First, the proposed model can solve the TSP with a high percentage of global minimal path-distance and a high percentage of valid path-distance when  $A=1, 2, 3, 10$ .

Second, the proposed model with variable  $\gamma(t)$  is superior to the one with invariable  $\gamma(t)$  in average percentage of global minimal path-distance. As is shown in Table4, Table5, the average percentage of global minimal path-distance of the Table5 where  $\beta_1$  is equal to 1 that means  $\gamma(t)$  invariable is somewhat lower than that of the Table4.

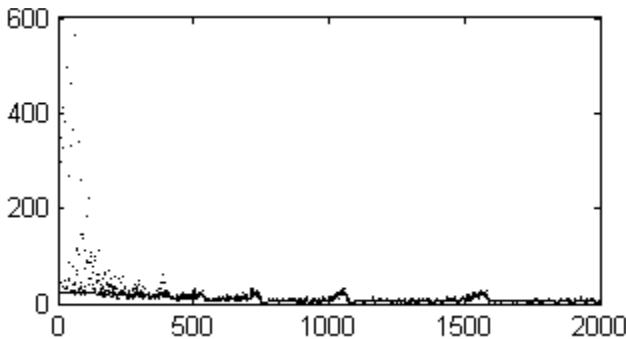
The time evolution figures of the energy function of M-SCNN and Chen's in solving TSP are respectively given in Fig.3 and Fig.4.

By comparison, it is concluded that M-SCNN is superior to Chen's model. From the Fig.3, Fig4, one can see that the velocity of convergence of G-SCNN is much faster than that of Chen's in solving TSP.

The superiority of G-SCNN contributes to several factors: First, because of the higher nonlinear nature of Gauss function, the activation function of G-SCNN has a stronger performance in solving combinatorial optimization problems than Chen's. Second, it is easier to produce chaotic phenomenon [8] in that the activation function is non-monotonic. Third, the activation function of G-SCNN is varied with time.



**Fig. 3.** Energy time evolution figure of M-SCNN



**Fig. 4.** Energy time evolution figure of Chen's

## 5 Conclusions

We have introduced two models of chaotic neural networks. To verify the effectiveness of it, we have made comparison with Chen's model in optimization problems. By comparison, one can conclude that G-SCNN is superior to Chen's in searching global minima of continuous nonlinear function.

Different from Chen's model, the activation function of G-SCNN is composed by Gauss and Sigmoid. So, besides it has the quality of sigmoid activation, the activation function of G-SCNN has a higher nonlinear nature than Sigmoid, which is easier to produce chaotic phenomenon<sup>[7]</sup> because of its non-monotonic. Due to these factors, G-SCNN is superior to Chen's in solving TSP.

## References

1. Hopfield J J., Tank D W.: Neural Computation of Decision in Optimization Problems. *Biol. Cybern.* Vol.52. (1985)141-152
2. Hopfield, J.: Neural Networks and Physical Systems with Emergent Collective Computational Abilities. Vol. 79. In: *Proc Natl Acad Sci*, (1982) 2554-2558

3. Hopfield, J.: Neural Networks and Physical Systems with Emergent Collective Computational Abilities. Vol. 81. Proc. Natl. Acad. Sci. (1984)3088-3092
4. Yao-qun Xu, Ming Sun, Guang-ren Duan.: Wavelet Chaotic Neural Networks and Their Application to Optimization Problems. Lecture Notes in Computer Science.vol.3971, Springer Berlin, (2006)379-384
5. Chen L., Aihara K. "Chaotic Simulated Annealing by a Neural Network Model with Transient Chaos." Neural Networks. Vol. 8. (1995)915-930
6. Shou-yu Sun, Jun-li Zheng.: A Kind of Improved Algorithm and Theory Testify of Solving TSP in Hopfield Neural Network. Vol.1. Journal of electron. (1995)73-78
7. A Potapove, M KAlich.: Robust Chaos in Neural Networks. Vol.277. Physics Letters A, (2000) 310-322
8. Aihara, K.: Chaos engineering and its application to parallel distributed processing with chaotic neural networks. Proceedings of the IEEE, 90 (2002) 919 -930
9. Wang, L.P., Li, S., Tian F.Y, Fu, X.J.: A noisy chaotic neural network for solving combinatorial optimization problems: Stochastic chaotic simulated annealing. IEEE Trans. System, Man, Cybern, Part B - Cybernetics 34 (2004) 2119-2125
10. Wang, L.P., Shi, H.: A noisy chaotic neural network approach to topological optimization of a communication network with reliability constraints. Proc. International Symposium on Neural Networks (ISNN 2004), LNCS, vol.3174. Springer Berlin, (2004) 230-235
11. Wang, L.P., Smith, K.: On chaotic simulated annealing. IEEE Trans Neural Networks, 9 (1998) 716-718

# Short-Term Load Forecasting Using Multiscale BiLinear Recurrent Neural Network

Dong-Chul Park<sup>1</sup>, Chung Nguyen Tran<sup>1</sup>, and Yunsik Lee<sup>2</sup>

<sup>1</sup> Dept. of Information Engineering, Myong Ji University, Korea  
`{parkd, tnchung, songjae}@mjnu.ac.kr`

<sup>2</sup> SoC Research Center, Korea Electronics Tech. Inst., Seongnam, Korea  
`leeys@keti.re.kr`

**Abstract.** In this paper, a short-term load forecasting model using wavelet-based neural network architecture termed a Multiscale BiLinear Recurrent Neural Network (M-BLRNN) is proposed. The M-BLRNN is a combination of several BiLinear Recurrent Neural Network (BLRNN) models. Each BLRNN predicts a signal at a certain resolution level obtained by the wavelet transform. The experiments and results on the load data from the North-American Electric Utility (NAEU) show that the M-BLRNN outperforms both a traditional MultiLayer Perceptron Type Neural Network (MLPNN) and the BLRNN in terms of the Mean Absolute Percentage Error (MAPE).

## 1 Introduction

In recent years, problems with improving the accuracy of load forecasting have been an important topic of researches in electrical engineering due to the requirements of power system operational planning, such as an optimum generating unit commitment, economical load dispatch, the need to maintain scheduling, and fuel constraints. In addition, the load forecasting is also useful for the system security in which forecasting the load can provide valuable information for detecting much vulnerability in advance of problems they may bring [1]. An accurate load forecasting is, therefore, critical from an economic as well as a security standpoint.

Various models have been proposed for load forecasting. Most previous approaches forecast a future electricity demand based on historical information on electricity consumption and weather conditions. The historical information on electricity consumption is modeled as a function of its past observed values, while the weather condition is embedded by using weather variables such as temperature. Statistical models such as the autoregressive model [2], the linear regression model [3], and the autoregressive moving average (ARMA) [4] have been widely used in practice. However, these models are based on a linear analysis which lacks the ability to capture nonlinearities in a load series.

Recently, several nonlinear models have been proposed for load forecasting. Among these models, neural networks have received great attention. Neural networks have been shown to have the ability not only to model time series load

curves but also to model an unspecified nonlinear relationship between a load demand and weather variables[1,5,6,7]. A comprehensive review of the application of neural networks to load forecasting from most recent papers shows that neural network-based models give encouraging results and are well accepted in practice by many utilities [8]. However, due to the very high costs associated with errors in practice, improvements of load forecasting models, including learning algorithms and network architecture, still need to be sought.

In this paper, a short-term load forecasting scheme based on the Multiscale BiLinear Recurrent Neural Network (M-BLRNN) is proposed. The M-BLRNN employs a wavelet transform [9,10] to decompose the load profile into a multiresolution representation. In doing this, a complex load profile is simplified by several sub-profiles at each level of resolution. Therefore, the difficult prediction task associated with the original load profile is simplified by predicting the decomposed sub-profiles at each resolution level. Furthermore, the BiLinear Recurrent Neural Network (BLRNN) [11] which has been shown to be very efficient in modeling time series [11,12] is employed for modeling the load profile at each resolution level.

The remainder of this paper is organized as follows: Section 2 presents a review of the multiresolution analysis with the wavelet transform. A brief review of the BLRNN is given in Section 3. The M-BLRNN is presented in Section 4. Section 5 presents experiments and results on the short-term load forecasting problem using the M-BLRNN, including a performance comparison with the traditional MLPNN and BLRNN models. Concluding remarks are given in Section 6.

## 2 Multiresolution Wavelet Analysis

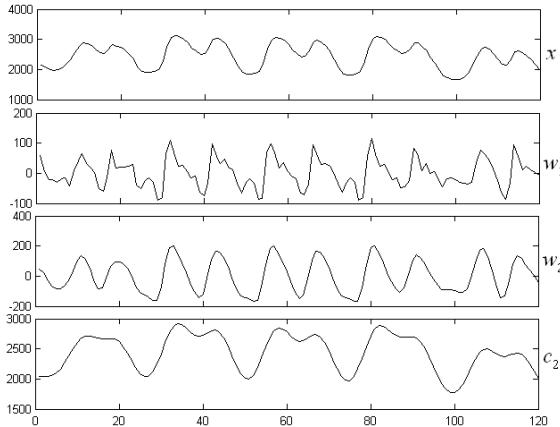
The aim of multiresolution analysis is to analyze a signal at different frequencies with different resolutions. It produces a high quality local representation of a signal in both the time domain and the frequency domain. The wavelet transform [9,10], a novel technology developed in the signal processing community, was proven to be suitable for the multiresolution analysis of time series data [13].

The à trous wavelet transform was first proposed by Shensa [10]. The calculation of the à trous wavelet transform can be described as follows: First, a low-pass filter is used to suppress the high frequency components of a signal and allows the low frequency components to pass through. Then, the scaling function associated with the low-pass filter is used to calculate the average of elements, which result in a smoother signal.

The smoothed data  $c_j(t)$  at given resolution  $j$ , can be obtained by performing successive convolutions with the discrete low-pass filter  $h$ ,

$$c_j(t) = \sum_k h(k)c_{j-1}(t + 2^{j-1}k) \quad (1)$$

where  $h$  is a discrete low-pass filter associated with the scaling function and  $c_0(t)$  is the original signal. A suitable low-pass filter  $h$  is the  $B_3$  spline, defined as  $(\frac{1}{16}, \frac{1}{4}, \frac{3}{8}, \frac{1}{4}, \frac{1}{16})$ .



**Fig. 1.** Example of wavelet and scaling coefficients for a electric load data

From the sequence of the smoothing of the signal, the wavelet coefficients are obtained by calculating the difference between successive smoothed versions:

$$w_j(t) = c_{j-1}(t) - c_j(t) \quad (2)$$

By consequently expending the original signal from the coarsest resolution level to finest resolution level, the original signal can be expressed in terms of the wavelet coefficients and the scaling coefficients as follow:

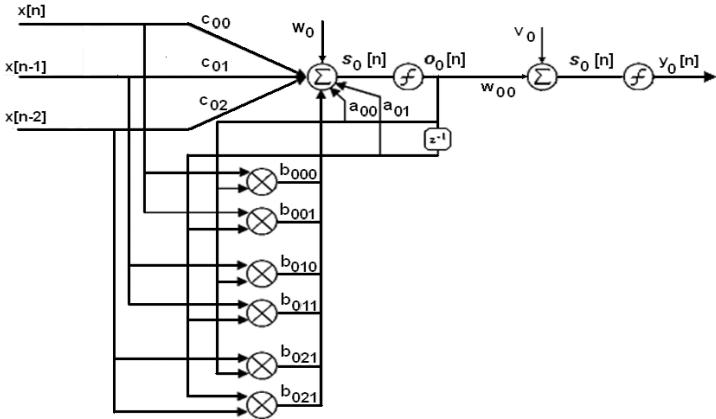
$$c_0(t) = c_J(t) + \sum_{j=1}^J w_j(t) \quad (3)$$

where  $J$  is the number of resolutions and  $c_J(t)$  is the finest version of the signal. Eq.(3) also provides a reconstruction formula for the original signal.

Fig. 1 shows an example of the wavelet coefficients and the scaling coefficients for two levels of resolution for the hourly electric load data of the North American Electric Utility (NAEU) from Jan. 1, 1995 to Jan. 5, 1995. From the top to the bottom are the original signal, two levels of the wavelet coefficients, and the finest scaling coefficients, respectively.

### 3 BiLinear Recurrent Neural Networks

The BLRNN is a simple recurrent neural network, which has a robust ability in modelling nonlinear systems and is especially suitable for time-series data. The model was initially proposed by Park and Zhu [11]. It has been successfully applied in modeling time-series data [11,12]. Fig. 2 illustrates a simple 3-1-1 BLRNN with 2 recursion lines.



**Fig. 2.** Simple BLRNN with structure 3-1-1 and 2 recursion lines

For a one-dimensional input/output case, the output value of a bilinear recurrent neuron is computed by the following equation:

$$s[n] = \sum_{i=1}^{N-1} a_i y[n-i] + \sum_{i=0}^{N-1} \sum_{j=1}^{N-1} b_{ij} y[n-j] x[n-i] + \sum_{i=0}^{N-1} c_i x[n-i] \quad (4)$$

where  $x[i]$  is the input,  $y[i]$  is the output, and  $N$  is the order of recursion.

In the following, we explain about a simple BLRNN that has  $N$  input neurons,  $M$  hidden neurons and where  $K = N - 1$  degree polynomials is given. The input signal and the nonlinear integration of the input signal to hidden neurons are:

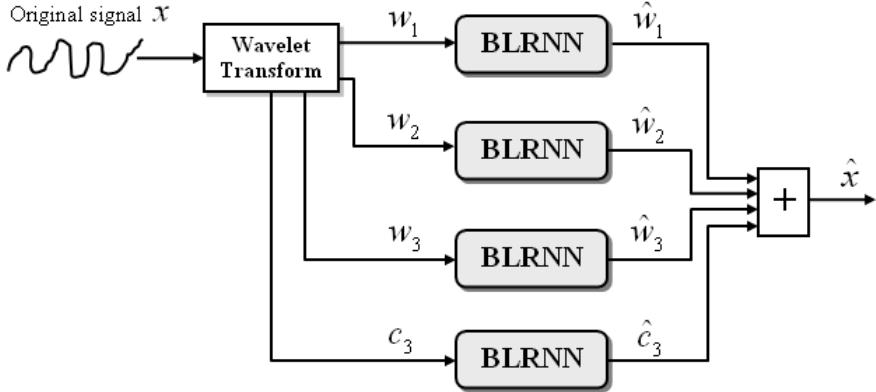
$$\begin{aligned} \mathbf{X}[n] &= [x[n], x[n-1], \dots, x[n-K]]^T \\ \mathbf{O}[n] &= [o_1[n], o_2[n], \dots, o_M[n]]^T \end{aligned}$$

where  $T$  denotes the transpose of a vector or matrix. For  $p$ -th hidden neuron, the following expressions are obtained:

$$\mathbf{Z}_p[n] = [o_p[n-1], o_p[n-2], \dots, o_p[n-K]]$$

And

$$\begin{aligned} s_p[n] &= w_p + \sum_{k_1=0}^{N-1} a_{pk_1} o_p[n-k_1] \\ &+ \sum_{k_1=0}^{N-1} \sum_{k_2=0}^{N-1} b_{pk_1 k_2} o_p[n-k_1] x[n-k_2] \\ &+ \sum_{k_2=0}^{N-1} c_{pk_2} x[n-k_2] \\ &= w_p + \mathbf{A}_p^T \mathbf{Z}_p^T[n] + \mathbf{Z}_p[n] \mathbf{B}_p^T \mathbf{X}[n] + \mathbf{C}_p^T \mathbf{X}[n] \end{aligned} \quad (5)$$



**Fig. 3.** Example of Multiscale BiLinear Recurrent Neural Network with 3 resolution levels

where  $c_p$  is the weight of bias neuron.  $\mathbf{A}_p$  is the weight vector for the recurrent portion,  $\mathbf{B}_p$  is the weight matrix for the bilinear recurrent portion and  $\mathbf{C}_p$  is the weight vector for the feedforward portion and  $p=1,2 \dots, M$ . Let  $\phi$  be the activation function of the hidden neuron; the output of  $p-th$  hidden neuron is then:

$$o_p[n] = \phi(s_p[n]) \quad (6)$$

From the hidden layer to the output layer, it is the same as a traditional feedforward-type neuron network

$$s_l[n] = v_l + \sum_{p=0}^{N_h-1} w_{lp} o_p[n] \quad (7)$$

where  $w_{lp}$  is the weight between the hidden and the output neurons, and  $N_h$  is the number of hidden neurons. The final output is obtained by applying the activation function

$$y_l[n] = \phi(s_l[n]) \quad (8)$$

More detailed information on the a BLRNN and its learning algorithm can be found in [1,5].

#### 4 Multiscale BiLinear Recurrent Neural Network

The M-BLRNN is a combination of several individual BLRNN models in which each BLRNN model is employed to predict a signal at each resolution level. Fig. 3 illustrates an example of a M-BLRNN with three resolution levels.

The prediction of a time series signal based on the M-BLRNN can be separated into three stages. In the first stage, the original signal is decomposed into the wavelet coefficients and the scaling coefficients based on the number of resolution levels. In the second stage, coefficients at each resolution level are predicted by an

individual BLRNN model. It should be noted that the predictions of coefficients at each resolution level are independent and can be done in parallel fashion. In the final stage, all of the prediction results from individual BLRNNs are combined using the reconstruction formula of Eq.(3):

$$\hat{x}(t) = \hat{c}_J(t) + \sum_{j=1}^J \hat{w}_j(t) \quad (9)$$

where  $\hat{c}_J(t)$ ,  $\hat{w}_j(t)$ , and  $\hat{x}(t)$  represent the predicted values of the finest scaling coefficients, the predicted values of the wavelet coefficients at level  $j$ , and the predicted values of the original time series signal, respectively.

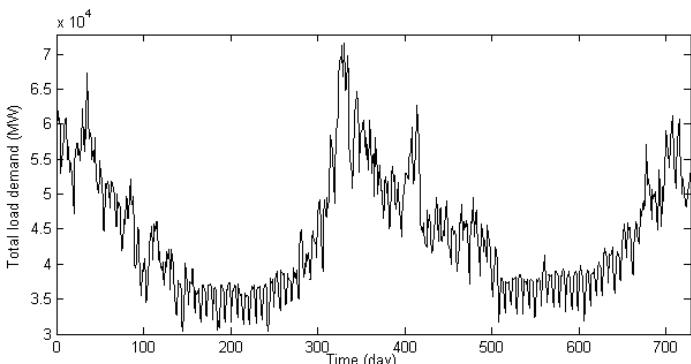
By adopting the BLRNN in the multiscale architecture, the M-BLRNN is able to learn the correlation structure and internal information at each resolution level hidden in the original signal, as the original signal is decomposed into several simpler sub-signals with the form of the wavelet coefficients.

## 5 Experiments and Results

### 5.1 Load Data

In order to verify the applicability of the M-BLRNN for short-term load forecasting, the load and temperature data from the NAEU are used for the experiments. The load and temperature data were recorded at every hour of the day from January 1, 1985 to October 12, 1992, rendering 2,834 days of load and temperature data. The load and temperature data can be downloaded at <http://www.ee.washington.edu/class/559/2002spr>.

Fig. 4 shows the daily total load demands from January 1, 1985 to December 31, 1986. The load demand displays seasonal patterns: a high demand for electricity in the winter and a low demand in the summer in this case. Based on these seasonal characteristics, the load data are separated into two sets: a



**Fig. 4.** Daily total load from January 1, 1985 to December 31, 1986

summer load data set and a winter load data set. Therefore, it is suggested that the load demand in the summer and winter are modeled individually.

Before the load data set is ready for neural network inputs, preprocessing is advisable in order for it to be suitable for the neural network inputs. First, the hourly load and hourly temperature data are linearly scaled to (0, 1); that is, to the range of the logistic function. Following this, a multiresolution analysis is applied to all load data to prepare them for the inputs of the M-BLRNN. Different numbers of resolution levels were tested in an effort to find a sufficient number. In the present experiments, three levels of resolution were chosen. An example of the wavelet coefficients and the scaling coefficients at each level of resolution is given in Fig. 1.

## 5.2 Short-Term Load Forecasting

In order to perform short-term load forecasting, a time series based model incorporating other exogenous variables was employed

$$x(t) = f(x(t-1), \dots, x(t-k+1); \mathbf{Y}) \quad (10)$$

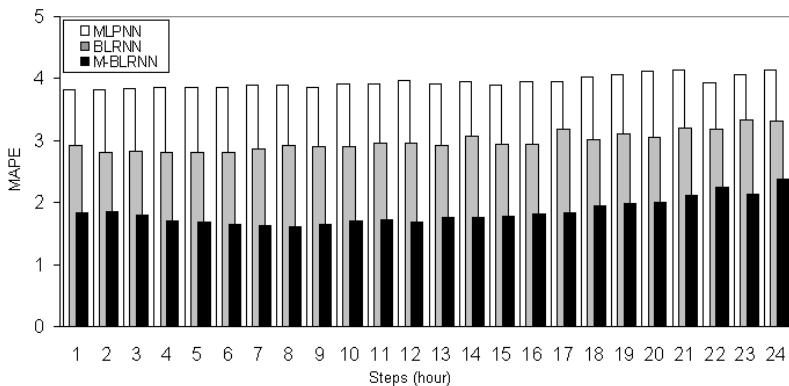
where  $x(t)$  is the hourly load demand at time  $t$ , and  $k$  is the embedding dimension.  $\mathbf{Y}$  is a vector representing the exogenous variables such as weather variables and calendar variables.

As can be seen from Eq. (10), the inputs of load forecasting models consist of past hourly load demand and other exogenous variables such as weather variables and calendar variables. The selecting of the load variables is based on a correlation analysis from the load data. The temperature variables are added for every time point at which a load variable was included. The calendar information is embedded by encoding two additional inputs,  $\sin(2\pi t/24)$  and  $\cos(2\pi t/24)$  [14], where  $t$  is the hour of the day. Table 1 summarizes the list of input variables for the short-term load forecasting model.

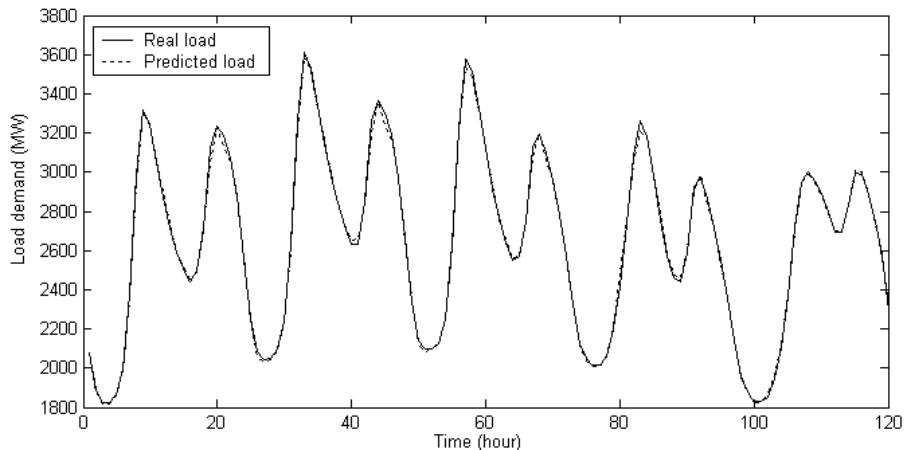
All of the above input variables are used for various load forecasting schemes including the traditional MLPNN, the BLRNN, and the M-BLRNN. It should be noted that the load variables of each individual BLRNN model at each resolution level are calculated from each sub-profile in the multiresolution representation for the M-BLRNN, while the temperature variables are calculated from actual temperature data. In the experiments, 3 resolution levels have been used for multiresolution representation.

**Table 1.** List of input variables for load forecasting models

Input	Variable name	Lagged value
1-5	Hourly load	1,2,3,24,168
6-10	Hourly temperature	1,2,3,24,168
11	Calendar variable	$\sin(2\pi t/24)$
12	Calendar variable	$\cos(2\pi t/24)$



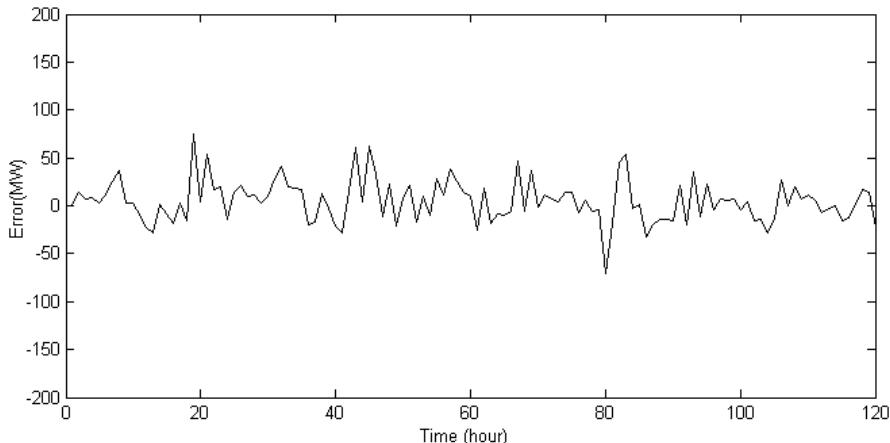
**Fig. 5.** 1-24 steps ahead of hourly forecasting performance in terms of MAPE



**Fig. 6.** Forecasting and real hourly load demand from January 10, 2002 to January 15, 2002

In order to perform the load forecasting for 1-24 hours ahead, the recursive forecasting method was employed. In the recursive forecasting method, the forecasted output is fed back as the input for the next time-unit forecasting and all other network inputs are shifted back one time unit. However, the future temperature is not available in practice when the recursive forecasting was performed. Therefore, it was necessary to estimate the temperature. In the experiments for this, the temperature was estimated from an average of the past temperature data.

All of the data used in the experiments were treated as normal working days. Holidays and anomalous days were not considered in this paper. The experiments



**Fig. 7.** Forecasting load errors from January 10, 2002 to January 15, 2002

were performed by traditional MLPNN, the BLRNN, and the M-BLRNN models. The performance of the M-BLRNN was evaluated and compared to other load forecasting schemes in terms of the mean absolute percentage error (MAPE).

Fig. 5 shows the performance over 1-24 hours ahead for the short-term load forecasting during the month of January 2002. As can be seen from Fig. 5, the M-BLRNN consistently outperformed the traditional MLPNN and the BLRNN. An improvement of the average MAPE of 53% and 35.5% over the MLPNN and the BLRNN, respectively, can be achieved. Figs. 6 and 7 depict the forecasting results and the forecasting errors, respectively, at each hour from January 10, 2002 to January 15, 2002 which had a typical load profile.

## 6 Conclusion

In this paper, a short-term load forecasting model using a wavelet-based neural network architecture termed the Multiscale BiLinear Recurrent Neural Network (M-BLRNN) is proposed. The short-term load forecasting was completed on the load data from the NAEU. By incorporating the wavelet transform in order to decompose the load profile into multiresolution levels and using the individual BLRNN model to forecast the sub-profiles at each level of resolution, the M-BLRNN was shown to outperform both the traditional MLPNN and the BLRNN in terms of the Mean Absolute Percentage Error (MAPE). These experiments and results show that an improvement of the average MAPE by 53% and 35.5% for the short-term load forecasting over the MLPNN and the BLRNN, respectively, was achieved. These are very encouraging results for applying the M-BLRNN to practical load forecasting problems.

## References

1. Park, D.C., El-Sharkawi, M.A., Marks II, R.J., Atlas, L.E., Damborg, M.J.: Electric Load Forecasting using An Artificial Neural Network. *IEEE Trans. on Power System* 6(2) (1991) 442-449
2. Papalexopoulos, A.D., Hesterberg, T.C.: A Regression-Based Approach to Short-Term System Load Forecasting. *IEEE Trans. on Power System* 5(4) (1990) 1535-1547
3. Hyde, O., Hodnett, P.F.: An Adaptable Automated Procedure for Short-Term Electricity Load Forecasting. *IEEE Trans. on Power System* 12(1) (1997) 84-94
4. Huang, S.J., Shih, K.R.: Short-Term Load Forecasting via ARMA Model Identification Including Non-Gaussian Process Considerations. *IEEE Trans. on Power System* 18(2) (2003) 673-679
5. Park,D.C.,Park,T.,Choi,S.: Short-Term Electric Load Forecasting using Recurrent Neural Network. *Proc. of ISAP'97*, (1997) 367-371
6. Taylor, J.W., Buizza, R.: Neural Network Load Forecasting With Weather Ensemble Predictions. *IEEE Trans. on Power System* (2002) 17(3) 626-632
7. Mohan Saini, L., Kumar Soni, M.: Artificial Neural Network-Based Peak Load Forecasting using Conjugate Gradient Methods. *IEEE Trans. on Power System* 17(3) (2002) 907-912
8. Hippert, H.S., Pedreira, C.E., Souza, R.C.: Neural Networks for Short-Term Load Forecasting: A Review and Evaluation. *IEEE Trans. on Power System* 16(1) (2001) 44-55
9. Mallat, S.G.: A Theory for Multiresolution Signal Decomposition: the Wavelet Representation. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 11(7) (1989) 674-693
10. Shensa, M.J.: The Discrete Wavelet Transform: Wedding the  $\hat{\Lambda}$  Trous and Mallat Algorithms. *IEEE Trans. on Signal Processing* 40(10) (1992) 2463-2482
11. Park, D.C., Zhu, Y.: Bilinear Recurrent Neural Network. *IEEE ICNN*, Vol. 3, (1994) 1459-1464
12. Park, D.C., Jeong, T.K.: Complex Bilinear Recurrent Neural Network for Equalization of a Satellite Channel. *IEEE Trans on Neural Networks* 13(3) (2002) 711-725
13. Renaud, O., Starck, J.L., Murtagh, F.: Wavelet-Based Combined Signal Filtering and Prediction. *IEEE Trans. on Systems, Man and Cybernetics* 35(6) (2005) 1241-251
14. Drezga, I., Rahman, S.: Input Variable Selection for ANN-based Short-Term Load Forecasting. *IEEE Trans. on Power System* 13(4) (1998) 1238-1244

# A Comparison of Selected Training Algorithms for Recurrent Neural Networks

Suwat Pattamavorakun<sup>1</sup> and Suwarin Pattamavorakun<sup>2</sup>

<sup>1</sup> Business Administration Faculty, Rajamangala University of Technology,  
Thanyaburi, Klong Six, Pathumthani, 12110, Thailand

suwat@rmut.ac.th, suwatpat@yahoo.com

<sup>2</sup> Science Faculty, Rajamangala University of Technology,  
Thanyaburi, Klong Six, Pathumthani, 12110, Thailand

suwarin@rmut.ac.th, suwarinp@yahoo.com

**Abstract.** Recurrent Neural Networks (RNNs) are one in which self-loops and backward weight connections between neurons are allowed. As a result of these network characteristics, recurrent networks can address temporal behaviors which not possible in feedforward neural networks, such as their behavior in the limit reaches a steady state (fixed point), an oscillation (limit cycle), and an a periodic instability (choas). Since RNNs have been increasingly applied to many dynamic system applications, there have been extensive efforts to develop a variety of architectures and training algorithms concerning on the enhancement of dynamic system characteristics. This work focuses on comparison of the selected and proposed training algorithms for RNNs. To evaluate the performance of the algorithms in the daily stock price forecasting in terms of efficiency index and computational time. A simple analysis on the complexity of RNNs was also carried out. It is noted that when comparing the speed of the algorithm, two components to be taken into account : the computation complexity and the space complexity.

## 1 Introduction

Recurrent Neural Networks (RNNs) can address potentially and richer temporal behaviors which are not possible in feed-forward networks (FNNs). Since a smaller RNN can often achieve the same accuracy as a large FNN, RNNs have better representation capabilities and therefore suit to be applied to temporal association networks. RNN can be divided mainly into two topologies; Fully Recurrent Neural Networks (FRNNs) and Partially Recurrent Neural Networks (PRNNs).

For sequence reproduction, a network must be able to generate the rest of a sequence from a part of that sequence (history). And in sequence recognition, a network creates a spatial pattern in response to a specific input sequence. “Layered networks are said to be fully connected if every node in each layer is connected to all the following layer nodes. If any of the connections is missing, then network is said to be partially connected” [1]. Although FRNN has been successfully applied to many applications, it has some weak points. One of the weak points is that the high requirement of time computation.

To tackle problems of neuron networks, learning is a key feature of systems in dynamic environments by adapting set of synaptic weights to approach the desired outputs with an error. Backpropagation of error is the most broadly used training algorithm that has been successfully applied to a variety of networks both in FNNs and RNNs. Nevertheless, learning with backpropagation is quite slow [2].

Moreover, there are several learning algorithms for training RNNs that have been recently proposed for increasing convergence speed and improving performance statistics. Some of the algorithms compute the gradient very efficiently, however, the main shortcoming of the algorithms is the excessive number of iterations needed to reach the minimum. Despite the potential and capability of RNNs, the main problem is the difficulty of training them, and the complexity and slow convergence of the existing training algorithms. It is desirable to make a comparison on the selected training algorithms for FRNNs. An evaluation of the performance and complexity analysis of some good training algorithms by applying RNN architectures to the financial forecasting problem. The study using the stock price data from three important companies list on the Stock Exchange of Thailand (SET).

## 2 Fully Recurrent Neural Networks (FRNNs)

FRNNs can be viewed as nonlinear dynamical systems because their connections can represent the dynamical complexity such as these three main dynamical behaviors (properties); their behavior in the limit reaches a steady state (fixed point), an oscillation (limit cycle), and a periodic instability (choas). These behaviors can be exploited for a useful purpose in neural networks [3].

## 3 Selected Training Algorithms

### 3.1 Notation

In an FRNN,  $m(n)-p-r$  topology, the total inputs pass directly with their connected weights and then obtain the outputs of both hidden and output nodes simultaneously. Let  $y(t)$  or  $y_t$  denote the  $n$ -tuple of outputs of the units in the network at time  $t$ , and let  $u(t)$  or  $u_t$  denote the  $m$ -tuple of external input signals to the network at time  $t$ . In a fully-connected RNN, each neuron in both the hidden and output layers has a feedback loop from the neuron itself and the recurrent links from other neurons. To distinguish the components of  $y$  representing unit output from those representing external input values, let  $\mathbf{O}$  denote the set of indices  $i$  such that  $y_i$  is the output of a unit in the network and  $\mathbf{I}$  denote the set of indices  $i$  for which  $y_i$  is an external input.

$$\begin{cases} y_i(t) = u_I(t), \quad i \in \mathbf{I} = 1, 2, \dots, m \\ y_i(t), \quad i \in \mathbf{O} = 1, 2, \dots, n \end{cases}$$

Let  $w_{ij}$  denote the weight on the connection to the  $i^{\text{th}}$  unit from either the  $j^{\text{th}}$  unit, if  $j \in \mathbf{O}$  or if  $j \in \mathbf{I}$ .

Let  $n = p + r$  and  $w_{ij}$  be the weight of link from node  $j$  to node  $i$ , where,

$$\begin{aligned} i &= 1, \dots, r, r+1, \dots, n = r+p; \text{ and} \\ j &= 1, \dots, r, r+1, \dots, n = r+p, n+1, \dots, n+m. \end{aligned}$$

### 3.2 Y-N and Atiya-Parlos Algorithms

The algorithm is derived using an algebraic method instead of a gradient approach by Yamamoto and Nikiforuk [4]. This algorithm, denoted Y-N algorithm, incorporates an Error Back Propagation (EBP) method for obtaining a fictitious target signal of output of the hidden nodes. The weight parameters are obtained by an Exponentially Weighted Least Squares (EWLS) method.

Atiya and Parlos [5] obtained the algorithm by approximating the error gradient. This is an online algorithm which assumes a small change in the network state and finds the direction of weight change by approximation. When minimizing the error, the error of hidden node is equal zero, the error of output node is the difference between desired and model output. The determination of the change in the weights will be according to the change in network state. The basic idea of Atiya-Parlos algorithm is to interchange the roles of the network states  $y(t)$  and the weight matrix  $\mathbf{W}$ . The states are considered as the control variables, and the change in weights is determined according to the change in  $y(t)$ .

### 3.3 First Modified (YNC) Algorithm

In this study the modified version of the Y-N algorithm was obtained by incorporating the Recursive Least Square (RLS) and Error Self Recurrent (ESR) into the Y-N algorithm. It updates the weights both in hidden nodes and output nodes in the same procedure while the Y-N algorithm updates the weights of hidden nodes and output nodes separately. Moreover, the errors calculated from the output unit are used as feedback for determining weight updates of output unit nodes.

The YNC algorithm can be summarized as follows:

1. Randomize all weights. For initialization, a choice of initial value is to take  $\mathbf{P}(0) = \delta \mathbf{I}$  or  $\mathbf{P}(0) = \mathbf{y}(0)\mathbf{y}(0)^T$ , where  $\delta$  is a small positive constant.
2. Present a training pattern pair  $(\mathbf{y}_i, \mathbf{d}_i)$  to the network where  $\mathbf{y}_i$  is the total input vector and  $\mathbf{d}_i$  is the desired output vector.
3. Calculate the model output  $\mathbf{y}_i$  for both hidden and output nodes. Usually, the sigmoid function is selected as an activation function:

$$s_i(t) = \sum_{l=1}^n w_{il} y_l(t) + \sum_{l=n+1}^{n+m} w_{il} u_l(t) = \sum_{l=1}^{n+m} w_{il} y_l(t) \quad (1)$$

For output node:

$$y_o(t+1) = f(s_o(t)) = \frac{1}{1 + \exp(-ps_o(t))} \quad (2)$$

For hidden node:

$$y_h(t+1) = f(s_h(t)) = \frac{1}{1 + \exp(-ps_h(t))} \quad (3)$$

4. Calculate the inverse function of the actual output  $d_i$  for finding an error signal  $e_o(t)$  by EBP method.

$$f^I(d_i(t+1)) = \frac{1}{\rho} \ln \left( \frac{d_i(t+1)}{1 - d_i(t+1)} \right) \quad (4)$$

$$e_o(t) = f^I(d(t+1)) - s_o(t) \quad (5)$$

5. Find a desired variation of hidden unit to obtain a fictitious target signal  $y_{ht}$ .

$$\Delta y(t) = \mathbf{W}_o (W_o^T W_o)^{-1} e_o(t) \quad (6)$$

A fictitious target signal  $y_{ht}$  is obtained as

$$y'_{ht}(t+1) = y_h(t+1) + \Delta y_h(t) \quad (7)$$

$$y_{ht}(t+1) = \mathbf{H} y'_{ht}(t+1) \quad (8)$$

Then calculate an error of hidden nodes.

$$e_h(t) = f^{-1}(y_{ht}(t+1)) - s_h(t) \quad (9)$$

6. Calculate a vector of error self-recurrent (ESR) for output and hidden nodes.

$$\mathbf{se}(t) = e(t) + \mu_o e(t-1) \quad (10)$$

where,  $\mathbf{se}(t) = [se_o(t), se_h(t), \dots, se_{(n)}(t)]$ , n-tuples are a number of output units.

7. Update weight by RLS method

$$\Delta \mathbf{W}(t) = \frac{P^{-1}(t)y(t)}{I + y^T(t)P^{-1}(t)y(t)} \cdot se(t) \quad (11)$$

$$\mathbf{P}^{-1}(t-1) = \frac{P^{-1}(t) - P^{-1}(t)y(t)y^T(t)P^{-1}(t)}{I + y^T(t)P^{-1}(t)y(t)} \quad (12)$$

$$\mathbf{W}(t+1) = \mathbf{W}(t) + \Delta \mathbf{W}(t) \quad (13)$$

8. Repeat steps 3-8 until end of data and stop training when the system error has reached an acceptable criterion.

### 3.4 Second Modified (YNS) Algorithm

Firstly, let consider the original Atiya algorithm which can be derived as the following:

The model output can be derived from the activation function of network inputs,  $net_i$ .

$$y(t+1) = f(net_i(t)) \quad (14)$$

From Eq.14, Assume

$$g(t+1) \equiv f(net_i(t)) - y(t+1) \text{ or } g(t+1) \equiv f(\mathbf{W}y(t)) - y(t+1) = 0 \quad (15)$$

Using inverse function, Eq.15 can be rewritten as

$$g'(t+1) \equiv \mathbf{W}y(t) - f^{-1}(y(t+1)) = 0 \quad (16)$$

Since Eq.16 is equal to zero, it can be expressed as

$$\frac{\partial g^*}{\partial w} \Delta w = -\frac{\partial g^*}{\partial y} \Delta y \quad (17)$$

Calculate the gradient of  $E$  respected to states  $y(t)$ , and assume a small change of states  $y(t)$  equals to negative direction of the gradient.

$$\Delta y(t) = -\eta \left( \frac{\partial E}{\partial y(t)} \right) \quad (18)$$

Equation 18 can be expressed as

$$\Delta y(t) = -\eta \varepsilon^T(t) \quad (19)$$

where  $\varepsilon^T(t)$  is equal to the following conditions.

$$\varepsilon_i(t) = \begin{cases} x_i(t) - d_i(t), & i \in O \\ 0, & \text{otherwise} \end{cases} \quad (20)$$

From Eq.17, the direction of weight change,  $\Delta w$  is required to be determined since it will result in changes of  $y(t)$  which needed to be as close as possible with the targeted changes,  $\Delta y(t)$ .

$$\Delta w = - \left[ \left( \frac{\partial g^*}{\partial w} \right)^T \left( \frac{\partial g^*}{\partial w} \right) \right]^{-1} \left( \frac{\partial g^*}{\partial w} \right)^T \frac{\partial g^*}{\partial y} \Delta y \quad (21)$$

The derivative of  $g^*$  with respect to  $w$  and  $y$  are  $\frac{\partial g^*}{\partial w} = \begin{pmatrix} Y(0) \\ Y(1) \\ \vdots \\ Y(t-1) \end{pmatrix}$

and

$$\frac{\partial g^*}{\partial y} = \begin{pmatrix} -D^{-1}(0) & 0 & \cdots & 0 \\ W & -D^{-1}(1) & & 0 \\ 0 & W & \ddots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & -D^{-1}(t-1) \end{pmatrix} \quad (23)$$

where

$$D(t) = \text{diag} \left( f \left( \sum_{j=1}^n w_{ij} y_j(t) \right) \right) \quad (24)$$

and

$$Y(t) = \begin{pmatrix} y^T(t) & 0 & 0 & \cdots & 0 \\ 0 & y^T(t) & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & y^T(t) \end{pmatrix} \quad (25)$$

From Eq.21, the inversion part,  $\left[ \left( \frac{\partial g'}{\partial w} \right)^T \left( \frac{\partial g'}{\partial w} \right) \right]^{-1}$  can be expressed as

$$\left[ \left( \frac{\partial g'}{\partial w} \right)^T \left( \frac{\partial g'}{\partial w} \right) \right]^{-1} = \begin{pmatrix} \left[ \sum_{t=0}^{T-1} y(t)y^T(t) \right]^{-1} & 0 & \dots & 0 \\ 0 & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \dots & \left[ \sum_{t=0}^{T-1} y(t)y^T(t) \right]^{-1} \end{pmatrix} \quad (26)$$

For on-line version, the weight update can be determined by

$$\Delta W(T) = \Delta W(T-1) + \eta \frac{\gamma(T)y^T(T-1)V^{-1}(T-1) - B(T-1)V^{-1}(T-1)y(T-1)[V^{-1}(T-1)y(T-1)]^T}{1 + y^T(T-1)V^{-1}(T-1)y(T-1)} \quad (27)$$

where

$$V^{-1}(T) = V^{-1}(T-1) - \frac{[V^{-1}(T-1)y(T-1)][V^{-1}(T-1)y(T-1)]^T}{1 + y^T(T-1)V^{-1}(T-1)y(T-1)}, \quad (28)$$

$$B(T) = B(T-1) + \gamma(T)y^T(T-1), \quad (29)$$

$$\gamma_i(t) = -\frac{1}{\eta} \frac{\partial g'}{\partial y_i} \Delta y_i \quad \text{or} \quad \gamma_i(T) = -D_i^{-1}(T-1)e_i^T(T) + We_i^T(T-1), \quad (30)$$

$$y^T(T-1) = \left( \frac{\partial g'}{\partial w} \right)^T \quad (31)$$

$$V^{-1}(T) = \left[ \left( \frac{\partial g'}{\partial w} \right)^T \left( \frac{\partial g'}{\partial w} \right) \right]^{-1} = \varepsilon I + \sum_{t=0}^T y(t)y^T(t), \quad (32)$$

and

$$D_i(t) = \text{diag} \left( f' \left( \sum_{j=1}^N w_{ij} y_j(t) \right) \right). \quad (33)$$

Secondly, the weight updates are adjusted in the following aspects, beginning with hidden weight connections.

$$\Delta W_H(T) = \Delta W_H(T-1) + \eta \frac{\gamma_H(T)y^T(T-1)V_H^{-1}(T-1) - B_H(T-1)V_H^{-1}(T-1)y(T-1)[V_H^{-1}(T-1)y(T-1)]^T}{1 + y^T(T-1)V_H^{-1}(T-1)y(T-1)} \quad (34)$$

Then, all new hidden outputs are calculated based on the updated hidden weights.

$$y_H^*(t) = W_H^T(t)y(t-1) \quad \text{and} \quad y^*(t)^T = (y_o(t)^T, y_H^*(t)^T, u(t)^T) \quad (35)$$

Finally, the network uses these new hidden outputs for determining the output node weight updates.

$$\Delta W_o(T) = \Delta W_o(T-1) + \eta \frac{\gamma_o(T)y^*(T-1)^T V_o^{-1}(T-1) - B_o(T-1)V_o^{-1}(T-1)y^*(T-1)[V_o^{-1}(T-1)y^*(T-1)]^T}{1 + y^*(T-1)^T V_o^{-1}(T-1)y^*(T-1)} \quad (36)$$

## 4 Complexity Analysis

In this comparative study of recurrent networks, the complexity of training algorithms is used. There are two aspects in complexity analysis, namely, space complexity and time complexity which are expressed in terms of the number of arithmetic operations required. For the analysis presented in this work, the additional computational effort required to update the weights is ignored for each learning algorithm. In addition, Big  $O$  notation is employed as the standard notation for describing the complexity which is considered mostly in the computational intensive part. There are two conventional learning algorithms and four selected training algorithms. (1) Back Propagation Through Time (BPTT) Algorithm [6]; (2) Real Time Recurrent Learning Rule (RTRL) Algorithm [7]; (3) Yamamoto to – Nikiforuk (Y-N) Algorithm; (4) Atiya – Parlors Algorithm; (5) The YNC Algorithm : The updating weights of this algorithm based on on-line fashion; therefore, the updated values of Eq.12 need to be updated and stored for using in the next step. It is clear that the space complexity is about  $O(w_U^2)$  per each time step. Because this algorithm has the processes for updating the network weight connections similar with Y-N algorithm that the fictitious signals are calculated before the network weight updates; therefore, the time complexity of this algorithm is about the same as Y-N algorithm which is  $O(w_H^3 p + w_O^3 r)$ . Nevertheless, this algorithm provides faster computational time than Y-N algorithm since the part of updating states of hidden nodes is not provided in this algorithm unlike Y-N algorithm. As a result, this algorithm takes the advantage of time consuming in this part over Y-N algorithm. (6) The YNS Algorithm : The complexities for updating weight connections in this algorithm can be counted at the same number of arithmetic operations as the complexities counted in Atiya-Parlors algorithm. This algorithm is only reorganized the weight update procedures in order to improve the net signals of hidden nodes, generating pre-image signals. This apparently is not involved and effected the complexities changes. Therefore, the space and time complexities are  $n^2 + n(r+1) + 2r$  and  $11n^2 + 2rn$  operations per data point respectively. Both can be considered as  $O(n^2)$  operations. For space complexity, the first term represents the space required for Eq.28. The second term represents the space required for Eq.29. The first and second terms are used to compute the weight updates in Eq.34 and Eq.36 respectively. The space required for storing input pattern is represented by the third term. For time complexity, it can be counted from the number of matrix-vector or vector-vector multiplications in Eq.30, Eq.34, Eq.36, and Eq.28.

The complexities of conventional and selected learning algorithms are summarized in Table 1.

**Table 1.** Order of magnitude of space and time requirements for the algorithms

Algorithm	Space Complexity	Time Complexity (per Time Step)
Epoch-wise BPTT	$O((m+n)h)$	$O(w_U h + w_A h)$
Real-Time BPTT	$O((m+n)L)$	$O((w_U + w_A)L/\Delta T)$
RTRL	$O(nw_A)$	$O(n^2 w_A)$
Y-N	$O(w_U^2)$	$O(w_H^3 p + w_O^3 r)$
Atiya-Parlos	$O((m+n)T)$	$O(n^2 T)$
Atiya-Parlos	$O(n^2)$	$O(n^2)$
YNC	$O(w_U^2)$	$O(w_H^3 p + w_O^3 r)$
YNS	$n^2 + n(r+1) + 2r$	$O(11n^2 + 2rn)$

Notes:

 $h$  = the amount of arithmetic operations.  $m$  = the number of external inputs. $n$  = the number of nodes.  $L$  = the training interval. $p$  = the number of hidden nodes.  $r$  = the number of output nodes. $T$  = the time index.  $w_A$  = the number of adaptable weights. $w_H = \{w_{ho}, w_{hh}, w_{hi}\}$ .  $w_O = \{w_{oo}, w_{oh}, w_{oi}\}$ . $w_U$  = the number of nonzero weights.

## 5 Empirical Study

### 5.1 Financial Data Series

*Stock Prices* in Thai Market are taken. All data were obtained from The Securities Exchange of Thailand (SET). All observed data series come from three main sectors of each group : Land and Houses Public Company Limited (LH) in real estate property sector, Bangkok Bank Public Company Limited (BBL) in banking sector, and Shin Corporation Public Company Limited (SHIN) in communication sector. Daily data on the stock prices and volumes in the market from 1993 to 1996 were used. The stock market is closed on weekend therefore consecutive data series are from Mondays to Fridays. In each case, the data series are divided into two phases, approximately 2/3 for calibration and 1/3 for validation, a calibration for training 1993-1994 and a validation for testing 1995-1996. Before being presented to the recurrent neural network, the data are transformed by a linear affine transformation to the range [0.05, 0.95] as following equation [8]:

$$y'_t = [0.9(y_t - y_{min}) / (y_{max} - y_{min})] + 0.05 \quad (37)$$

$$y_t = [(y_{max} - y_{min})(y'_t - 0.05) / 0.9] + y_{min} \quad (38)$$

where  $y'_t$  and  $y_t$  denote the transformed and original data, and  $y_{\max}$  and  $y_{\min}$  are the maximum and minimum of the original data in the calibration phase.

From the denotation  $m(n)-p-r$ , the fully-connected recurrent model used is 6(3)-2-1; 6 external inputs, 2 hidden and 1 output nodes in the output unit. The external input nodes consist of the past values of stock price (the amount of purchasing or selling debt instruments)  $P$ , and stock volume  $V$ . The feedback lines come from the previous outputs of both hidden and output nodes,  $P(k-1)$ . Therefore, the number of total inputs, 6(3) including previous output feedback, has 9 nodes.

The equation of recurrent model used is as follow: The number of external inputs = 6

$$\begin{aligned} P(t+1) = & g(P(t), P(t-1), P(t-2), V(t), V(t-1), V(t-2)) \\ & + g(P_{H1}(k-1), P_{H2}(k-1), P_O(k-1)) \end{aligned} \quad (39)$$

The number of previous outputs = 3

$t$  = daily data time (Mondays to Fridays),  $k$  = time step of the network

To compare the performance of algorithms, the same initial weights were used in the range of [0,1]. All feedback connection weights equal 1.0. The sigmoid slope ( $\rho$ ) of sigmoid function was set to 1 for easily. The forecasting lead-time is equal to one day.

When training the network by a chosen algorithm, it should be done step by step until the system error has reached an acceptable criterion. The stopping point ( $\epsilon$ ), determination of the optimal network structure, is based upon the relative difference of the sum of squared error ( $SE$ ) between the previous epoch and current iteration. The smaller the number of  $\epsilon$ , the better the neural network performs. It was chosen to be 0.05

$$\left| \frac{SE(t+1) - SE(t)}{SE(t)} \right| \leq \epsilon \quad (40)$$

where  $\epsilon$  is a constant that indicates the acceptable level of the algorithm and  $SE(t)$  denotes the value of  $SE$  at iteration  $t$ . The maximum number of iterations for the training process was limited to 20,000. All parameter values obtained in the training phase cannot be changed in validation phase. In order to evaluate the performance of the recurrent neural network, the following performance statistics are used.

Nash and Sutcliffe [9] proposed the **efficiency index (EI)** for measuring the performance of a given model:

$$EI = \frac{SR}{ST} \quad (41), \quad SR = ST - SE \quad (42), \quad ST = \sum_{i=1}^M (y_i - \bar{y})^2 \quad (43),$$

$$SE = \sum_{i=1}^M (y_i - \hat{y}_i)^2 \quad (44), \quad \bar{y} = \frac{1}{M} \sum_{i=1}^M y_i \quad (45),$$

where

$SR$  = Variation explained by the model,  $ST$  = Total variation,

$SE$  = Total sum of squared errors,  $y_i$  = Actual output (observed value) at time  $i$ ,

$\bar{y}$  = Mean value of the actual output,  $\hat{y}_i$  = Model output at time  $i$ ,

$M$  = Number of data points (training patterns) used.

EI is the indicator measuring the efficiency of the network. The closer the value is to 1 (or 100 percent), the better is the model representation.

The training parameters, for the Y-N algorithm was set the forgetting factor ( $\beta$ ) suitable at 0.99; for the Atiya-Parlos algorithm the most appropriate  $\eta$  value is 0.009 and  $\epsilon$  equals 13.0 for data of LH and SHIN, and 8.5 for BBL; for the YNC algorithm, the value of  $\mu_0$  was set to 1.0; for the YNS algorithm

LH		BBL		SHIN	
$\eta$	$\epsilon$	$\eta$	$\epsilon$	$\eta$	$\epsilon$
0.009	13.000	0.009	8.500	0.007	28.000

## 5.2 Comparison of Four Training Algorithms

Making a comparison, there are criteria in terms of:

- Efficiency index based upon FRNN  

$$\text{YNC} \geq \text{Y-N} \geq \text{YNS} > \text{Atiya-Parlos}$$

**Table 2.** Efficiency indices of four training with proposed algorithms for financial forecasts (Calibration Phase)

Stock Company	Y-N		Atiya-Parlos		YNC		YNS	
	EI	Time(s)	EI	Time(s)	EI	Time (s)	EI	Time(s)
LH	0.991	0.412	0.916	1.456	0.991	0.459	0.988	1.433
BBL	0.987	0.411	0.980	0.343	0.987	0.309	0.960	1.489
SHIN	0.932	0.501	0.871	0.162	0.966	0.302	0.937	1.427

**Table 3.** Efficiency indices of four training with proposed algorithms for financial forecasts (Validation Phase)

Stock Company	Y-N	Atiya-Parlos	YNC	YNS
LH	0.951	0.869	0.940	0.930
BBL	0.910	0.832	0.894	0.675
SHIN	0.983	0.839	0.970	0.790

- Computational time  

$$\text{Y-N} > \text{YNC} > \text{Atiya-Parlos} > \text{YNS}$$

The complexity of the selected training algorithms has analyzed. Table 2-3 show that the fastest speed of computation was Atiya-Parlos algorithm apart from the YNS algorithm. The second and third were YNC and Y-N algorithm.

## 6 Conclusions

The simulations were conducted to compare and evaluate the convergence behavior of the selected training algorithms on the fully recurrent neural network (FRNN) using daily data on stock prices in the Thai market. Space and time complexities are used to

assess the analysis of the selected training algorithms. The purpose of these algorithms is for fast training network. The main conclusions drawn from this study are as follows:

1. The fully connected recurrent networks with fast on-line training algorithms can work successfully in forecasting time series. Their performances achieved are very satisfactory for the stock prices data sets considered.
2. The YNC algorithm was found to be not so sensitive to initial weights. It can therefore solve the problem of selecting the learning parameter and can make a convergence fast.

The YNS algorithm can be satisfactorily used for forecasting. Due to the estimation of pre-image signals at the hidden layer which especially required large number of operations for FRNN.

## References

1. Koivo, H.N. : Neuro-Fuzzy Computing in Automation. Lecture Notes of Helsinki University of Technology, AZ-74.115 (2002)
2. Sutton, R.S. : Learning to Predict by the Methods of Temporal Differences. Machine Learning, Vol.3 (1986) 9-44
3. William, R., Zipser, D. : A Learning Algorithm for Continually Running Fully Recurrent Neural Networks. Neural Computation. 1(2) (1989) 270-280
4. Yamamoto, Y., Nikifuruk, P.N. : A Learning Algorithm for Recurrent Neural Networks and Its Application to Nonlinear Identification. IEEE International Symposium on Computer Aided Control System Design, Hawaii. (1999) 551-556
5. Atiya, A.F., Parlos, A.G. : New Result on Recurrent Network Training : Unifying the Algorithms and Accelerating Convergence. IEEE Transactions on Neural Networks. 11(3) 2000 697-709
6. Williams,R. : Complexity of Exact Gradient Computation Algorithms for Recurrent Neural Networks. Northeastern Univ. College Comp.Sci., Tech.Rep. NU-CCS-89-27 Boston MA (1989)
7. William, R., Zipser, D. : Experimental Analysis of the Real – Time Recurrent Learning Algorithm. Connection Science Vol.1. No.1 (1989)
8. Danh, N.T., Phien, H.N., Gupta, A.D. : Neural Network Models for River Flow Forecasting. Water SA. 25(1) (1999) 33-39
9. Nash, J.E., Sutcliffe, J.V. : River Flow Forecasting through Conceptual Models. J. Hydrology 10(3) (1970) 282-290

# Neural Network Recognition of Scanning Electron Microscope Image for Plasma Diagnosis

Byungwhan Kim<sup>1</sup>, Wooram Ko<sup>1</sup>, and Seung Soo Han<sup>2</sup>

<sup>1</sup> Department of Electronic Engineering,  
Sejong University, Seoul, Korea  
kbwhan@sejong.ac.kr

<sup>2</sup> Department of Information Engineering,  
Myongji University, Yongin, Korea  
shan@mju.ac.kr

**Abstract.** To improve equipment throughput and device yield, a malfunction in plasma equipment should be accurately diagnosed. An ex-situ diagnosis model was constructed by applying neural network to scanning electron microscope (SEM) image of plasma-etched patterns. The experimental data were collected from a plasma etching of tungsten thin films. Faults in plasma were generated by simulating a variation in process parameters. Feature vectors were obtained by applying direct and wavelet techniques to SEM images. The wavelet techniques generated three feature vectors composed of detailed components. The diagnosis models constructed were evaluated in terms of the recognition and diagnosis accuracies. The direct technique yielded much smaller recognition accuracy with respect to the wavelet technique. The improvement was about 82%. For the diagnosis accuracy, the improvement was about 30%. All these results demonstrate that the direct method is a more effective feature extraction in constructing a SEM-based neural network diagnosis model.

## 1 Introduction

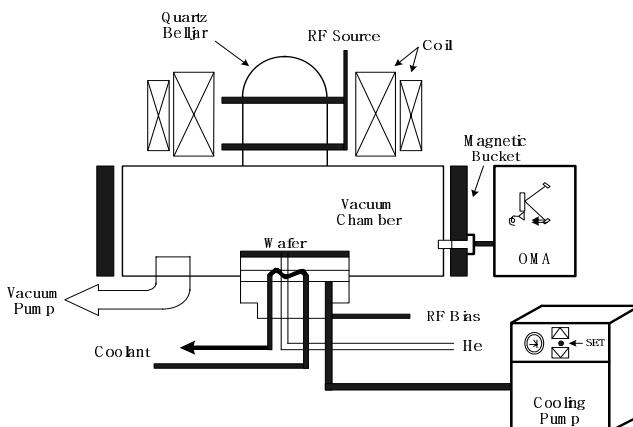
In manufacturing integrated circuits, plasmas play a crucial role in depositing or etching thin films. To maintain device yield and equipment throughput, plasma states should be tightly monitored and diagnosed in case a fault occurs. Monitoring plasmas were conducted by means of in-situ diagnostic sensors, including optical emission spectroscopy [1], radio frequency impedance sensor [2], and radio frequency (RF) match monitor system [3]. Apart from the in-situ sensor data, information remaining on plasma-processed film surface may provide detailed clues to plasma diagnosis. In the context of plasma etching, the ex-situ measurements are frequently collected by the scanning electron microscope (SEM). The SEM measures the depth of deposited thin films as well as the sidewall feature of etched patterns. Previously, neural network was applied to film surface information to identify a fault cause [4]. Unfortunately, this approach is limited in that the number of surface measurements was very small. With this insufficient information, neural network is not likely well trained, resulting in a poor recognition and prediction performance. Another feature to circumvent this limitation is the profile sidewall. This is attributed to the fact that the sidewall feature is unique to plasma condition. This attractive feature was examined

from the standpoint of plasma monitoring [5]. However, a recognition model of SEM profile image for plasma diagnosis has not been investigated.

In this study, a new recognition model of SEM is presented. This type of model can be used to check plasma states on a run-to-run wafer level. For this, information regarding the profile sidewall was extracted in direct and discrete wavelet transformation (DWT) [6] techniques. Applying the DWT led to three types of detailed feature information. Therefore, a total of four feature information was evaluated. As a pattern recognizer, the backpropagation neural network (BPNN) [7] was used. Previously, intelligent techniques including the BPNN were applied to predict various type of plasma data [8-11]. The performance of BPNN recognition model is evaluated in terms of prediction and recognition accuracies. The experimental data examined were collected during a plasma etching tungsten metal in a SF<sub>6</sub> plasma.

## 2 Experimental Data

Schematic of a Pinnacle 8000 helicon plasma etch system is shown in Fig. 1. A high density plasma is produced in a source quartz by coupling 13.56 MHz power to an antenna via a matching network. The coil surrounding the quartz provides an efficient transfer of energy into the center of the plasma as well as an effective confinement of the plasma, thereby increasing the plasma density. A magnetic bucket encircling the process chamber comprised 24 permanent magnets with alternating polarity, yielding a highly uniform plasma.



**Fig. 1.** Schematic of plasma etch equipment

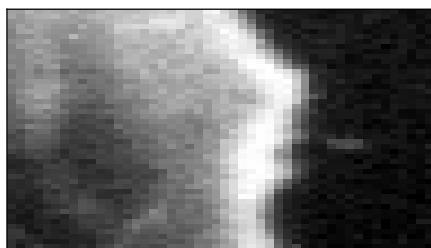
Test patterns were fabricated on 8-inch diameter silicon wafers of (100) orientation. The pattern was structured as a 3500 Å W with a 600 Å TiN diffusion barrier layer and a 500 Å Ti adhesive film on a 4000 Å silicon dioxide film. On the top of the W layer, a 300 Å TiN/5000 Å Al film was deposited. A deep ultra-violet photoresist was spun by about 1 μm thickness and baked for 30 min. at 120°C. The etching was

conducted in a SF<sub>6</sub> helicon plasma. Faults in the plasma were simulated with the variations in the process parameters, including the radio frequency source power, bias power, substrate temperature, and SF<sub>6</sub> flow rate. In other words, the fault was generated by varying a process parameter from a reference condition. The reference condition was set at 0 W bias power, -20 °C temperature, 100 sccm SF<sub>6</sub> flow rate, and 2000 W source power. From the reference condition, more specifically, the bias power was varied to 0 or 30 W, temperature to -50 or 10 °C, SF<sub>6</sub> flow rate to 80 or 120 sccm, and source power only to 1500 W. Therefore, a total of 7 faults were simulated.

### 3 Feature Extraction

#### 3.1 Direct Method

In the direct method, an original SEM photo shown in Fig. 2 was first scanned. The gray code of the scanned image is then changed into the bitmap version. To extract the contour line, the non-varying DC component is removed and the resulting image is depicted in Fig. 3. The detailed quantitative information on the profile sidewall is obtained by reading the DC-removed image. This feature consisted of 48 numerical values.



**Fig. 2.** SEM profile image obtained at 0 W bias power



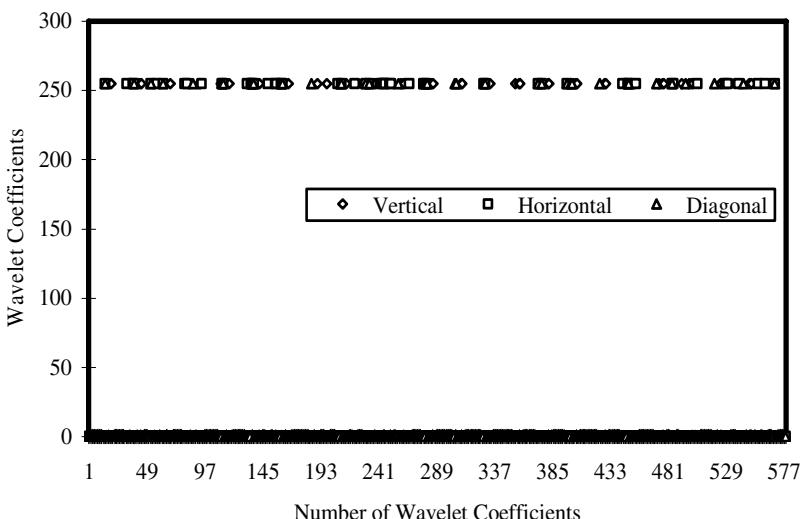
**Fig. 3.** The contour line without the DC component

#### 3.2 Wavelet Transformation

The DWT has an efficient space-frequency localization, which enables us to examine the local variations of a signal or image at multiple levels. For an image such as a SEM photo concerned here, meanwhile, the DWT is successively conducted by a pair

of two low and high pass filters, separately in row and column directions. Filtering each row of an image array produces high pass and low pass sub-image. Subsequently, each sub-image is separately filtered column-wise, resulting in 4 sub-images corresponding to low-low-pass, low-high-pass, high-low-pass, and high-high pass row column filtering, respectively. This procedure was detailed in [6].

Applying the DWT to the SEM image results in the four blocks, including one approximation and three detailed components. As the transformation factors, the scale level of one and the Daubechies function of type one were employed. Using the histogram method, a profile contour was extracted from the approximation component. The DWT was then applied to the extracted profile contour and the resulting 576 wavelet coefficients are shown in Fig. 4. Closely looking at the wavelet coefficient distributions reveals that they are different from each other. The uniqueness in the distribution patterns was also observed for the variations in the process parameters. These evidences indicate that the wavelet distribution patterns can be effectively used to characterize a fault in a plasma processing system.



**Fig. 4.** Distribution patterns of wavelet coefficients corresponding to the three types of components  
4 Evaluation of Recognition Model

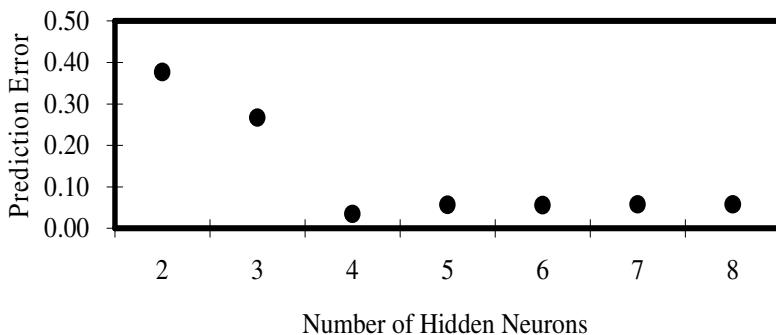
## 4 Diagnosis

### 4.1 Recognition Accuracy

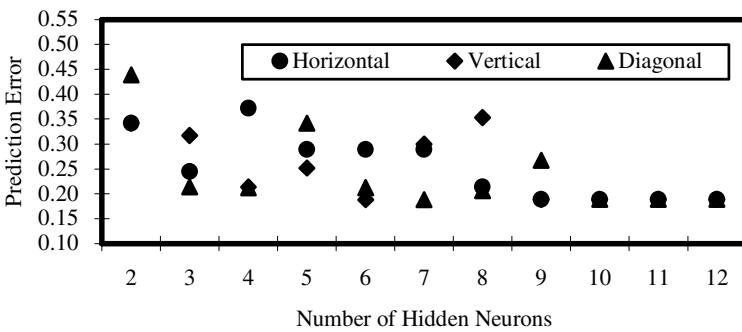
To build a fault diagnosis model, the BPNN with one single hidden layer was adopted. The inputs to the BPNN consisted of 48 and 576 for the feature vectors extracted by the direct and wavelet techniques, respectively. The number of output neurons was four, each corresponding to one of process parameters. As the training algorithm, the generalized delta rule [7] was employed. Only the number of hidden neurons was varied from 2 to 8. The other training factors were set to their default

values. The recognition accuracy of the direct method-applied model is plotted in Fig. 5. Here the accuracy was measured by the root mean-squared error. As shown in Fig. 5, the smallest RMSE is obtained at 4 hidden neurons and it is about 0.0344.

Next, the recognition accuracies of the models trained with three wavelet components are shown in Fig. 5. The neuron number was increased to 12 since each feature vector was composed of much more elements than that for the direct method. As shown in Fig. 5, the variation in the recognition accuracy is considerable for the hidden neurons less than nine. In contrast, the variation is little for large number of hidden neurons. As depicted in Fig. 5, the smallest RMSE is about 0.189, identical at 10, 11, 12 hidden neurons for each wavelet model. Compared to this error, the error determined earlier for the direct method yields an improvement of about 82%. This indicates that the direct method is more effective to building a neural network recognition model of SEM image.



**Fig. 5.** Recognition accuracy of the model constructed with the direct method



**Fig. 6.** Recognition accuracy of the model constructed with the wavelet method

#### 4.2 Diagnosis Accuracy

Diagnosis models are evaluated in terms of the number of identified faults (NIF). In the presence of the threshold, the NIF is defined as the number of patterns satisfying

$$|d_{ij} - out_{ij}| < \text{Threshold} \quad (1)$$

where  $d_{ij}$  and  $out_{ij}$  represent the actual and predicted outputs at the  $i$ th output neuron for the  $j$ th test pattern. The threshold plays a role of adjusting the severity of monitoring conditions. A smaller threshold represents a stricter monitoring condition. The threshold was varied from 0.1 to 0.5 with an increment of 0.1. Another measure of diagnosis is the accumulated NIF (ANIF). The ANIF is simply the sum of NIFs computed at all five thresholds for a given neuron. This measure can evaluate the total diagnostic efficacy of a model for all severities in the monitoring conditions.

**Table 1.** NIF of direct diagnosis model as a function of hidden neurons

Hidden Neurons	0.1	0.2	0.3	0.4	0.5
2	1	1	3	3	3
3	1	3	3	5	7
4	7	7	7	7	7
5	6	7	7	7	7
6	6	7	7	7	7
7	6	6	7	7	7
8	7	7	7	7	7

**Table 2.** NIF of horizontal models as a function of hidden neurons

Hidden Neurons	0.1	0.2	0.3	0.4	0.5
2	0	0	0	2	4
3	4	4	4	4	5
4	0	0	0	0	4
5	5	5	5	5	6
6	5	5	5	5	7
7	5	5	5	5	7
8	4	4	4	6	6
9	5	5	5	5	7
10	5	5	5	5	7
11	5	5	5	5	7
12	5	5	5	5	7

First, the NIFs calculated for the direct method are shown in Table 1. As shown in Table 1, a perfect recognition is obtained at 4 or 8 hidden neurons at all thresholds. Due to the smaller RMSE at 4 hidden neurons, it is determined to be the best model for the direct method. Next, the NIFs for the horizontal model were calculated and they are shown in Table 2. The NIF data for other remaining two models are not presented since they are similar to that for the horizontal model. Their best performances were even identical to that for the horizontal model determined later. Table 2 reveals that better models are obtained at 6, 7, 9, 10, 11, or 12 hidden neurons. The NIF for each of these models is identical at each threshold. The difference among the techniques becomes more transparent as the ANIFs calculated for the determined models

are compared. Actually, the ANIFs for the direct and wavelet techniques are 35 and 27. This demonstrates that the improvement of the direct technique over the wavelet technique is about 30%. In consequence, the direct technique yielded the smallest recognition error and highest diagnosis accuracy.

## 5 Conclusion

A new ex-situ diagnosis model was constructed by recognizing SEM profile patterns using neural network. Feature vectors for SEM images were extracted by the direct and wavelet techniques. Compared to the wavelet technique, the direct technique provided a smaller recognition error as well as a higher diagnosis. The diagnosis model presented here can be used to check equipment plasma state on the basis of run-to-run wafer processing. Moreover, the diagnosis model can be used to identify a fault cause once equipment faults and the corresponding causes were recognized. It is expected that by integrating the presented SEM-based ex-situ diagnosis into current in-situ diagnosis a fault in plasma equipment could be more strictly diagnosed.

## Acknowledgements

This study was supported by “System IC 2010” project of Korea Ministry of Commerce, Industry, and Energy, and partly by the MIC (Ministry of Information and Communication), Korea, under the ITRC (Information Technology Research Center) support program supervised by the IITA (Institute of Information Technology Assessment).

## References

1. Stevenson, J.O., Ward P.P., Smith, M.L., Markle R.J.: A Plasma Process Monitor/Control System. *Surf Interf Anal* 26 (1998) 124-133
2. Bose, F., Patrick, R. Baltes H.: Measurement of Discharge Impedance for Dry Etch Process Control. *SPIE* 2336 (1994) 101-110
3. Kim, B., Lee C.: Monitoring Plasma Impedance Match Characteristics in a Multipole Inductively Coupled Plasma for Process Control. *J Vac Sci Technol A* 18(1) (2000) 58-62
4. Kim, B., May, G.S.: Real-Time Diagnosis of Semiconductor Manufacturing Equipment Using a Hybrid Neural Network Expert System. *IEEE Trans Comp Packag Manufact Technol* 20(1) (1997) 39-47
5. Kim, B., Choi, W., Lim, M.T.: Wavelet Monitoring of Plasma Etching. *J Vac Sci Technol B* 21(6) (2003) 2329-2333
6. Prasad, L., Iyengar, S.S.: *Wavelet Analysis with Applications to Image Processing*. CRC Press (1997)
7. Rummelhart, D.E., McClelland, J. L.: *Parallel Distributed Processing*. MIT Press, Cambridge (1986)
8. Kim, B., Park, J.H.: Qualitative Fuzzy Logic Model of Plasma Etching Process. *IEEE Trans Plasma Sci.* 30(2) (2002) 673-678

9. Kim, B., Han, S.S., Kim, T.S., Kim, B.S., Shim, I.J.: Modeling refraction characteristics of silicon nitride film deposited in a SiH<sub>4</sub>/NH<sub>3</sub>/N<sub>2</sub> plasma using neural network. *IEEE Trans Plasma Sci* 31(3) (2003) 317-323
10. Kim, B., Kwon K.H., Kwon, S.K., Park, J.M., Yoo, S.W., Park K.S., You I.K., Kim, B.W.: Modeling etch rate and uniformity of oxide via etching in a CHF<sub>3</sub>/CF<sub>4</sub> plasma using neural networks. *Thin Solid Films* 426(1-2) (2003) 8-15
11. Kim, B., Kim, S., Kim, K.: Modeling of Plasma Etching Using a Generalized Regression Neural Network. *Vacuum* 71(4) (2003) 497-503

# A New Multi-constrained QoS Routing Algorithm in Mobile Ad Hoc Networks

Hu Bin<sup>1</sup> and Liu Hui<sup>2</sup>

<sup>1</sup> Information Center of Shandong Labour and Social Security Department, Jinan,  
Shandong, 250002, P.R. China  
hb@ldt.sd.gov.cn

<sup>2</sup> College of Mechanical and Electronic Engineering, Shandong University of Architecture,  
Jinan, Shandong, 250101, P.R. China  
avery1@sohu.com

**Abstract.** The task of QoS routing is to find a route in the network, which has sufficient resources to satisfy the constraints on delay, delay-jitter, cost, etc. The delay-cost-constrained routing problem is NP-complete. In this paper we present a new method of multi-constrained routing based on Hopfield neural network (HNN) that solves the optimum routing problem for supporting QoS in Ad hoc networks, which is called MCADHP. The idea of MCADHP is to reduce the NP-complete problem to a simpler one, which can be solved in polynomial time in order that the neural network could be robust to the changed of the network topology. Under this assumption the general principles involved in the design of the proposed neural network and the method regarding the relationships of different coefficients in the energy function are discussed. The performance of the proposed neural model is studied by both theoretical analysis and computer simulations. Simulation results indicate that our proposed scheme is very effective and outperforms previous schemes.

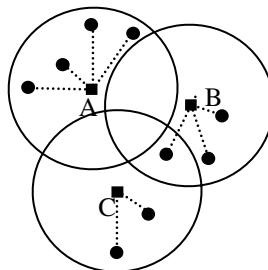
## 1 Introduction

A wireless ad-hoc network is self-organizing, rapidly deployable, and without fixed infrastructure. Due to the lack of fixed infrastructure, the development of efficient protocols to support the various networking operations (e.g. Quality of Service (QoS) support, etc.) presents many issues and challenges. Recently there have been many reports on QoS efforts in ad hoc networks [1], which have focused on designing QoS routing algorithm. The QoS routing algorithms are constraint-based routing method and depend on the metrics chosen for the routes. Common route metrics in constraint-based routing can be divided into three classes, which are additive, multiplicative and concave. According to this classification, metrics *delay*, *jitter*, and *cost* are additive, *reliability* is multiplicative, and *bandwidth* is concave [2]. The multi-constrained routing problem is difficult because different constraints can conflict with one another. In particular, the delay-cost-constrained routing, i.e., finding a route between two nodes in the network with both end-to-end delay and end-to-end cost bounds, can be formalized as a multi-constrained path problem (MCP), which is NP-complete [3].

Neural networks have often been formulated to solve difficult (e.g. NP-complete) optimization problems. Hopfield and Tank [4] initially demonstrated the computational power of the neural network. The advantage of the HNN is the rapid computational capability of solving the combinatorial optimization problem. In particular, Ahn and Ramakrishna [5] proposed a near-optimal routing algorithm employing a modified HNN. Dongfeng Yuan and Jian Liu [6] have proposed a HNN based QoS routing algorithm in mobile Ad hoc networks under minimum cost constraint. In this paper, a new multi-constrained routing method based on HNN model, which is called MCADHP, is proposed in Ad hoc networks to speed up convergence under multi-constraints, namely the delay and cost.

## 2 Multi-constrained QoS Routing Algorithm Based on Hopfield Neural Network

In a large scale of Ad hoc network the cluster approaches are usually employed, in which all nodes are grouped into clusters. A cluster is a subset of nodes, which can communicate with a clusterhead and possibly with each other. In Fig. 1, nodes A, B and C are clusterheads for their coverage area respectively. Each of them serves as a regional broadcast node, and as a local coordinator to enhance channel throughput.



**Fig. 1.** Example of Clustering Architecture in Ad hoc network

Multi-constrained Ad hoc network path problem (MCADHP) can be described by the directed graph  $\mathbf{G} = (\mathbf{V}, \mathbf{E})$ , where  $\mathbf{V}$  is a set of vertices ( $N$  nodes) and  $\mathbf{E}$  is a set of its edges (links) [7]. A path inclusion criterion matrix, denoted by  $\alpha = [\alpha_{ij}]$ , represents the output values of neurons and each element is defined as follows:

$$\alpha_{ij} = \begin{cases} 1 & \text{if link from node } i \text{ to node } j \text{ exists in routing path} \\ 0 & \text{otherwise} \end{cases}$$

The link connection indicator matrix  $\beta = [\beta_{ij}]$  is defined as follows:

$$\beta_{ij} = \begin{cases} 1 & \text{if link from node } i \text{ to node } j \text{ does not exist in routing path} \\ 0 & \text{otherwise} \end{cases}$$

Multi-constrained Ad Hoc path problem (MCADHP) can be denoted as MCADHP ( $G, s, d, \omega_1, \omega_2, C_1, C_2$ ), in which vector  $s$  is called source nodes,  $d$  is destination nodes,  $\omega_1$  and  $\omega_2$  are weight functions and  $C_1$  and  $C_2$  are two constants. A solution for MCADHP ( $G, s, d, \omega_1, \omega_2, C_1, C_2$ ) is to find a path  $p$  from  $s$  to  $d$  such that  $\omega_1(p) \leq C_1$  and  $\omega_2(p) \leq C_2$  if such a path exists.

Since MCADHP ( $G, s, d, \omega_1, \omega_2, C_1, C_2$ ) is NP-complete, we introduce a new method to solve this problem. The algorithm can be represented as follows

There are the two constants, which are delay  $t_{ij}$  and cost  $c_{ij}$ , associated with each link  $(i, j)$  and they are specified by the delay matrix  $T = [t_{ij}]$  and the cost matrix  $C = [c_{ij}]$ . It is obvious that all the diagonal elements of  $\alpha_{ij}$  must be zero. Then the expression of the delay and cost from source to destination node can be given by

$$\Gamma_{\text{delay}} = \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N t_{ij} \alpha_{ij} \quad (t_{ij} \geq 0) \quad (1)$$

$$\Psi_{\text{cost}} = \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N c_{ij} \alpha_{ij} \quad (c_{ij} \geq 0) \quad (2)$$

The delay-cost-constrained routing problem is to find a path  $p$  from  $s$  to  $d$  such that  $\text{delay}(p) \leq \text{Delay}$  and  $\text{cost}(p) \leq \text{Cost}$ , where  $\text{Delay}$  and  $\text{Cost}$  are the required end-to-end delay bound and cost bound, respectively. It can be represented as follows: create two new functions *new-delay* and *new-cost*.

$$\Gamma_{\text{new-delay}} = \left\lceil \frac{\Gamma_{\text{delay}} \cdot x}{\text{Delay}} \right\rceil \quad (3)$$

$$\Psi_{\text{new-cost}} = \left\lceil \frac{\Psi_{\text{cost}} \cdot x}{\text{Cost}} \right\rceil \quad (4)$$

Where  $x = \text{coef} \times d_{s,t}$   $\text{coef}$  is a given positive integer and  $d_{s,t}$  is the distance from  $s$  to  $d$ . We reduce the original problem MCADHP ( $G, s, d, \omega_1, \omega_2, \text{Delay}, \text{Cost}$ ) to two simpler problems, MCADHP( $G, s, d, \omega_1, \omega_2, \text{Delay}, x$ ) and MCADHP( $G, s, d, \omega_1, \omega_2, x, \text{Cost}$ ). The theorems [8] guarantee that a solution for the simpler problem must be a solution for the original problem.

By using the definitions above and assuming the maximum delay in Ad hoc network to be  $T_0$ , the delay-cost-constrained QoS routing problem in Ad hoc network can be represented as constrained combinatorial optimization problem as follows

$$\begin{aligned}
\text{minimize} \quad & \Psi_{\text{new-cost}} = \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N c'_{ij} \alpha_{ij} \\
\text{subject to} \quad & \Gamma_{\text{new-delay}} = \sum_{i=1}^N \left( \sum_{\substack{j=1 \\ j \neq i}}^N t'_{ij} \alpha_{ij} - \sum_{\substack{j=1 \\ j \neq i}}^N t'_{ij} \alpha_{ji} \right) \leq T_0 \quad \alpha_{ij} \in \{0,1\}
\end{aligned} \tag{5}$$

To formulate this optimization problem in terms of the HNN, the computational network requires  $N \times (N-1)$  neurons since it is organized in an  $(N \times N)$  matrix with all diagonal elements removed. Defining the Lyapunov (energy) function as equation (6), its minimization drives each neuron of the neural network into its stable state, thereby providing the routing algorithm solution:

$$\begin{aligned}
E = & \frac{\mu_1}{2} \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N c'_{ij} \cdot \alpha_{ij} + \frac{\mu_2}{2} \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N \beta_{ij} \cdot \alpha_{ij} + \frac{\mu_3}{2} \sum_{i=1}^N \left( \sum_{\substack{j=1 \\ j \neq i}}^N \alpha_{ij} - \sum_{\substack{j=1 \\ j \neq i}}^N \alpha_{ij} - \chi_i \right) \\
& + \frac{\mu_4}{2} \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N \alpha_{ij} \cdot (1 - \alpha_{ij}) + \frac{\mu_5}{2} \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N \alpha_{ij} \cdot \alpha_{ji} + \frac{\mu_6}{2} \left( \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N \left( \sum_{k=1}^N \alpha_{ik} - 1 \right) \cdot \alpha_{ij}^2 \right) \\
& + \frac{\mu_7}{2} \left( \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N \left( \sum_{k=1}^N \alpha_{kj} - 1 \right) \cdot \alpha_{ij}^2 \right) + \frac{\mu_8}{2} \left( T_0 - \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N t'_{ij} \cdot \alpha_{ij} \right) \\
& \left( \left( T_0 - \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N t'_{ij} \cdot \alpha_{ij} \right) - \left| T_0 - \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N t'_{ij} \cdot \alpha_{ij} \right| \right)
\end{aligned} \tag{6}$$

$$\text{where } \chi_i = \begin{cases} 1 & \text{if } i = s \\ -1 & \text{if } i = d \\ 0 & \text{if } i \neq s, d \end{cases}$$

In equation (6), the  $\mu_1$  term minimizes the total link cost  $\Psi_{\text{new-cost}}$  of a routing path by taking into account the cost of all existing links; the  $\mu_2$  term prevents nonexistent links from being included in the chosen routing path; the  $\mu_3$  term oversees constraint satisfaction and maintains the equality of the number of incoming and outgoing links; the  $\mu_4$  term drives the neurons towards convergence to a valid route consisting of connected nodes; the  $\mu_5$  term was proposed to guarantee that the flow vector is always directed towards the destination; the  $\mu_6$  and  $\mu_7$  terms are added to eliminate the possibility of loops or partitions in a path. The  $\mu_8$  term is presented from the constraint to minimize the delay  $\Gamma_{\text{new-delay}}$ . These terms also nudge the neurons towards speedy and precise convergence to an optimal solution.

The motion equation of the HNN is obtained:

$$\frac{dy_{ij}}{dt} = -\frac{y_{ij}}{\tau} + \sum_{k=1}^N \sum_{\substack{l=1 \\ l \neq k}}^N (\omega_{ij,kl}^1 \alpha_{kl} + \omega_{ij,kl}^2 \alpha_{kl} \alpha_{ij}) + h_{ij} \quad (7)$$

where  $\tau$  is the time constant of the circuit, and the connection weights  $\omega$  and the biases  $h$  are given as:

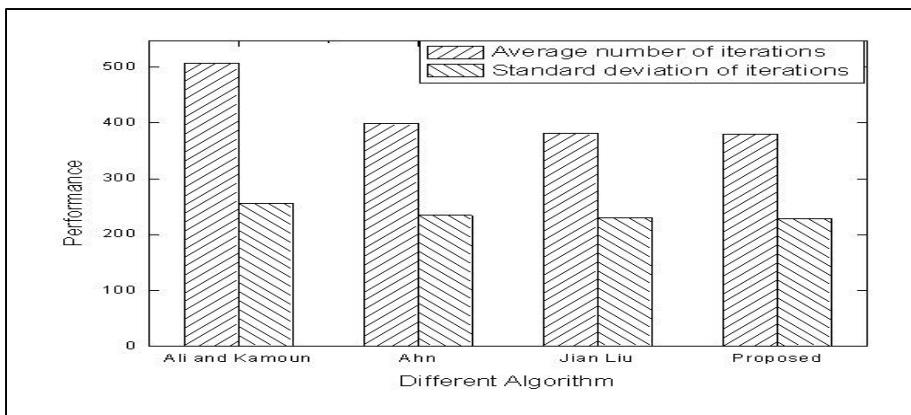
$$\begin{aligned} \omega_{ij,kl}^1 &= \mu_3 \delta_{jk} + \mu_3 \delta_{li} - \mu_3 \delta_{ik} - \mu_3 \delta_{jl} + \mu_4 \delta_{ik} \delta_{jl} - \frac{\mu_5}{2} \delta_{jk} \delta_{il} \\ &\quad - \mu_6 \delta_{ik} \delta_{jl} - \mu_7 \delta_{ik} \delta_{jl} - \mu_8 \delta_{ik} \delta_{jl} \end{aligned} \quad (8)$$

$$\omega_{ij,kl}^2 = -\mu_6 \delta_{ik} - \mu_7 \delta_{lj} \quad h_{ij} = -\frac{\mu_1}{2} c_{ij} - \frac{\mu_2}{2} \beta_{ij} + \mu_3 \chi_i - \mu_3 \chi_j - \frac{\mu_4}{2}$$

where  $\delta_{ij}$  is the Kronecker delta function. The HNN finds the optimal Qos routing between two nodes by properly adjusting the input biases in (8).

### 3 Performance Evaluation

The simulation program is written in C++. The simulation is used to compare the performance of the proposed algorithm with previous ones for a 30-node network. Fig. 2 compares four algorithms. A total of 10000 random network topologies with varying sizes and randomly assigned link costs were investigated. The proposed algorithm converges to a stable state in about 380 iterations, about 25% improvement over the Ali and Kamoun's algorithms and about 5% improvement over Ahn and Jian.

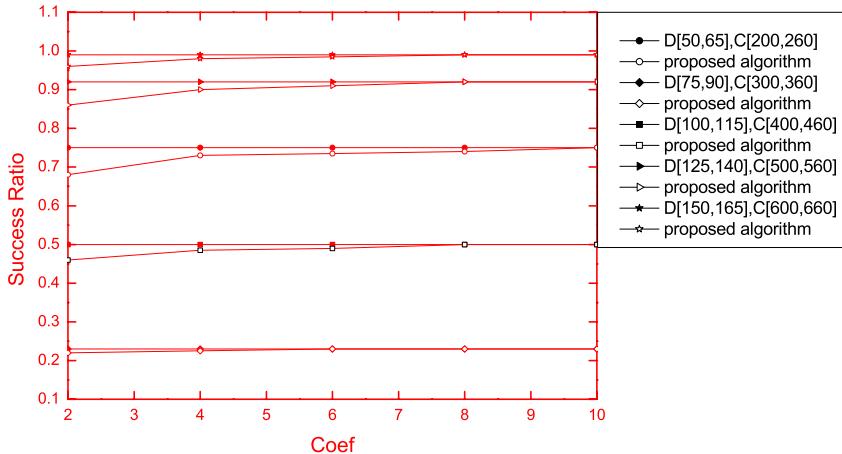


**Fig. 2.** Performance comparison of four algorithms

The other performance metric we considered was *success ratio*.

$$\text{success\_ratio} = \frac{\text{number\_of\_requests\_successfully\_routed}}{\text{total\_number\_of\_routing\_requests}}$$

Fig. 3 showed the success ratio with respect to coef, *Delay* and *Cost*. In Fig. 3 we can observe the larger the value of coef, the higher success ratio. The smaller the values of *Delay* and *Cost*, the tighter the constraints of a routing request, which leads to a lower success ratio.



**Fig. 3.** Success ratio with respect to coef with different Delay and Cost in Proposed rouging algorithm and optimum one

## 4 Conclusions

In this paper we have proposed a new multi-constrained routing algorithm based on Hopfield neural network (HNN) in mobile Ad hoc networks. The algorithm reduces the problem MCADHP ( $G, s, d, \omega_1, \omega_2, C_1, C_2$ ) to a simpler one MCADHP ( $G, s, d, \omega_1, \omega_2, C_1, x$ ), and then finds a solution for the new problem.

Simulation results show that the proposed multi-constrained QoS routing algorithm converges to a stable state was seen to occur in about 380 iterations that is a performance improvement over previous algorithms. It also showed that higher performance of the algorithm could be achieved at the expense of higher overhead.

## References

1. Yang, X., Haizhon, L.: Local data control and admission control for QoS support in wireless ad hoc networks. IEEE Transactions on Vehicular Technology (2004), Vol. 53, Issue. 5, 1558 – 1572
2. Xipeng X., Ni, L.M.: Internet QoS: a big picture. IEEE Network (1999), Vol. 13, Issue. 2, 8 – 18
3. Wand, Z., Crowcroft, J.: QoS routing for supporting multimedia applications. IEEE Journal on Selected Areas in Communications (1996), Vol. 14, Issue. 7, 1228 - 1234
4. Hopfield, J. J., Tank, D.W.: Neural Computations of Decision in Optimization Problems. Biological Cybernetics (1986), Vol. 52, 141-152

5. Ahn, C.W., Ramakrishna, R.S., Kang, C.G., Choi, I.C.: Shortest path routing algorithm using Hopfield neural network. *Electronics Letters* (2001), Vol. 37, Issue. 19, 1176 – 1178
6. Jian, L., Dongfeng, Y., Song, C., Yingji, Zh.: A new QoS routing optimal algorithm in mobile ad hoc networks based on Hopfield neural network. *ADVANCES IN NEURAL NETWORKS – ISNN* (2005), PT 3, 3498: 343-348
7. Kocak, T., Seeber, J., Terzioglu, H.: Design and implementation of a random neural network routing engine. *IEEE Transactions on Neural Networks* (2003) Vol. 14, Issue. 5, 1128 – 1143
8. Shigang, C., Nahrstedt, K.: On finding multi-constrained paths. *ICC* (1998) Vol. 2, 874 - 879

# Sparse Kernel Ridge Regression Using Backward Deletion

Ling Wang, Liefeng Bo, and Licheng Jiao

Institute of Intelligent Information Processing  
710071, Xidian University, Xi'an, China  
`{wliip, blf0218}@163.com`

**Abstract.** Based on the feature map principle, Sparse Kernel Ridge Regression (SKRR) model is proposed. SKRR obtains the sparseness by backward deletion feature selection procedure that recursively removes the feature with the smallest leave-one-out score until the stop criterion is satisfied. Besides good generalization performance, the most compelling property of SKRR is rather sparse, and moreover, the kernel function needs not to be positive definite. Experiments on synthetic and benchmark data sets validate the feasibility and validity of SKRR.

## 1 Introduction

Regression problem is one of the fundamental problems in the field of supervised learning. It can be thought of as estimating the real valued function from a samples set of noise observation. A very successful approach for regression is Support Vector Regression (SVR) [1-2] that attempts to simultaneously minimize empirical risk and confidence interval, leading to good generalization. Due to  $\varepsilon$ -insensitive loss function, SVR obtains a sparse model (prediction for a new input only needs the subset of training samples). Though very successful, SVR also has some disadvantages [3]:

- The solution is usually not very sparse and the prediction speed for a new input is significantly slower than that of some other learning machines such as Neural Networks [4-5].
- Kernel function must satisfy Mercer's condition. It is well known that different kernel functions will induce different algorithms, achieving different performance. However, kernel functions in SVR must satisfy Mercer's positive definite condition, which limits the usable kernels.

In order to deal with the problems mentioned above, Relevance Vector Machines (RVM) [3] is proposed, which is very elegant and obtains a high sparse solution. However, RVM needs to solve linear equations, whose cost is very expensive, and therefore, it is not applicable to large scale problems.

In this paper, we propose a new learning model, Sparse Kernel Ridge Regression (SKRR) to overcome the above problems. In SKRR, samples are mapped into the feature space whose dimension is equal to the sample size, and then Ridge Regression (RR) [6] is implemented in the feature space. When the training process is completed, a backward deletion feature selection procedure is applied to obtain a sparse model.

Besides good generalization performance, the most compelling property of SKRR is its sparseness that is comparable with that of RVM. Another advantage of SKRR is that the kernel function needs not to be positive definite. Experiments on synthetic and benchmark data sets assess the feasibility and validity of SKRR.

## 2 Kernel Ridge Regression

Let  $\{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_l, y_l)\}$  be empirical samples set drawn from

$$y_i = f(\mathbf{x}_i) + \varepsilon_i, \quad i = 1, 2, \dots, l, \quad (1)$$

where  $y_i$  is corrupted by additive noise  $\varepsilon_i$ , whose distributions are usually unknown. Learning aims to infer the function  $f(\mathbf{x})$  from the finite data set  $\{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_l, y_l)\}$ . A classical method for the problem is Ridge Regression (RR), which is an extension of linear regression by adding a quadratic penalizing term:

$$\hat{\mathbf{w}} = \arg \min \left( \sum_{i=1}^l (\mathbf{w}^T \mathbf{x}_i + w_{l+1} - y_i)^2 + \lambda \|\mathbf{w}\|^2 \right), \quad (2)$$

where  $\lambda$  is a fixed positive constant, called regularization coefficient. Equation (2) can be rewritten as

$$\hat{\mathbf{w}} = \arg \min \left( \mathbf{w}^T (\mathbf{X}^T \mathbf{X} + \lambda \mathbf{I}) \mathbf{w} - 2\mathbf{w}^T \mathbf{X}^T \mathbf{y} \right), \quad (3)$$

$$\text{where } \mathbf{X} = \begin{bmatrix} \mathbf{x}_1, 1 \\ \vdots \\ \mathbf{x}_l, 1 \end{bmatrix}, \mathbf{w} = \begin{bmatrix} w_1 \\ \vdots \\ w_{l+1} \end{bmatrix}.$$

Ridge regression [6] is a well-known approach for the solution of regression problems, which has a good generalization performance as well as SVR, and the model does not need the kernel function satisfying Mercer's condition, moreover, there is an efficient leave-one-out cross-validation model selection method. In order to make RR applicable to the nonlinear problems, we generalize it by a feature map. Define a vector made up of a set of real-valued functions  $\{k(\mathbf{x}, \mathbf{x}_i) | i = 1, 2, \dots, l\}$ , as shown

$$\mathbf{z} = [k(\mathbf{x}, \mathbf{x}_1), \dots, k(\mathbf{x}, \mathbf{x}_l)]^T, \quad (4)$$

where  $k(\mathbf{x}, \mathbf{x}_i)$  is kernel function that needs not to be positive definite.

We call

$$F = \left\{ \mathbf{z} | \mathbf{z} = [k(\mathbf{x}, \mathbf{x}_1), \dots, k(\mathbf{x}, \mathbf{x}_l)]^T, \mathbf{x} \in R^n \right\} \quad (5)$$

feature space. In particular,  $\{(\mathbf{z}_1, y_1), \dots, (\mathbf{z}_l, y_l) | \mathbf{z}_i = [k(\mathbf{x}_i, \mathbf{x}_1), \dots, k(\mathbf{x}_i, \mathbf{x}_l)]^T\}$  are  $l$  empirical samples in the feature space. Substituting  $\mathbf{z}_i$  for  $\mathbf{x}_i$ , we have the Kernel Ridge Regression (KRR) [7] in the feature space  $F$

$$\hat{\mathbf{w}} = \arg \min \left( \mathbf{w}^T (\mathbf{Z}^T \mathbf{Z} + \lambda \mathbf{I}) \mathbf{w} - 2 \mathbf{w}^T \mathbf{Z}^T \mathbf{y} \right), \quad (6)$$

where  $\mathbf{Z} = \begin{bmatrix} \mathbf{z}_1, 1 \\ \vdots \\ \mathbf{z}_l, 1 \end{bmatrix}$ ,  $\mathbf{w} = \begin{bmatrix} w_1 \\ \vdots \\ w_{l+1} \end{bmatrix}$ . The corresponding decision function is

$$f(\mathbf{x}) = \sum_{i=1}^l w_i k(\mathbf{x}, \mathbf{x}_i) + w_{l+1}. \quad (7)$$

From equation (6) and (7), we can obtain

$$\hat{\mathbf{w}} = [\mathbf{Z}^T \mathbf{Z} + \lambda \mathbf{I}]^{-1} \mathbf{Z}^T \mathbf{Y}. \quad (8)$$

### 3 Sparse Kernel Ridge Regression

For a suitable  $\lambda$ , KRR has a good generalization performance. Its key disadvantage is no sparseness, which seems to prohibit its application in some fields. In this paper, we consider how to delete redundant features and simultaneously keep good generalization performance at an acceptable computational cost.

#### 3.1 Feature Selection by Backward Deletion

In order to obtain the sparseness, a backward deletion procedure is implemented after the training process. Backward deletion procedure recursively removes the feature with the smallest leave-one-out score until the stop criterion is satisfied.

Let  $\mathbf{H} = (\mathbf{Z}^T \mathbf{Z} + \lambda \mathbf{I})$  and  $\mathbf{b} = \mathbf{Z}^T \mathbf{y}$ , then equation (6) can be rewritten as

$$\hat{\mathbf{w}} = \arg \min \left( L = (\mathbf{w}^T \mathbf{H} \mathbf{w} - 2 \mathbf{w}^T \mathbf{b}) \right). \quad (9)$$

When the  $k^{th}$  feature is deleted at the  $t^{th}$  iteration, let  $\mathbf{H}^{(t,k)}$  represent the sub-matrix formed by omitting the  $k^{th}$  row and column of  $\mathbf{H}$ . Let  $\mathbf{R}^{(t,k)}$  represent the inverse of  $\mathbf{H}^{(t,k)}$ ,  $\mathbf{w}^{(t,k)}$  the weights and  $f^{(t,k)}$  the optimal value of  $L$ . According to equation (8), we have

$$f^{(t,k)} = - \sum_{i,j \in P - \{k\}} b_i (\mathbf{R}^{(t,k)})_{ij} b_j. \quad (10)$$

where  $P$  is a set of remaining features (variables) at the  $t^{th}$  iteration.

In terms of a rank-1 update [8-9],  $\mathbf{R}^{(t,k)}$  and  $\mathbf{w}^{(t,k)}$  can be formulated in equation (11) and (12) (see Appendix for details), where  $\mathbf{R}^t$  represents the inverse of  $\mathbf{H}^t$ .

$$(\mathbf{R}^{(t,k)})_{ij} = (\mathbf{R}^t)_{ij} - \frac{(\mathbf{R}^t)_{ik} (\mathbf{R}^t)_{kj}}{(\mathbf{R}^t)_{kk}}, \quad i, j \in P - \{k\}. \quad (11)$$

$$\left(\mathbf{w}^{(t,k)}\right)_i = \sum_{j \in P - \{k\}}^n \left( \left(\mathbf{R}^t\right)_{ij} - \frac{(\mathbf{R}^t)_{ik}(\mathbf{R}^t)_{kj}}{(\mathbf{R}^t)_{kk}} \right) b_j, \quad i \in P - \{k\}. \quad (12)$$

Together with  $\mathbf{w}_P^t = \mathbf{R}^t \mathbf{b}_P$ , equation (12) is simplified as

$$\left(\mathbf{w}^{(t,k)}\right)_i = \mathbf{w}_i^t - \mathbf{w}_k^t \frac{(\mathbf{R}^t)_{ik}}{(\mathbf{R}^t)_{kk}}, \quad i \in P - \{k\}. \quad (13)$$

Substituting equation (11) into (10), we obtain

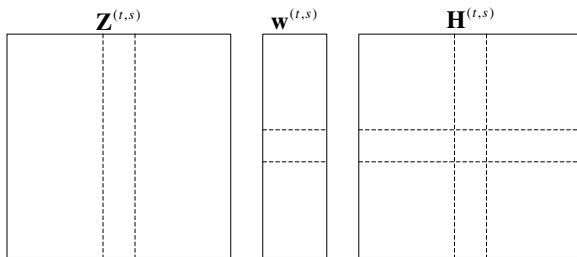
$$\begin{aligned} f^{(t,k)} &= - \sum_{i,j \in P} b_i \left(\mathbf{R}^t\right)_{ij} b_j + \sum_{i,j \in P} b_i \frac{(\mathbf{R}^t)_{ik}(\mathbf{R}^t)_{kj}}{(\mathbf{R}^t)_{kk}} b_j \\ &= f^t + \frac{\left( \sum_{j \in P} (\mathbf{R}^t)_{kj} b_j \right)^2}{(\mathbf{R}^t)_{kk}}. \end{aligned} \quad (14)$$

By the virtue of  $\sum_{j \in P} (\mathbf{R}^t)_{kj} b_j = w_k^t$ , equation (14) is translated into

$$\Delta f^{(t,k)} = f^{(t,k)} - f^t = \frac{(w_k^t)^2}{(\mathbf{R}^t)_{kk}}. \quad (15)$$

We call  $\Delta f^{(t,k)}$  leave-one-out score. At each iteration, we remove the feature with the smallest leave-one-out score. The feature to be deleted can be obtained by the following expression.

$$s = \arg \min_{k \in P - \{l+1\}} (\Delta f^{(t,k)}). \quad (16)$$



**Fig. 1.** Distribution of parameters after the  $s^{th}$  feature was deleted

Figure 1 shows the distribution of parameters after the  $s^{th}$  feature was deleted. Note that the  $(l+1)^{th}$  variable is bias that is reserved during the feature selection process.

At the  $t^{th}$  iteration, the total increase of loss function  $L$  is

$$\Delta f^t = -\sum_{i \in P} b_i w_i^t - L^{opt}. \quad (17)$$

where  $L^{opt}$  is the minimum of equation (9). We terminate the algorithm if

$$\Delta f^t \geq \varepsilon |L^{opt}|. \quad (18)$$

where  $\varepsilon$  is a small positive number.

According to the derivation above, Backward Deletion Feature Selection (BDFS) can be described as the following Algorithm 1:

---

**Algorithm 1. BDFS**


---

1. Set  $P = \{1, 2, \dots, l\}$ ,  $\mathbf{R}^1 = \mathbf{H}^{-1}$ ,  $\mathbf{w}^1 = \mathbf{R}^1 \mathbf{b}$ ;
  2. For  $k = 1$  to  $l$ , do:
    3. (a)  $s = \arg \min_{k \in P - \{l+1\}} \left( \frac{(w_k^t)^2}{(\mathbf{R}^t)_{kk}} \right)$ ;
    4. (b)  $(\mathbf{R}^{(t,s)})_{ij} = (\mathbf{R}^t)_{ij} - \frac{(\mathbf{R}^t)_{is} (\mathbf{R}^t)_{sj}}{(\mathbf{R}^t)_{ss}}$ ,  $i, j \in P - \{s\}$ ;
    5. (c)  $(\mathbf{w}^{(t,s)})_i = \mathbf{w}_i^t - \mathbf{w}_s^t \frac{(\mathbf{R}^t)_{is}}{(\mathbf{R}^t)_{ss}}$ ,  $i \in P - \{s\}$ ;
    6. (d)  $P = P - \{s\}$ ,  $\mathbf{R}^{t+1} = \mathbf{R}^{(t,s)}$ ,  $\mathbf{w}^{t+1} = \mathbf{w}^{(t,s)}$ .
    7. (e) IF  $(\mathbf{b}_P^T \mathbf{w}^{t+1} - \mathbf{b}^T \mathbf{w}^1) \geq \varepsilon \mathbf{b}^T \mathbf{w}^1$ , Stop.
  8. End For
  9. End Algorithm
- 

### 3.2 Model Selection

There exist free parameters including kernel parameter and regularization parameter in SKRR. In order to obtain good generalization, it is needed to choose the suitable parameters. Cross-validation is a model selection method often used in estimating the generalization performance of statistical classifiers. 10-fold cross-validation is often used in some kernel-based learning algorithms such as SVMs. Leave-one-out cross validation is the most extreme form of cross-validation.

Leave-one-out cross validation error is an attractive model selection criterion since it provides an almost unbiased estimator of generalization performance [2]. However, this method is rarely adopted in the kernel machines because it is computationally expensive.

Fortunately for SKRR, there is an efficient implementation for leave-one-out cross validation that only incurs a computational cost of  $O(l^3)$  [10]. Let  $E(\Gamma)$  be leave-one-out cross validation error, where  $\Gamma$  denotes free parameters.

**Lemma 1.** [11-12].  $E(\Gamma) = \frac{1}{l} \|\mathbf{B}(\Gamma)(\mathbf{I} - \mathbf{A})\mathbf{y}\|_2^2$ , where  $\mathbf{B}(\Gamma)$  is a diagonal matrix with the  $j$ th entry  $1/(1 - \alpha_{jj}(\Gamma))$ ,  $\alpha_{jj}(\Gamma)$  being the  $j$ th entry of  $\mathbf{A}(\Gamma) = \mathbf{Z}(\mathbf{Z}^T \mathbf{Z} + \lambda \mathbf{I})^{-1} \mathbf{Z}^T$ .

Let the singular value decomposition of  $\mathbf{Z}$  be

$$\mathbf{Z} = \mathbf{U} \mathbf{D} \mathbf{V}^T, \quad (19)$$

where  $\mathbf{U}, \mathbf{V}$  are orthogonal matrices,  $\mathbf{D}$  is a diagonal matrix. Substituting equation (19) into  $\mathbf{A}(\Gamma)$ , it can be simplified as

$$\mathbf{A}(\Gamma) = \mathbf{U} \mathbf{D} (\mathbf{D}^T \mathbf{D} + \lambda \mathbf{I})^{-1} \mathbf{D}^T \mathbf{U}^T, \quad (20)$$

where  $\mathbf{D}^T \mathbf{D} + \lambda \mathbf{I}$  is a diagonal matrix.

Hence for different  $\lambda$ , we only need to perform once matrix decomposition. For clarity, here we give the detail steps of leave-one-out model selection algorithm in which  $\sigma$  represents kernel parameter:

---

### Algorithm 2. LOO Model Selection

---

**For**  $\sigma_i$ ,  $i = 1, 2, \dots, q$

$$\mathbf{Z} = \mathbf{U} \mathbf{D} \mathbf{V}^T;$$

**For**  $\lambda_j$ ,  $j = 1, 2, \dots, p$

$$\alpha_{kk} = \mathbf{U}_k \mathbf{D}_k (\mathbf{D}_k^T \mathbf{D}_k + \lambda_j \mathbf{I})^{-1} \mathbf{D}_k^T \mathbf{U}_k^T;$$

$$\mathbf{B}_{kk} = 1/(1 - \alpha_{kk});$$

$$r_{(k)} = \mathbf{B}_{kk} (\mathbf{y}_k - \mathbf{A}_k \mathbf{y}_k)$$

$$E_{ij} = \frac{1}{l} \sum_{k=1}^l \{r_{(k)}\}_k^2$$

**End for**

**End for**

$$(\lambda^{opt}, \sigma^{opt}) = \arg \min_{i,j} (E_{ij}).$$


---

Thus, according to the analysis in section 3.1 and 3.2, SKRR can be described in the following Algorithm 3:

---

### Algorithm 3. SKRR

---

1. Let  $\lambda$  be noise variance, and choose the kernel parameters using LOO model selection algorithm;
  2. Train KRR using the selected parameters;
  3. Implement BDFS for KRR;
  4. Re-estimate  $\lambda$  using LOO model selection algorithm;
  5. Re-train KRR on the simplified features.
-

## 4 Simulation

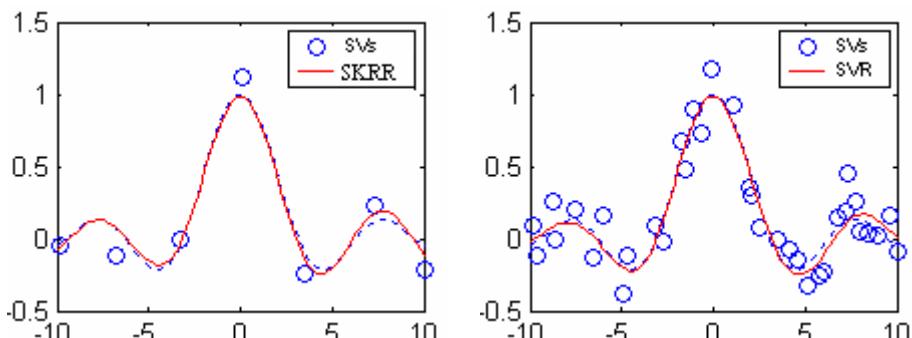
In order to evaluate the performance of the proposed algorithm, we performed SKRR on three data sets: Sinc, Boston Housing and Abalone data sets, and compared its performance with that of KRR, SVR, and RVM. For the sake of comparison, different algorithms use the same input sequence. The elements of Gram Matrix are constructed using Gaussian kernel of the form  $k(\mathbf{x}, \mathbf{y}) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{y}\|_2^2}{2\sigma^2}\right)$ . For SVR and RVM, we utilize 10-fold cross validation procedure to choose the free parameters. For all datasets, parameter  $\epsilon$  in BDFS is set 0.01. Large amounts of experiments show that it is a good selection.

### 4.1 Toy Experiment

The Sinc function  $y = \sin(x) / x + \sigma N(0, 1)$  is a popular choice to illustrate support vector machines regression. We generate training samples from Sinc function at 100 equally-spaced  $x$ -value in  $[-10, 10]$  with added Gaussian noise of standard deviation 0.1. Results are averaged over 100 random instantiations of the noise, with the error being measured over 1000 noise-free test samples in  $[-10, 10]$ . The decision functions and support vectors obtained by SKRR and SVR with  $\epsilon = 0.1$  are shown in Figure 2. Support vectors number and mean square error are summarized in Table 1. For Sinc data set, SKRR outperforms other three algorithms. SKRR and RVM obtain similar support vectors number that is significantly less than that of SVR.

**Table 1.** Generalization error obtained by the four algorithms on Sinc data set. MSE denotes mean square error and NSV denotes support vectors number

	SKRR	KRR	SVR	RVM
MSE	$0.85 \pm 0.42$	$0.88 \pm 0.41$	$1.45 \pm 0.64$	$0.98 \pm 0.47$
NSV	$7.20 \pm 0.49$	$100 \pm 0.00$	$35.41 \pm 4.71$	$6.97 \pm 0.38$



**Fig. 2.** Support vectors and decision boundary obtained by SKRR and SVR. Real red lines denote the decision boundary and dot lines denote the Sinc function; circles denote the support vectors.

## 4.2 Benchmark Comparison

Boston Housing and Abalone data sets that come from STATLOG COLLECTION [13] are popular choices to evaluate the performance of algorithms. Boston Housing data set includes 506 examples, 13 attributes of each example and Abalone data set includes 4177 examples, 7 attributes of each example. For the Boston Housing data set, we average our results over 100 random splits of the full dataset into 481 training samples and 25 testing samples. For the Abalone data set, we average our results over 10 random splits of the mother dataset into 1000 training samples and 3177 testing samples. Before experiments, we scale all training samples into [-1, 1] and then adjust testing samples using the same linear transformation. The results are summarized in Table 2 and 3.

**Table 2.** Generalization error obtained by the four algorithms on Boston Housing data set. MSE denotes mean square error and NSV denotes support vectors number.

	SKRR	KRR	SVR	RVM
MSE	$10.05 \pm 6.72$	$9.75 \pm 6.90$	$10.51 \pm 7.99$	$10.08 \pm 6.82$
NSV	$52.23 \pm 1.76$	$481.00 \pm 0.00$	$165.98 \pm 4.90$	$48.80 \pm 2.49$

From the experimental results (Table 1, 2 and 3), we observe the following

- SKRR, KRR and RVM obtain similar performance and are slightly superior to SVR.
- Both SKRR and RVM are rather sparse. Support (relevance) vectors number of SKRR and RVM is much less than that of SVR.

**Table 3.** Generalization error obtained by the four algorithms on Abalone data set. MSE denotes mean square error and NSV denotes support vectors number.

	SKRR	KRR	SVR	RVM
MSE	$4.64 \pm 0.12$	$4.61 \pm 0.10$	$4.66 \pm 0.11$	$4.59 \pm 0.12$
NSV	$18.90 \pm 5.38$	$1000 \pm 0.00$	$468.90 \pm 58.77$	$11.30 \pm 2.45$

## 5 Conclusion

Backward deletion feature selection provides a state-of-the-art tool that can delete redundant features and simultaneously keep good generalization performance at an acceptable computational cost. Based on BDFS, we propose SKRR that obtains high sparseness. Further application of BDFS includes condensing the two-stage RBF Networks. Application to linear regression is in progress.

## Acknowledgments

This work was supported by the National Natural Science Foundation of China under grant No. 60372050 and the National Grand Fundamental Research 973 Program of China under Grant No. 2001CB309403.

## References

1. Vapnik, V: The Nature of Statistical Learning Theory. New York Springer-Verlag (1995)
2. Vapnik, V: Statistical Learning Theory. New York Wiley-Interscience Publication (1998)
3. Tipping, M.E: The Relevance Vector Machine. Proc. Advances in Neural Information Processing Systems 12, Solla,S., Leen,T. and Müller K.-R. eds. Cambridge, Mass MIT Press (2001) 652-658.
4. Neal, R: Bayesian Learning for Neural Networks. New York Springer Verlag (1996)
5. Ripley R: Pattern Recognition and Neural Networks. Cambridge, U.K. Cambridge Univ. Press (1996)
6. Hoerl, A. and Kennard, R: Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics* 12 (1970) 55-67
7. Saunder C. and Gammerman A: Ridge regression learning algorithm in dual variables. In: Shavlik J. editor. Machine learning Proceedings of the 15<sup>th</sup> International Conference, Morgan Kaufmann (1998)
8. Stoer, J. and Bulirsch, R: Introduction to Numerical Analysis. Springer Verlag, New York, Second Edition (1993)
9. Bo, L.F., Wang, L., and Jiao, L.C.: Sparse Gaussian processes using backward elimination. *Lecture Notes in Computer Science* 3971 (2006) 1083-1088
10. Bo, L.F., Wang, L., and Jiao, L.C.: Feature scaling for kernel Fisher discriminant analysis using leave-one-out cross validation. *Neural Computation* 18(4) (2006) 961-978
11. Allen, D.M: The relationship between variable selection and prediction. *Technometrics* 16 (1974) 125-127
12. Gavin C.C.: Efficient leave-one-out cross-validation of kernel fisher discriminant classifiers. *Pattern Recognition* 36(11) (2003) 2585-2592
13. Michie, D, Spiegelhalter, D. J., and Taylor, C.C.: Machine Learning, Neural and Statistical Classification. Prentice Hall, Englewood Cliffs, N.J., 1994, Data available at anonymous ftp: <ftp://ftp.ncc.up.pt/pub/statlog/>.

## Appendix

The matrix inversion formula for a symmetric matrix with block sub-matrices is given in the following Lemma 2.

**Lemma 2 [8]:** Given invertible matrices  $\mathbf{A}$  and  $\mathbf{C}$ , and matrix  $\mathbf{B}$ , the following equality holds:

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^T & \mathbf{C} \end{bmatrix}^{-1} = \begin{bmatrix} \mathbf{A}^{-1} + \mathbf{A}^{-1}\mathbf{B}\mathbf{E}^{-1}\mathbf{B}^T\mathbf{A}^{-1} & -\mathbf{E}^{-1}\mathbf{B}\mathbf{C}^{-1} \\ -\mathbf{C}^{-1}\mathbf{B}^T\mathbf{E}^{-1} & \mathbf{C}^{-1} + \mathbf{C}^{-1}\mathbf{B}^T\mathbf{D}^{-1}\mathbf{B}\mathbf{C}^{-1} \end{bmatrix}, \quad (\text{A.1})$$

where  $\mathbf{D} = \mathbf{A} - \mathbf{B}\mathbf{C}^{-1}\mathbf{B}^T$ .  
 $\mathbf{E} = \mathbf{C} - \mathbf{B}^T\mathbf{A}^{-1}\mathbf{B}$ .

Equation (11) can be derived in terms of Lemma 2. We assume here that  $\begin{bmatrix} \mathbf{B} \\ \mathbf{C} \end{bmatrix}$  is the  $k^{th}$  feature to be deleted. Thus in our problem,  $\mathbf{R}^{(t,k)}$  corresponds to  $\mathbf{A}^{-1}$ , and

$(\mathbf{R}^t)_{kk}$  corresponds to  $\mathbf{E}^{-1}$ , and  $(\mathbf{R}^t)_{ik}$  corresponds to  $-\mathbf{A}^{-1}\mathbf{B}\mathbf{E}^{-1}$ ,  $i \in P - \{k\}$ . Observing the top left block of the last part in equation (A.1), we have

$$\mathbf{A}^{-1} + \frac{\mathbf{A}^{-1}\mathbf{B}\mathbf{E}^{-1}\mathbf{E}^{-1}\mathbf{B}^T\mathbf{A}^{-1}}{\mathbf{E}^{-1}}, \quad (\text{A.2})$$

which corresponds to

$$(\mathbf{R}^{(t,k)})_{ij} + \frac{(\mathbf{R}^t)_{ik}(\mathbf{R}^t)_{kj}}{(\mathbf{R}^t)_{kk}}, \quad i, j \in P - \{k\}. \quad (\text{A.3})$$

Hence, we have

$$(\mathbf{R}^{(t,k)})_{ij} = (\mathbf{R}^t)_{ij} - \frac{(\mathbf{R}^t)_{ik}(\mathbf{R}^t)_{kj}}{(\mathbf{R}^t)_{kk}}, \quad i, j \in P - \{k\}. \quad (\text{A.4})$$

# Using Locally Weighted Learning to Improve SMOreg for Regression<sup>\*</sup>

Chaoqun Li<sup>1</sup> and Liangxiao Jiang<sup>2</sup>

<sup>1</sup> Faculty of Mathematics and Physics, China University of Geosciences  
Wuhan, Hubei, P.R.China, 430074  
[chqli@cug.edu.cn](mailto:chqli@cug.edu.cn)

<sup>2</sup> Faculty of Computer Science, China University of Geosciences  
Wuhan, Hubei, P.R.China, 430074  
[ljiang@cug.edu.cn](mailto:ljiang@cug.edu.cn)

**Abstract.** Shevade et al.[1] are successful in extending some improved ideas to Smola and Scholkopf's SMO algorithm[2] for solving regression problems, simply named SMOreg. In this paper, we use SMOreg in exactly the same way as linear regression(LR) is used in locally weighted linear regression[5](LWLR): a local SMOreg is fit to a subset of the training instances that is in the neighborhood of the test instance whose target function value is to be predicted. The training instances in this neighborhood are weighted, with less weight being assigned to instances that are further from the test instance. A regression prediction is then obtained from SMOreg taking the attribute values of the test instance as input. We called our improved algorithm locally weighted SMOreg, simply LWSMOreg. We conduct extensive empirical comparison for the related algorithms in two groups in terms of relative mean absolute error, using the whole 36 regression data sets obtained from various sources and recommended by Weka[3]. In the first group, we compare SMOreg[1] with NB[4](naive Bayes), KNNDW[5](k-nearest-neighbor with distance weighting), and LR. In the second group, we compare LWSMOreg with SMOreg, LR, and LWLR. Our experimental results show that SMOreg performs well in regression and LWSMOreg significantly outperforms all the other algorithms used to compare.

**Keywords:** SMOreg, locally weighted SMOreg, locally weighted learning, regression, SVM.

## 1 Introduction

Regression is one of the most important tasks in machine learning and data mining. Different from classification, regression is used to predict numeric target values. In regression, a model is built from a set of training instances with numeric target values and is typically measured by its relative mean absolute error

\* This work was supported by Excellent Youth Foundation of China University of Geosciences(No.CUGQNL0505).

on the test instances. When our learning goal is to build a model for regression, many state-of-the-art algorithms such as naive Bayes, k-nearest-neighbor, linear regression, and support vector machine are all alternative.

In this paper, we focus our interest on support vector machine (simply SVM). In other words, we address the regression problem using support vector machine. In 1998, Smola and Scholkopf[2] proposed an iterative algorithm called Sequential Minimal Optimization (simply SMO), for solving the regression problem using SVM. This algorithm is an extension of the SMO algorithm proposed by Platt[6] for SVM classifier design. The remarkable feature of the SMO algorithms is that they are fast as well as very easy to implement.

In 1999, Keerthi and Shevade[7] improve Platt's SMO algorithm for SVM classification. After this, Shevade et al.[1] are successful in extending Keerthi and Shevade's improved ideas to Smola and Scholkopf's SMO algorithm[2] for solving regression problem, simply named SMOreg. SMOreg overcomes an important source of confusion and inefficiency caused by the way SMO maintains a single threshold value. Getting clues from optimality criteria associated with the dual problem, SMOreg use two threshold parameters and devise two modified versions of SMO for regression. Their experiments show that: 1) SMOreg enhances the value of SMO for regression even further; 2) SMOreg is much more efficient than the original SMO; 3) SMOreg performs significantly better than the original SMO. Our experimental results provide another evidence to support their conclusions.

Linear regression is an excellent and simply algorithm for regression which has been widely used in statistical application for decades. Its improved algorithm called locally weighted linear regression makes a great success by applying locally weighted learning[8] to linear regression. Motivated by this improved idea, we present to use SMOreg in exactly the same way as linear regression is used in locally weighted linear regression: a local SMOreg is fit to a subset of the training instances that is in the neighborhood of the test instance whose target function value is to be predicted. The training instances in this neighborhood are weighted, with less weight being assigned to instances that are further from the test instance. A regression prediction is then obtained from SMOreg taking the attribute values of the test instance as input. We called our improved algorithm locally weighted SMOreg, simply LWSMOreg. Our experimental results show that LWSMOreg significantly outperforms the original SMOreg.

In fact, except for locally weighted linear regression, locally weighted naive Bayes[9] is another successful example applying locally weighted learning to improve some other learning algorithms.

The rest of the paper is organized as follows. In Section 2, we look back related work on regression. In Section 3, we present our improved regression algorithm called Locally Weighted SMOreg. In Section 4, we describe the experimental setup and results in detail. In Section 5, we make a conclusion and outline our main directions for future research.

## 2 Related Work

Just as discussed in Section 1, many state-of-the-art algorithms such as naive Bayes, k-nearest-neighbor, linear regression, and support vector machine can be used for regression. Now let us simply look back these regression algorithms.

### 2.1 Naive Bayes

Naive Bayes relies on an assumption (attribute conditional independence assumption) that is rarely valid in practical learning problems: that the attributes used for deriving a prediction are independent of each other, given the predicted value. However, recent work in supervised learning has shown that naive Bayes with unrealistic attribute conditional independence assumption surprisingly has good classification performance. This suggests that, for classification problems where the predicted value is categorical, the attribute conditional independence assumption is less restrictive than might be expected. Therefore, it is interesting to see how it performs in domains where the predicted value is numeric. Motivated by this interest, Frank et al.[4] extend naive Bayes to numeric prediction tasks (namely regression problems). Their experimental results indicate that the attribute conditional independence assumption that naive Bayes makes is indeed more restrictive for regression than for classification.

### 2.2 K-Nearest-Neighbor

K-Nearest-Neighbor assumes all instances correspond to points in the  $n$  dimensional space  $R^n$ . The nearest neighbors of an instance are defined in terms of the standard Euclidean distance. More precisely, let an arbitrary instance  $x$  be described by the feature vector  $\langle a_1(x), a_2(x), \dots, a_n(x) \rangle$ , where  $a_r(x)$  denotes the value of the  $r$ th attribute  $A_r$  of the instance  $x$ . Then the distance between two instances  $x_i$  and  $x_j$  is defined to be  $d(x_i, x_j)$ , where

$$d(x_i, x_j) = \sqrt{\sum_{r=1}^n (a_r(x_i) - a_r(x_j))^2} \quad (1)$$

The k-nearest-neighbor algorithm approximating a numeric-valued target function  $f : R^n \rightarrow R$  returns the value of  $\hat{f}(x_q)$  as its estimate of  $f(x_q)$ , which is just the mean value of  $f$  among the  $k$  training instances nearest to the test instance  $x_q$ . Namely

$$\hat{f}(x_q) = \frac{\sum_{i=1}^k f(x_i)}{k} \quad (2)$$

One obvious refinement to the k-nearest-neighbor algorithm is to weight the contribution of each of the  $k$  neighbors according to their distance to the test instance  $x_q$ , giving greater weight to closer neighbors. Simply, we can weight the contribution of each of the  $k$  neighbors according to the inverse of its distance

from the test instance  $x_q$ . Thus, we can replace Equation 2 by Equation 3. We call this improved algorithm k-nearest-neighbor with distance weighting, simply KNNDW.

$$\hat{f}(x_q) = \frac{\sum_{i=1}^k w_i f(x_i)}{\sum_{i=1}^k w_i} \quad (3)$$

where

$$w_i = \frac{1.0}{1.0 + d(x_q, x_i)}. \quad (4)$$

### 2.3 Linear Regression

Linear Regression is a natural algorithm to consider when the value of the target function is numeric. The basic idea of linear regression is to express the target function  $\hat{f}(x)$  as a linear combination of the attributes with predetermined weights. Namely linear regression approximate the target function  $\hat{f}(x)$  using a linear function of the form:

$$\hat{f}(x) = w_0 + w_1 a_1(x) + \dots + w_n a_n(x) \quad (5)$$

where  $a_r(x)(r = 1, 2, \dots, n)$  is the value of the  $r$ th attribute  $A_r$  of the test instance  $x$ , and  $w_r(r = 0, 1, \dots, n)$  is the related weight.

In order to fit such linear function to a given set  $D$  of training instances, the training algorithm called gradient descent is used to find the coefficients  $w_0, w_1, \dots, w_n$  to minimize the error  $E(x)$ . In other words, we derived methods to choose weights that minimize the squared error summed over the set  $D$  of training instances, namely

$$E(x) = \frac{1}{2} \sum_{x \in D} (f(x) - \hat{f}(x))^2 \quad (6)$$

which led us to the gradient descent training rule as follow, where  $\eta$  is a constant learning rate.

$$\Delta w_r = \eta \sum_{x \in D} (f(x) - \hat{f}(x)) a_r(x). \quad (7)$$

### 2.4 Support Vector Machine

Support Vector Machine (SVM) is an excellent model for solving the regression problems. Smola and Scholkopf[2] proposed an iterative algorithm called Sequential Minimal Optimization (simply named SMO), for solving the regression problems using SVM. This algorithm is an extension of the SMO algorithm proposed by Platt[6] for SVM classifier design. In order to address the classification problems, Keerthi and Shevade[7] present an improve version of Platt's SMO algorithm. After this, Shevade et al.[1] are successful in extending Keerthi and Shevade's improved ideas to Smola and Scholkopf's SMO algorithm[2] for solving regression problem, simply named SMOreg. SMOreg overcomes an important source of confusion and inefficiency caused by the way SMO maintains a

single threshold value. Getting clues from optimality criteria associated with the dual problem, SMOreg use two threshold parameters and devise two modified versions of SMO for regression.

### 3 Locally Weighted SMOreg

Locally weighted linear regression is a locally weighted version of linear regression. It uses a local linear regression to fit to a subset of the training instances that is in the neighborhood of the test instance whose target function value is to be predicted. The training instances in this neighborhood are weighted according to the inverse of its distance from the test instance, with less weight being assigned to instances that are further from the test instance. A regression prediction is then obtained from linear regression taking the attribute values of the test instance as input. Thus, the error and training rule used in linear regression should respectively be replaced by Equation 8 and Equation 9.

$$E(x_q) = \frac{1}{2} \sum_{i=1}^k K(d(x_q, x_i))(f(x_i) - \hat{f}(x_i))^2 \quad (8)$$

$$\Delta w_r = \eta \sum_{i=1}^k K(d(x_q, x_i))(f(x_i) - \hat{f}(x_i))a_r(x_i) \quad (9)$$

where  $k$  is the number of neighbors,  $K(d(x_q, x_i))$  is the weighting kernel function calculated by Equation 10.

$$K(d(x_q, x_i)) = \frac{1.0}{1.0 + d(x_q, x_i)} \quad (10)$$

For solving the regression problems using support vector machine (SVM), Smola and Scholkopf[2] proposed an iterative algorithm called Sequential Minimal Optimization (simply SMO). This algorithm is an extension of the SMO algorithm proposed by Platt[6] for SVM classifier design. After this, Keerthi and Shevade[7] present an improve version of Platt's SMO algorithm for SVM classifier design. Motivated by this work and aiming at scaling up the value, efficiency, and regression performance of Smola and Scholkopf's SMO algorithm[2], Shevade et al.[1] extend Keerthi and Shevade's improved ideas to Smola and Scholkopf's SMO algorithm[2] and make a great success. The resulting regression algorithm is simply named SMOreg.

Just as discussed before, locally weighted learning has already successfully applied to improve the regression performance of linear regression and to improve the classification performance of naive Bayes[9]. This fact raises the question of whether such locally weighted learning can be used to improve the regression performance of SMOreg.

Responding to this question, we present a hybrid algorithm called locally weighted SMOreg by combining locally weighted learning with SMOreg, simply

LWSMOreg. LWSMOreg use SMOreg in exactly the same way as linear regression is used in locally weighted linear regression: a local SMOreg is fit to a subset of the training instances that is in the neighborhood of the test instance whose target function value is to be predicted. The training instances in this neighborhood are weighted according to the inverse of its distance from the test instance, with less weight being assigned to instances that are further from the test instance. A regression prediction is then obtained from SMOreg taking the attribute values of the test instance as input.

The subset of the training instances used to train each locally weighted SMOreg are determined by a  $k$ -nearest-neighbor algorithm. A user-specified parameter  $k$  controls how many instances are used. So, like locally weighted linear regression and locally weighted naive Bayes, our locally weighted SMOreg also is a  $k$ -relates algorithm. Fortunately, we get almost same experimental results with locally weighted naive Bayes: LWSMOreg is not particularly sensitive to the choice of value of  $k$  as long as it is not too small. This makes it a very attractive alternative to the  $k$ -nearest-neighbor algorithm, which requires fine-tuning of  $k$  to achieve good results.

Although our experimental results show that LWSMOreg significantly outperforms the original SMOreg. Our improvements turn a eager learning algorithm into a lazy learning algorithm. Like all the other lazy learning algorithms, LWSMOreg simply stores the training instances and defers the effort involved in learning until prediction time. When called upon to predict a test instance, LWSMOreg constructs an SMOreg using a weighted set of training instances in the neighborhood of the test instance. In a word, an obvious disadvantage with LWSMOreg is that it has relatively higher time complexity. So, enhancing LWSMOreg's efficiency is one main direction for our future research.

## 4 Experimental Methodology and Results

We run our experiments on the whole 36 regression data sets obtained from various sources and recommended by Weka[3]. We download these data sets in format of *arff* from main web of Weka[3].

We conduct extensive empirical comparison for the related algorithms in two groups in terms of relative mean absolute error. In the first group, we compare SMOreg[1] with NB[4], KNNDW[5], and LR. In the second group, we compare LWSMOreg with SMOreg, LR, and LWLR[5]. We use the implementation of KNNDW(IBk), LR(LinearRegression), SMOreg, and locally weighted learning(LWL) in Weka and the implementation of NB(naive Bayes for regression) provided by Frank, E.. We set the weighting kernel function to the inverse of their distance in KNNDW, LWLR, and LWSMOreg, the number of neighbors to 5 in KNNDW, 50 in LWLR and LWSMOreg.

The relative mean absolute error of each algorithm was obtained via the ten-fold cross validation for all data sets. Runs with the various algorithms were carried out on the same training sets and evaluated on the same test sets. In particular, the cross-validation folds are the same for all the experiments on each

**Table 1.** Experimental results: relative mean absolute error and standard deviation

Datasets	NB	KNNDW	LR	SMOreg
auto93	59.1±21.78	54.96±16.28	60.54±17.28	61.85±12.7
autoHorse	27.69±12.57	34.14±6.95	25.14±5.69	24.7±5.85
autoMpg	39.15±5.55	34.9±6.33	34.96±8.63	35.02±8.1
autoPrice	41.28±9.69	35.61±11.76	43.4±11.37	39.24±7.9
basketball	105.68±30.95	90.28±22.98	80.46±16.85	86.17±18.81
bodyfat	21.64±6.19	34.96±7.1	7.48±2.66	3.81±3.7
bolts	38.59±32.8	46.4±36.15	35.76±14.32	34.28±33.39
breastTumor	133.64±16.95	113.02±15.77	98.05±11.72	99.18±9.61
cholesterol	115.88±14.95	111.07±14.3	100.11±7.04	101.89±9.41
cleveland	60.23±15.02	61.89±10.93	64±6.95	62.91±6.6
cloud	49.39±8.08	67.48±13.74	34.34±10.46	34.68±11.32
cpu	27.44±9.58	29.73±13.11	43.12±11.38	21.94±6.46
detroit	109.06±109.68	118.45±136.75	250.92±479.89	77.11±99.77
echoMonths	79.07±21.5	71.67±19.01	64.9±14.92	68.92±14.22
elusage	51.69±17.06	49.97±19.67	43.25±19.67	54.31±19.68
fishcatch	23.8±10.46	22.97±6.18	24.05±5.61	21.99±5.25
fruitfly	147.44±52.77	133.41±30.41	100.24±0.76	103.01±9.83
gascons	30.69±18.75	18.44±9.84	84.33±26.12	24.77±18.41
housing	57.84±12.67	41.18±5.36	51.86±8.85	49.46±10.65
hungarian	31.83±10.56	44.15±7.75	54.55±8.34	42.4±11.39
longley	39.31±42.36	52.47±64.16	24.09±16.95	13.22±13.3
lowbwt	70.83±12.42	64.6±9.72	63.97±8.81	65.85±12.55
mbagrade	96.4±30.41	98.89±27.52	84.48±21.31	85.33±25.34
meta	69.12±27.15	89.91±14.98	121.25±24.54	52.79±21.75
pbc	86±8.07	87.6±7.41	78.49±5.78	81.18±5.36
pharynx	86.78±20.36	75.89±10.63	70.71±12.07	70.49±11.7
pollution	91±34.55	71.21±14.3	66.18±25.48	61.3±20.01
pwLinear	56.56±16.47	51.33±9.38	52±15.48	51.27±15.66
quake	121.89±6.71	105.41±2.51	99.58±0.43	94.23±3.61
schlvote	84.83±29.21	115.01±82.96	119.98±43.76	86.58±28.39
sensory	127.83±16.84	89.92±4.77	93.31±5.29	96.81±7.14
servo	67.24±11.64	46.62±11.15	54.78±16.71	46.43±25.17
sleep	95.56±57.23	79.9±36.81	82.61±30.12	77.16±28.29
strike	97.2±40.4	68.51±14.09	74.57±9.02	60.6±8.78
veteran	72.44±21.22	106.35±27.85	91.91±24.8	75.24±18.04
vineyard	61.2±32.23	58.97±24.24	66.83±28.91	67.26±24.34
<b>Mean</b>	71.54±23.47	68.81±21.47	70.73±27.17	59.26±16.46

**Table 2.** Experimental results: two-tailed t-test with the 95% confidence level

	NB	KNNDW	LR	SMOreg
NB	-	9/23/4	13/19/4	13/22/1
KNNDW		-	10/19/7	11/23/2
LR			-	12/19/5
SMOreg				-

**Table 3.** Experimental results: relative mean absolute error and standard deviation

Datasets	LR	SMOreg	LWLR	LWSM0reg
auto93	60.54±17.28	61.85±12.7	61.83±13.84	58.46±12.24
autoHorse	25.14±5.69	24.7±5.85	23.28±5.83	21.08±5.9
autoMpg	34.96±8.63	35.02±8.1	33.21±8.13	33.45±7.59
autoPrice	43.4±11.37	39.24±7.9	41.14±11.01	36.78±9.37
baskball	80.46±16.85	86.17±18.81	80.78±17.55	85.63±18.61
bodyfat	7.48±2.66	3.81±3.7	6.83±2.7	3.62±3.64
bolts	35.76±14.32	34.28±33.39	33.58±14.38	31.45±30.8
breastTumor	98.05±11.72	99.18±9.61	99.09±12.3	100.7±10.1
cholesterol	100.11±7.04	101.89±9.41	100.44±7.72	100.62±9.08
cleveland	64±6.95	62.91±6.6	63.43±6.48	62.08±7.2
cloud	34.34±10.46	34.68±11.32	34.34±10.44	34.34±11.14
cpu	43.12±11.38	21.94±6.46	36.79±10.7	20.15±6.61
detroit	250.92±479.89	77.11±99.77	231.43±465.59	64.84±67.02
echoMonths	64.9±14.92	68.92±14.22	64.69±14.13	69.21±14.41
elusage	43.25±19.67	54.31±19.68	42.57±22.4	54.86±15.84
fishcatch	24.05±5.61	21.99±5.25	22.88±5.4	21.37±5.17
fruitfly	100.24±0.76	103.01±9.83	100.66±0.61	103.34±9.95
gascons	84.33±26.12	24.77±18.41	75.35±23.66	17.73±11.58
housing	51.86±8.85	49.46±10.65	46.33±7.76	43.64±8.86
hungarian	54.55±8.34	42.4±11.39	54.17±8.26	42.99±11.87
longley	24.09±16.95	13.22±13.3	22.76±16.13	12.63±11.81
lowbwt	63.97±8.81	65.85±12.55	63.65±8.64	65.37±12.91
mbagrade	84.48±21.31	85.33±25.34	83.82±21.39	83.59±25.92
meta	121.25±24.54	52.79±21.75	117.76±23.14	52.69±21.69
pbc	78.49±5.78	81.18±5.36	78.73±6.12	82.31±4.68
pharynx	70.71±12.07	70.49±11.7	72.03±11.73	70.57±12.17
pollution	66.18±25.48	61.3±20.01	65.08±23.85	60.97±18.24
pwLinear	52±15.48	51.27±15.66	49.98±14.5	48.15±15.11
quake	99.58±0.43	94.23±3.61	99.31±0.49	94.27±3.9
schlvote	119.98±43.76	86.58±28.39	121.69±47.63	83.98±31.84
sensory	93.31±5.29	96.81±7.14	92.72±4.98	94.95±6.84
servo	54.78±16.71	46.43±25.17	52.49±15.89	43.26±24.52
sleep	82.61±30.12	77.16±28.29	82.72±30.03	77.91±26.9
strike	74.57±9.02	60.6±8.78	71.98±8.75	58.65±9.21
veteran	91.91±24.8	75.24±18.04	91.39±25.1	75.73±18.37
vineyard	66.83±28.91	67.26±24.34	65.87±29.39	68.73±23.78
Mean	70.73±27.17	59.26±16.46	69.02±26.57	57.78±15.14

**Table 4.** Experimental results: two-tailed t-test with the 95% confidence level

	LR	SMOreg	LWLR	LWSM0reg
LR	-	12/19/5	15/20/1	15/17/4
SMOreg		-	5/23/8	14/22/0
LWLR			-	12/20/4
LWSM0reg				-

data set. Finally, we compare each pair of algorithms via two-tailed t-test with significantly different probability of 95%, because we speak of two results for a data set as being “significantly different” only if the difference is statistically significant at the 0.05 level according to the corrected two-tailed t-test[10].

Table 1 and Table 3 show the detailed experimental results on relative mean absolute error and standard deviation of each algorithm on each data set, and their average values are summarized at the bottom of the tables. Table 2 and Table 4 show the compared results of two-tailed t-test with a 95% confidence level between each pair of algorithms, each entry  $w/t/l$  in Tables means that the algorithm at the corresponding column wins in  $w$  data sets, ties in  $t$  data sets, and loses in  $l$  data sets, compared to the algorithm at the corresponding row.

From Table 1-4, we can see that SMOreg performs well in regression and LWSMOreg significantly outperforms all the other algorithms used to compare. Now, we summarize the highlights as follows:

1. Measured by relative mean absolute error, SMOreg significantly outperforms NB, KNNDW, and LR. the  $w/t/l$  value between SMOreg and them respectively is 13/22/1, 11/23/2, and 12/19/5. SMOreg’s average value on 36 data sets is 59.26%, which also is the lowest one in all algorithms.
2. Measured by relative mean absolute error, LWSMOreg significantly outperforms LR, SMOreg, and LWLR. the  $w/t/l$  value between LWSMOreg and them respectively is 15/17/4, 14/22/0, and 12/20/4. SMOreg’s average value on 36 data sets is 57.78%, which also is the lowest one in all algorithms.
3. Besides having better regression performance, LWSMOreg has better robustness and stability than all the other algorithms used to compare. LWSMOreg’s average standard deviation on relative mean absolute error is 15.14, which are the lowest in all algorithms.

## 5 Conclusions

Shevade et al.[1] extend some improved ideas to Smola and Scholkopf’s SMO algorithm[2] for solving regression problem. The resulting regression algorithm is simply named SMOreg. Linear regression(LR) is another typical regression algorithm. Its improved algorithm called locally weighted linear regression(LWLR) is an elegant regression algorithm, which combines locally weighted learning with linear regression. Motivated by LWLR’s success, we present to apply locally weighted learning to Shevade’s SMOreg. We call our improved algorithm locally weighted SMOreg, simply LWSMOreg. Our experimental results show that SMOreg performs well in regression and LWSMOreg significantly outperforms all the other algorithms used to compare. Unfortunately, LWSMOreg is a lazy learning, which has relatively higher time complexity. So, enhancing LWSMOreg’s efficiency is one main direction for our future research.

## Acknowledgements

Many thanks to Frank, E. for kindly providing us with the naive Bayes implementation used to regression.

## References

1. S.K, Shevade, S.S. Keerthi, C. Bhattacharyya and K.R.K. Murthy, Improvements to SMO Algorithm for SVM Regression, Technical Report CD-99-16, Control Division, Dept. of Mechanical and Production Engineering, National University of Singapore, Singapore, 1999
2. Smola A.J. and Scholkopf B., A tutorial on support vector regression, NeuroCOLT2 Technical Report Series NC2-TR-1998-030, ESPRIT working group on Neural and Computational Learning Theory Neuro- COLT 2, 1998.
3. Witten, I. H., Frank, E.: data mining-Practical Machine Learning Tools and Techniques with Java Implementation. Morgan Kaufmann (2000). <http://prdownloads.sourceforge.net/weka/datasets-numeric.jar>
4. Frank, E., Trigg, L., Holmes, G., Witten, I. H. (2000). Naive bayes for regression. Machine Learning, 41, 5-15.
5. Mitchell, T.M.: Instance-Based Learning. Chapter 8 in Machine Learning. McGraw-Hill, 1997.
6. J. Platt. Fast training of support vector machines using sequential minimal optimization. In B. Scholkopf, C. J. C. Burges, and A. J. Smola, editors, Advances in Kernel Methods.Support Vector Learning, pages 185.208, Cambridge, MA, 1999. MIT Press.
7. S. S. Keerthi, S. K. Shevade, C. Bhattacharyya, and K. R. K. Murthy. Improvements to Platt's SMO algorithm for SVM classifier design. Technical Report CD-99-14, Dept. of Mechanical and Production Engineering, Natl. Univ. Singapore, Singapore, 1999.
8. Atkeson, C. G., Moore, A. W., and Schaal, S., Locally Weighted Learning, Artificial Intelligence Review, Vol. 11, No. 1-5, 1997, pp.11-73.
9. Frank, E., Hall, M., Pfahringer, B.: Locally Weighted Naive Bayes. Proceedings of the Conference on Uncertainty in Artificial Intelligence (2003). Morgan Kaufmann(2003), 249-256.
10. Nadeau, C., Bengio, Y.: Inference for the generalization error. Advances in Neural Information Processing Systems 12 (1999) 307-313. MIT Press.

# Palmprint Recognition Using Wavelet and Support Vector Machines\*

Xinhong Zhou, Yuhua Peng, and Ming Yang

School of Information Science and Engineering, Shandong University, Jinan, Shandong,  
People's Republic of China  
zhouxinhong@mail.sdu.edu.cn, pyuhua@sdu.edu.cn,  
mingyang@mail.sdu.edu.cn

**Abstract.** In recent years, palmprint identification has been developed for security purpose. In this paper, a novel scheme of palmprint identification is proposed. We apply 2-dimensional 2\_band (Discrete Wavelet Transform) and 3\_band wavelet decomposition to get the low subband images, and then use them as identification feature vectors. We choose support vector machines as classifier. The experimental results demonstrate that it is a simple and accurate identification strategy and the correct recognition rate is high up to 100%.

## 1 Introduction

### 1.1 Palm Print Identification

Recently, automatic personal authentication has been developed and used for security system; one of the popular and powerful means of it is biometrics, because of that an increasing worldwide effort has been devoted to biometrics technologies. Similar to fingerprint and face recognition and iris verification, the palmprint is also one of the most reliable means of biometrics technologies. But compared with other technologies, palmprints have a number of unique advantages. Firstly, they are rich in features such as principal lines, wrinkles, and textures and those features are stable and can provide distinctive information sufficient for separating an individual from a large population. Secondly, just low resolution palmprints are enough for identification. Thirdly, they are relatively user-friendly and convenient .All those good features (unique and stable, low cost, non-intrusion) make it attractive and useful for personal authentication.

Commonly used schemes for palmprint identification are line or point [1] feature extraction for model matching. [2] developed a BDPPCA from PCA which can reduce the dimension of the original image matrix in both column and row directions. In [3], authors integrated the multi-resolution (wavelet) analysis method and ICA to represent the palmprint features. In the paper [4], they proposed to use the wavelet energy feature (WEF) of the scalable detailed sub-images. However, our work

---

\* This project is sponsored by SRF for ROCS, SEM (2004.176.4) and NSF SD Province (Z2004G01) of China.

concentrate only on the low resolution approximation sub-image, the experiments show that even just the hardly blurred (to 8×8,even 4×4)resolution approximation sub-images can reflect sufficient information of nearly all the features of palmprints (Geometrical feature, Principal lines, Wrinkles, Delta points, Minutiae),so are quite discriminable approximation coefficients (without all the scalable detailed coefficients) resulted from Wavelet Transform should be paid more attention on.

## 1.2 Wavelet Transform

The basic features in a palmprint, including principal lines, wrinkles and ridges, have different resolutions, so for classification we extract the Wavelet domain features: The two-dimensional (2D) 2-band wavelet decomposition (DWT) on J scales of a discrete image  $a_0(m,n)$  represents the image in terms of  $3J + 1$  sub-images and the 3-band wavelet decomposition (we call it TWT) in terms of  $8J + 1$  sub-images:

$$[a_J, \{d_j^1, d_j^2, d_j^3\}_{j=1,2,\dots,J}] \text{ or } [a_J, \{d_j^1, d_j^2, \dots, d_j^8\}_{j=1,2,\dots,J}]$$

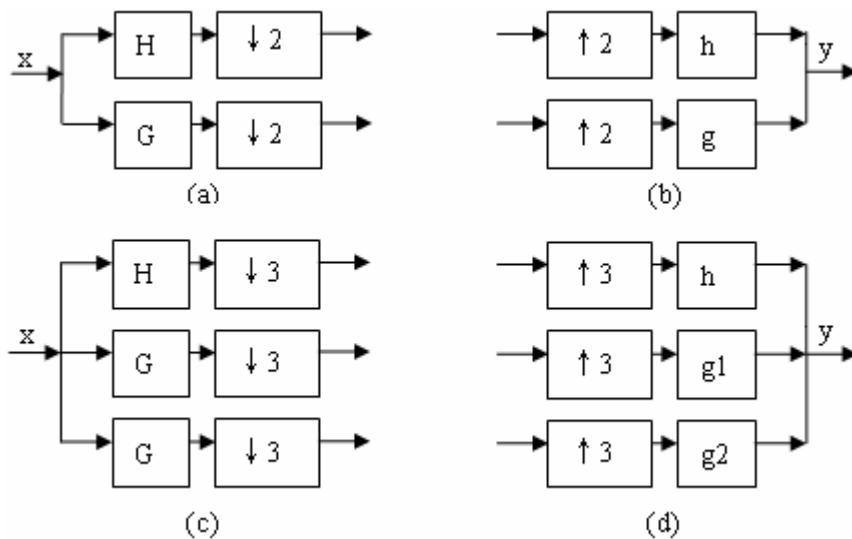
Where  $a_J$  is a low resolution approximation of the original image, and  $d_j^l$  are the wavelet sub-images containing the image details at different scales ( $2^j$  or  $3^j$ ) and orientations. Here in our text, we only use the low resolution approximation coefficient  $a_J$ .

## 1.3 Support Vector Machines (SVM)

Recently, support vector machine is hot and widely used in many artificial intelligence areas such as isolated handwritten digit recognition, object recognition, speaker identification, text categorization, face detection and identification, etc. However, it is seldom used for palmprint recognition. SVM tend to minimize an upper bound on the expected risk (structural risk minimization), while other learning techniques such as neural networks usually tend to minimize the error on the training set (empirical risk minimization), so it has good generalization performances on many pattern recognition problems. Besides this, we choose SVC for its several other attractive features, such as simplicity of implementation, few free parameters required to be tuned, and the ability to deal with few samples and high-dimensional input data on many classifying problems.

## 2 Two-Dimensional 2\_Band and 3\_Band Wavelet Decomposition for Feature Extraction

We use 2-band and 3-band wavelet [5] to analyze the palmprint images, the structure figures of Mallat's decomposition and reconstruction algorithms are shown in Fig.1, where H is the low-pass filter,G1 and G2 are band-pass filters, h, g1 and g2 are



**Fig. 1.** The structure of 2-band and 3-band wavelet decomposition and reconstruction

(a) 2-band decomposition figure(x is the original signal)

(b) 2-band reconstruction figure (y is the reconstructing signal)

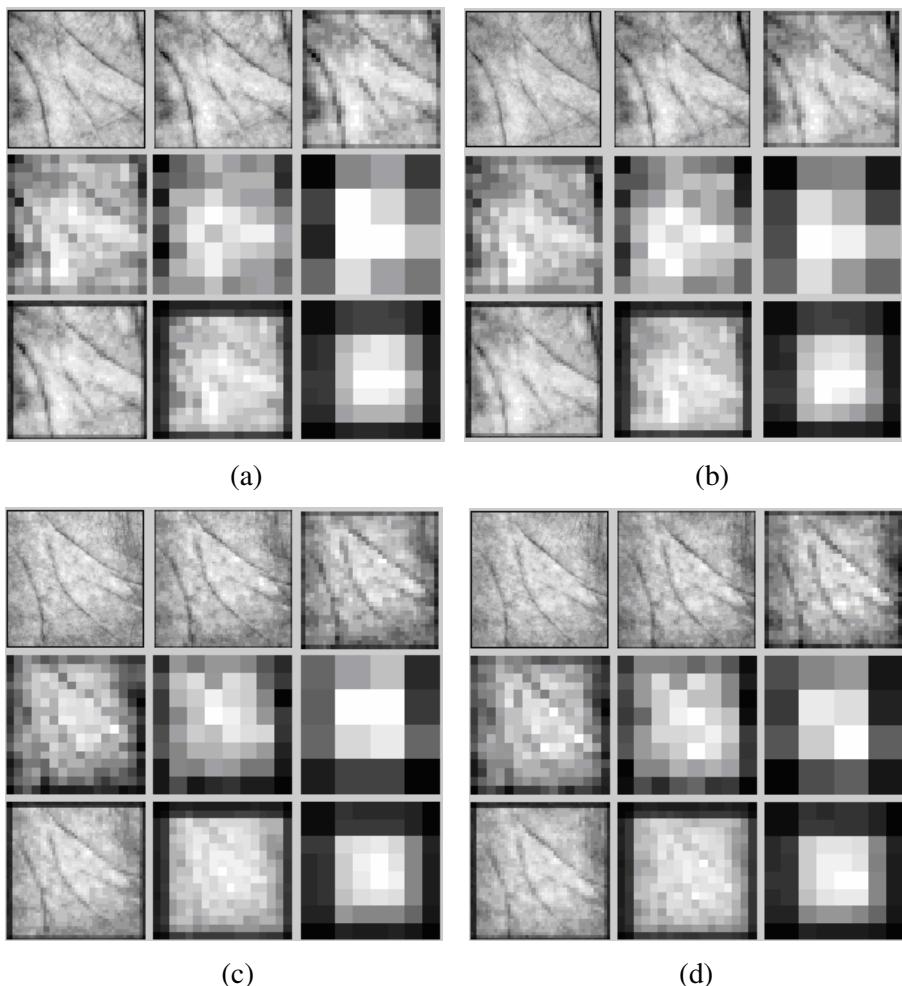
(c) 3-band decomposition figure(x is the original signal)

(d) 3-band reconstruction figure (y is the reconstructing signal)

reconstruction filters corresponding to H, and G1and G2 respectively. The 3-band orthogonal wavelet here we used is HARR filter banks, whose coefficients are given in Tab.1. Some example of low resolution subband images are shown in Fig.2. After decomposition, we extend the low resolution approximation image arrays ( $n_1 \times n_2$ ) to one dimensional vectors( $1 \times n_1 \bullet n_2$ ) as feature vectors for classifier.

**Table 1.** The Coefficients of 3-band HARR Filter Banks

Decomposition filter	Filter coefficients	Reconstruction filter	Filter coefficients
H	1/3, 1/3, 1/3	h	1,1,1
G1	-1/3 ,1/3,0	g1	1,1,-2
G2	0,1/3,-1/3	g2	-2,1,1

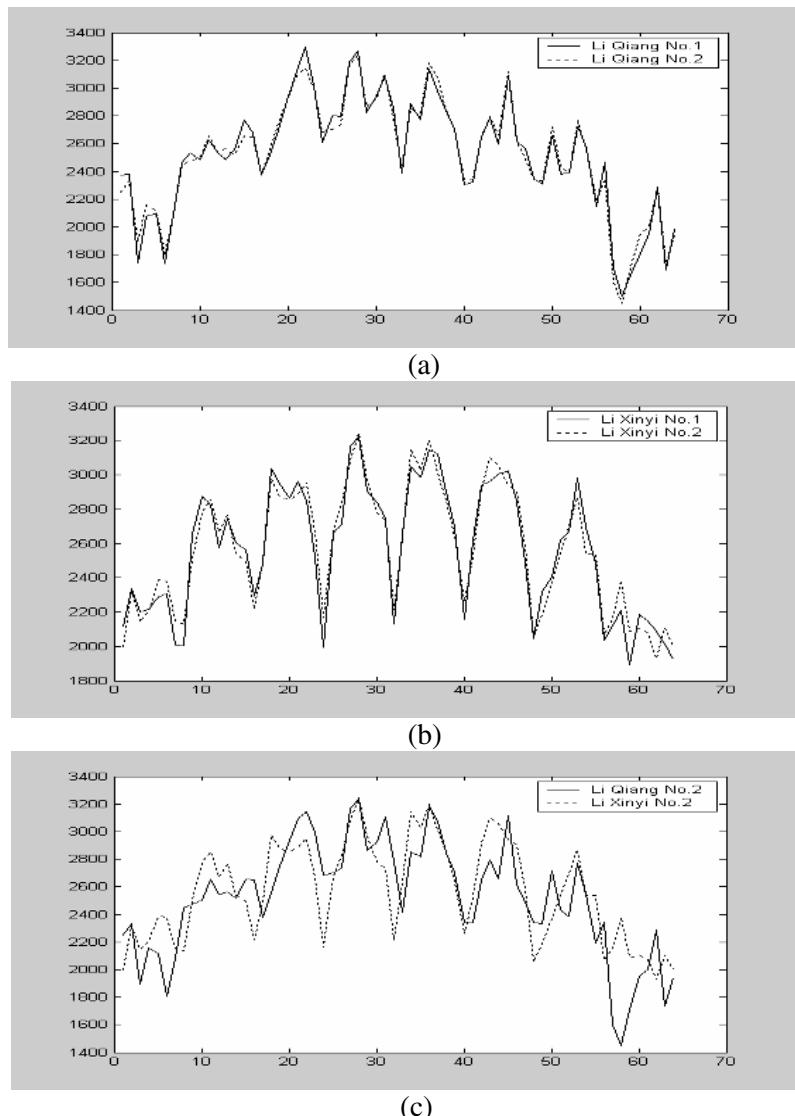


**Fig. 2.** Four palmprints from two similar palm and the corresponding WT low resolution subbands

- (a) Li Qiang's first palmprint and the corresponding DWT and TWT low subbands
- (b) Li Qiang's second palmprint and the corresponding DWT and TWT low subbands
- (c) Li Xinyi's first palmprint and the corresponding DWT and TWT low subbands
- (d) Li Xinyi's second palmprint and the corresponding DWT and TWT low subbands

1	2	3
4	5	6
7	8	9

- 1. Original palmprint    2. one-level DWT    3. two-level DWT
- 4. three-level DWT    5. four-level DWT    6. five-level DWT
- 7. one-level TWT    8. two-level TWT    9. three-level TWT low subband



**Fig. 3.** The 8\*8 approximation coefficients of Li Qiang and Li Xinyi's

- (a) The 8\*8 approximation coefficients of Li Qiang No.1 and Li Qiang No.2.
- (b) The 8\*8 approximation coefficients of Li Xinyi No.1 and Li Xinyi No.2.
- (c) The difference of (a) and (b).

### 3 LIB-SVM

LIB-SVM [6][7] is a library for support vector classification and regression. It is developed by Department of Computer Science and Information Engineering, National Taiwan University, Taipei 106, Taiwan (<http://www.csie.ntu.edu.tw/~cjlin>)

In the library, we use the C-Support Vector Classification for Binary Case and “one-against-one” Multi-class classification approach.

#### 3.1 C-Support Vector Classification (Binary Case)

Given training vectors  $x_i \in R^n$ ,  $i = 1, \dots, l$ , in two classes, and a vector  $y \in R^l$  such that  $y_i \in \{1, -1\}$ , C-SVC solves the following primal problem:

$$\begin{aligned} \min_{w, b, \xi} \quad & \frac{1}{2} w^T w + C \sum_{i=1}^l \xi_i \\ \text{subject to} \quad & y_i (w^T \phi(x_i) + b) \geq 1 - \xi_i, \\ & \xi_i \geq 0, i = 1, \dots, l. \end{aligned}$$

Its dual is

$$\begin{aligned} \min_{\alpha} \quad & \frac{1}{2} \alpha^T Q \alpha - e^T \alpha \\ \text{subject to} \quad & 0 \leq \alpha_i \leq C, i = 1, \dots, l, \\ & y^T \alpha = 0 \end{aligned}$$

Where  $e$  is the vector of all ones,  $C > 0$  is the upper bound,  $Q$  is an  $l$  by  $l$  positive semidefinite matrix,  $Q_{ij} \equiv y_i y_j K(x_i, x_j)$  and  $K(x_i, x_j) \equiv \phi(x_i)^T \phi(x_j)$  is the kernel. Here training vectors  $x_i$  are mapped into a higher (maybe infinite) dimensional space by the function  $\phi$

The decision function is

$$\text{sgn} \left( \sum_{i=1}^l y_i \alpha_i K(x_i, x) + b \right).$$

#### 3.2 Multi-class Classification

Several methods were proposed to extend SVM to multi-class case. The one-against-all method consists of constructing as many SVC as classes. The  $i$  th SVC is trained with all the examples in the  $i$  th class with positive label and all other examples with negative labels. It results in a set of decision functions as many as the number of

classes. Then a test data  $x_i$  is attributed to the class with the largest decision value. Another popular method is called the one-against-one method.

Here, the LIB-SVM use the “one-against-one” approach in which  $k(k-1)/2$  classifiers are constructed and each one trains data from two different classes. For training data from the  $i$  th and the  $j$  th classes, we solve the following binary classification problem:

$$\begin{aligned} & \min_{w^{ij}, b^{ij}, \xi^{ij}} \quad \frac{1}{2} (w^{ij})^T w^{ij} + C \left( \sum_{t=1}^l (\xi_t^{ij})_t \right) \\ \text{subject to} \quad & \left( (w^{ij})^T \phi(x_t) \right) + b^{ij} \geq 1 - \xi_t^{ij}, \text{ if } x_t \text{ is the } i \text{ th class,} \\ & \left( (w^{ij})^T \phi(x_t) \right) + b^{ij} \leq -1 + \xi_t^{ij}, \text{ if } x_t \text{ is the } j \text{ th class,} \\ & \xi_t^{ij} \geq 0. \end{aligned}$$

In classification we use a voting strategy: each binary classification is considered to be a voting where votes can be cast for all data points  $x$ , in the end point is designated to be in a class with maximum number of votes.

## 4 Experiment Results

We use a palmpoint database supported by Beijing Jiaotong University. All of images from this database come from the right hands of 50 different persons and are preprocessed. 5 samples are taken for each person, in which 4 samples are used for training and other samples are used for testing the recognition rate. The size of the central part of a palm is  $128 \times 128$ . The graylevel of these images is 256.

The correct recognition implies that for each sample in the testing set, the classifier answer the question: which palm is this image from. The correct recognition rate is defined as follows:

$$\text{The correct rate} = \frac{\text{numbers of correctly recognized images}}{\text{the number of testing samples}}$$

In v-fold cross-validation, we first divide the training set into v subsets of equal size. Sequentially one subset is tested using the classifier trained on the remaining  $v - 1$  subsets. Thus, each instance of the whole training set is predicted once so the cross-validation accuracy is the percentage of data which are correctly classified. Commonly, v is set to 10, and 10-fold cross-validation rate is computed.

For every decomposition strategy, the total time used (feature extraction of 250 palmpoint images + training process+ testing process) is from 10 to 20 seconds. So our scheme is little time consuming.

The testing results are in Table 2 and Table 3.

**Table 2.** The comparison of K-nearest classifier and SVM classifier for palmprint recognition

	image size	feature points	K-nearest recognition rate	SVM recognition rate
three-level DWT	16×16	256	96%	100%
four-level DWT	8×8	64	94%	100%
five-level DWT	4×4	16	72%	96%
three-level TWT	8×8	64	86%	98%

**Table 3.** The correct recognition rate

	image size	feature points	recognition rate	10-fold cross validation rate
one-level DWT	64×64	4096	100%	96%
two-level DWT	32×32	1024	100%	96%
three-level DWT	16×16	256	100%	96%
four-level DWT	8×8	64	100%	94%
five-level DWT	4×4	16	96%	86.5%
one-level TWT	45×45	2025	100%	96%
two-level TWT	17×17	289	100%	95.5%
three-level TWT	8×8	64	98%	91%

## 5 Conclusions

The presented experimental results demonstrated that our scheme of using wavelet and SVM is fast and accurate for palmprint categories. According to Fig.2 and Fig.3 the low resolution approximations resulted from 2-band and 3-band wavelet decomposition are similar in the images from the same person's palm and quite different for two person's palm, so are discriminable. Table 2 shows that SVM classifier is more accurate and more suitable for palmprint recognition than K-nearest method. Table 3 represents that the support vector classifier has high performance on few samples and high dimensional data categories. Considering the correct recognition rate and other factors such as storage, computation complex and time consuming, the four-level DWT is recommended.

In the future work, we will investigate the SVM's classification performance using scalable detailed WT coefficients. Thanks to Professor Zhengding Qiu (Beijing Jiaotong University) and his graduate students, they provided us the palmpoint database.

## References

1. Nicolae Duta, Anil K.Jain, Kanti V.Mardia: Matching of palmpoints. Pattern Recognition Letters 23 (2002) 477-485
2. Wangmeng Zuo, Kuanquan Wang, David Zhang: Bi-Directional PCA with Assembled Matrix Distance Metric. Image Processing, 2005. ICIP 2005. IEEE International Conference on Volume 2, 11-14 Sept. 2005 Page(s):II - 958-61
3. Guangming Lu , Kuanquan Wang: Wavelet Based Independent Component Analysis for Palmpoint Identification. Proceedings of the Third International Conference on Machine Learning and Cybernetics. Shanghai , China , 2004: 3547-3550
4. Xianqian Wu, Kuanquan Wang, David Zhang: Wavelet based Palmpoint Recognition. ICMLC2002, Nov.3-7, Beijing
5. P.Steffen, P.N.Hellor, R.A.Gopinath, and C.S.Burrus: Theory of regular M-Band wavelet Bases. IEEE Trans., Signal Processing, vol..41,pp.3497-3511, Dec.1993
6. C. Chih-Chung and L. Chi-Jen: LIBSVM: A Library for Support Vector Machines. 2004
7. Christopher JC Burges: A Tutorial on Support Vector Machines for Pattern Recognition. Data Mining and Knowledge Discovery 2:121 - 167, 1998

# Context Awareness System Modeling and Classifier Combination

Mi Young Nam, Suman Sedai, and Phill Kyu Rhee

Dept. of Computer Science & Engineering, Inha University  
253, Yong-Hyun Dong, Incheon, Korea  
{rera, suman}im.inha.ac.kr, pkrhee@inha.ac.kr

**Abstract.** This paper proposes a novel classifier combination system that can be used by classification systems under dynamically varying environments. The proposed method adopts the concept of context-awareness and the similarity between classes, and the system working environments are learned (clustered) and identified as environmental contexts. The proposed method fitness correlation table is used to explore the most effective classifier combination for each identified context. We use t-test for classifier selection and fusion decision and proposed context modeling and t-test. The group of selected classifiers is combined based on t-test decision model for reliable fusion. The knowledge of individual context and its associated chromosomes representing the optimal classifier combination is stored in the context knowledge base. Once the context knowledge is accumulated the system can react to dynamic environment in real time.

## 1 Introduction

In contrast to the huge amount of research in this active area [1, 2, 3], little work has been done on combining the specific classifier: the k nearest neighbor classifier (kNN). Classifier decision methods for identification are illustrated their better reliance on recognition than single classifier and implemented in various ways. Clustering the data set into different regions is added value to recognition systems by finding specific sophisticated system for particular region in ways as selection and fusion of classifiers [4, 5, 6].

This approach will use the ensemble as an individual means and covariance in a class. In this paper, we propose the method Bayesian based similarity between classes that is clustered using unsupervised method for varying illuminant. In additional method, we discuss about an adaptive run-time framework of classifier combination by employing the concept of context-awareness and the genetic algorithm. The proposed method can construct most effective classifier combination by selecting preprocessing, feature representation. System contexts associated with the system working environments are learned by clustering them. Context learning can be performed by an unsupervised learning method such as SOM, FuzzyArt, etc. We assume that an environmental context changes continuously. That is, we exclude an impulse style of context changes in this paper. The structures of classifier

combination are encoded as artificial chromosomes, and genetic algorithm is used to explore a most effective classifier combination for each identified context. The knowledge of an individual context and its associated chromosomes of effective classifier combination are stored in the context knowledge base. There is no general approach or theory for efficient classifier combination, yet [6, 7]. We assume that classifier systems include preprocessing, feature representation, and class decision components. The design of classifier combination can be divided into two stages: 1) the generation of possible classifiers using the classifier components and 2) the decision of classifier combination scheme using an aggregation method. The classifier components are a set of (combined) preprocessing, feature representation, and class decision methods. We assume that  $s$  (combined) preprocessing,  $t$  different feature representation,  $u$  class decision, methods are provided, and we focus on combining them to produce a output of high performance in terms of accuracy. The proposed method primarily aims at robust object recognition under uneven environments by selection and fusion of different classifiers. We method searches the group of most effective classifier system for each environmental context by proposed t-test method. The chosen best classifiers are then combined using t-test based decision model derived from the reliability condition of combination. The main difference of the proposed classifier combination method from other methods is that it can combine classifiers in accordance with the identified context in reliable way. The proposed method adopts the strategy of context knowledge accumulation so that classifier combination scheme is able to adapt itself under changing environments based on experience. Once the context knowledge base is constructed, the system can react to changing environments at run-time.

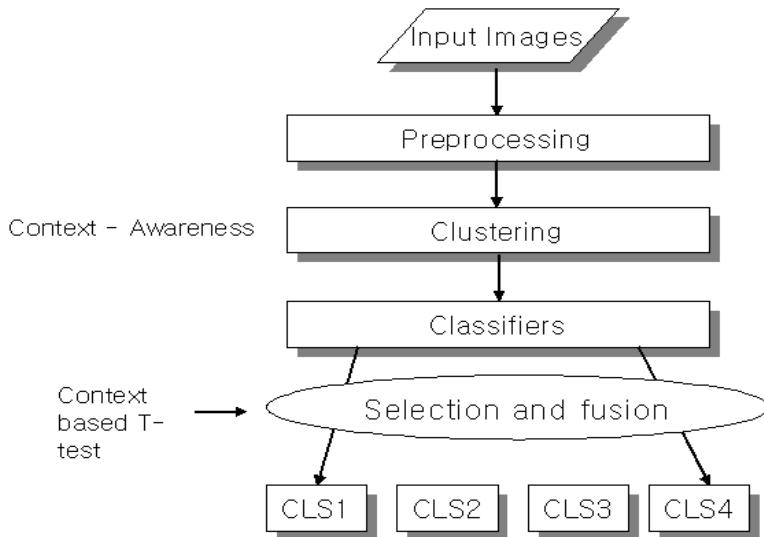
## 2 Model of Context-Aware Classifier Combination Scheme

In this section, we discuss about the model of context-aware classifier combination scheme with the capability of adaptation. The implementation issues of the proposed adaptive framework employing the context-awareness and the genetic algorithm will be discussed in the next section.

### 2.1 Exhaustive Classifier Combination

There is no general approach or theory for efficient classifier combination, yet [8]. we discuss about the framework of context-aware classifier combination with the capability of adaptation using accumulated context knowledge. The implementation issue of the proposed adaptive framework employing the context-awareness and the genetic algorithm will be discussed in the next section. An example of classifier combination scheme is given in Fig. 1.

Classifier combination scheme combines classifier components and produce the combined classifier system to achieve high performance. The model of classifier combination can be described as follows. Assuming that there are  $c$  possible classes  $\Omega = \{\omega_1, \omega_2, \dots, \omega_c\}$ . Let  $R^n$  be an input space. The input of each classifier system can be represented by input vector  $x \in R^n$  i.e.  $x = \{x_1, x_2, \dots, x_n\}^T$ . Then the

**Fig. 1.** Classifier Combination scheme

individual classifier system assigns class label from  $\Omega$  to an input vector  $x$ . That is,  $CLS_i : R^n \rightarrow \Omega$  with  $i = 1, 2, k$ . Where  $k$  is total number of classifier possible. Then the output of classifier  $CLS_i(x)$  can be represented as a following vector.

$$CLS_i(x) = [O_{i,1}(x), O_{i,2}(x), \dots, O_{i,c}(x)] \quad (1)$$

where  $O_{i,j}(x)$  is the output derived from  $CLS_i$  using the input vector  $x$ . Finally, the output of classifier combination can be described as follows.

$$CCS(x) = CCO(CLS_1(x), CLS_2(x), \dots, CLS_k(x)) \quad (2)$$

where  $ACCS$  denotes a classifier combination scheme and  $CCS(x)$  denotes the result of  $ACCS$ .  $CCO$  is a classifier combination operator and denotes classifier combining method, i.e. classifier selection and classifier fusion methods.

## 2.2 Context-Aware Classifier Combination

In general, the total number of classifiers  $k$  is too huge to be evaluated for finding optimal output .So context awareness is used to search the best classifiers from classifiers bank. Here two types of data inputs are used: context data and action data inputs to  $ACCS$ . The action data, denoted by  $x$ , is normal data being processed and context data, denoted by  $y$ , is used to identify environmental context of system and to control classifier combination. In many cases, the action data itself can be used as the context data. Thus, the proposed context-aware classifier combination can be described as follows.

$$ACCS(x, y) = CCO(CKO(y, (CLS_1(x), CLS_2(x), \dots, CLS_k(x)))) \quad (3)$$

where  $ACCS$  represents a context-aware classifier scheme, and  $CKO$  is a context-aware knowledge operator.  $CLS_i$  is an  $i$ -th classifier.  $CKO$  is used to select the best  $r$  classifiers out of  $k$  total combinations for an identified environmental context. Then Classifier Combination Operator ( $CCO$ ) is carried out on the  $r$  classifiers selected by  $CKO$  to give the final result. The context knowledge for  $CKO$  is accumulated by some learning method and stored in the  $CKB$  (Context Knowledge Base) as a context profile (see Fig.2). The details of implementation issues will be discussed in the next section. An example of context-aware classifier combination is depicted in Fig. 2. The intermediate feature space considering an identified environmental context is organized by the activated elements in  $DP(x)$ . The activated elements in  $DP(x)$  are decided by applying context-aware knowledge operator  $CKO$ .  $CKO$  is denoted as the activating vector  $CP$ , called context profile. The element of  $CP_i(y)$  in  $CP$  indicates whether the corresponding classifier  $CLS_i$  is activated or not under a given environmental context  $y$ . That is, if  $CP_i(y) = 1$ , then  $CLS_i$  is activated otherwise,  $CLS_i$  is deactivated.

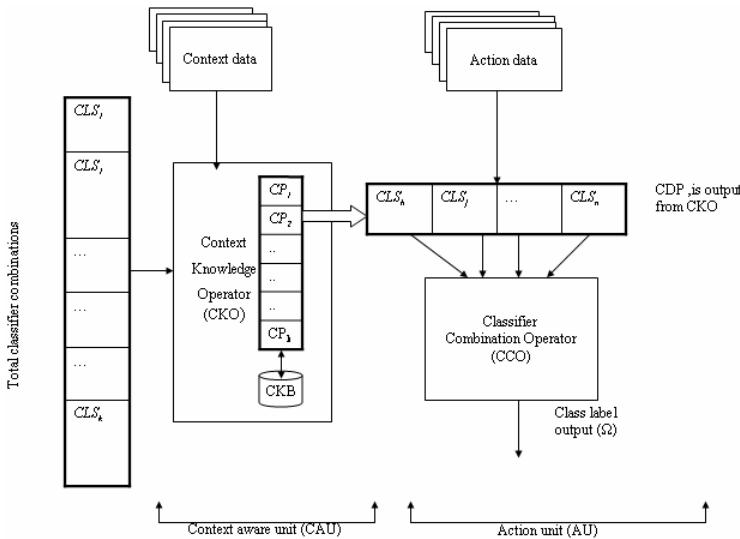
Figure 2 shows that a subset of classifier combinations is selected by Context Knowledge Operator ( $CKO$ ) filter. Here CAU uses  $CP$  to calculate  $CDP$  from total classifiers combinations. The classifier output is represented as a context decision profile as shown in the following equation assuming that only four classifiers are activated by  $CKO$ .

$$CDP(x, y) = CP^t(y) \times DP(x) = \begin{bmatrix} CLS_h(x) \\ CLS_j(x) \\ CLS_m(x) \\ CLS_n(x) \end{bmatrix} = \begin{bmatrix} O_{h,1}(x), O_{h,2}(x), \dots, O_{h,c}(x) \\ O_{j,1}(x), O_{j,2}(x), \dots, O_{j,c}(x) \\ O_{m,1}(x), O_{m,2}(x), \dots, O_{m,c}(x) \\ O_{n,1}(x), O_{n,2}(x), \dots, O_{n,c}(x) \end{bmatrix} \quad (4)$$

where  $CLS_h$ ,  $CLS_j$ ,  $CLS_m$ , and  $CLS_n$ , are classifiers activated by  $CKO$  see Fig 2. Now original problem of classifier combination  $P(\omega_i|x)$  can be transformed into the problem of estimating  $P(\omega_i|CDP(x,y))$ . Hence classifier decision is also conditionally related to the context  $y$ . Note that only the activated classifiers output are included  $CDP$ . Let the number of activated classifiers, involving in classifier fusion is  $r$ . Then,  $r \ll k$  where  $k$  is total number of possible classifier combinations. Note that  $r$  is controllable parameter. The output of context-aware classifier combination scheme can be described as follows from equation 6.

$$ACCS(x, y) = CCO(CDP(x, y)) \quad (5)$$

where  $ACCS$  denotes the classifier combination scheme using context awareness.  $ACCS(x, y)$  denotes the output of  $ACCS$  using an input  $x$  under a context data  $y$ . The main difference of  $ACCS$  from the  $CCS$  is that it can select a small subset of classifiers in accordance with an identified environmental context. Furthermore,  $ACCS$  can store its accumulated knowledge in the  $CKB$ . Hence, once  $CKB$  is constructed, the scheme can react in real-time. It can be used in dynamically varying environments in real-time while the  $CCS$  can hardly be used under such a dynamic environments.



**Fig. 2.** Context Aware Classifier Combination Scheme

### 3 Framework of Context-Aware Classifier Combination Scheme

#### 3.1 Bayesian Classifier

Bayesian classifier is classifier based on probabilities. We propose the method that has estimate final result computed decision value each classifier.

##### Naïve Bayes Classifier fusion

The Naive Bayes classifier consists of two parts, training and classifying. We have to able to specify which columns should be looked at and which should be looked at and which should be skipped by the classifier during training. A “feature string” is used, a string of face feature.

##### The training part

The training part of the algorithm works by counting instances of particular class-feature attribute combination. This is done by using a hash table, with the class-feature-attribute combination used as a key for the hash table. The classifying part: we can now present it new samples which it has to classify. The classifier will return the class of the sample with a measure of probability.

##### Bagging and Boosting

Boosting stands for bootstrap aggregation and is discussed in the book. An introduction to the Bootstrap by Efron and Tibshirani[ ]. Bagging works in the following manner : each classifier will pick a random number of training data points from dataset. The idea is that each classifier will see different training data points and thus will make different decision.

### Single Classifier

The eigenface is constructed registration images of FERET, Yale, Inha database.

First step: We made in covariance matrix of registration data. Next figure is shown eigen-face of each person. The eigenface is belong to global recognition.

The registration data computed covariance matrix.

- a) Eigenface
  - b) Gabor9 : This classifier select the 9 feature extraction results
  - c) Gabor13 : The face features are selected 13 points
  - d) Gabor28 : The face features are selected 28 pints
  - e) Gabor30: The face features are selected 30points
  - f) Individual best (IB) : IB indicates the highest recognition rate among single classifiers.
- 2) Super combination (SC) : The super combination obtains a correct result if the correct class label is produced vt at least one individual classifier. It is not real combination, and just gives an abstraction of an ideal classifier combination. It should be excluded from comparison.
- 3) Classifier fusion methods
- a) Majority voting (MV) : simple voting, also called majority voting and select all majority[ X], considers each component classifier as an equally weighted vote.
  - b) Decision Template (DT) :
  - c) Maximum based fusion (MX)
  - d) Minimum based fusion (MN)
  - e) Naive Bayes (NB)
- 4) Classifier Selection (3, 6, 9):
- a) Dynamic classifier selection with the number of environmental contexts 3, 6, and 9 denoted by DCS(3), DCS(6), and DCS(9), respectively.
  - b) Static classifier selection : SCS(3), SCS(6), and SCS(9).

## 4 T-Test Decision Model for Classifier Combination

In this section we discuss about the statistical model to achieve the reliable fusion system for combining the results from several constituent classifiers. Fusion gives better result when the classifiers being fused are closed related to each other. Kunchiva have applied a paired t-test to find whether the best classifier is different enough from rest in decision space [5]. Here we try to estimate the fusion condition to achieve the reliability of the fusion system without much loss in accuracy. We want to increase the lower limit of the recognition rate of the fusion system than that of best classifier, hence achieving higher reliability. Lets us consider a restricted case of only two classifiers  $CLS_A$  and  $CLS_B$ , where  $CLS_A$  is found to be best one in decision space.

Classifier  $CLS_A$  recognition rate is normally distributed with mean  $\hat{P}_A$  and standard deviation  $\delta_A$  and the distribution is denoted by  $P_A \sim N(\hat{P}_A, \delta_A)$ . Similarly for classifier  $CLS_B$  distribution is  $P_B \sim N(\hat{P}_B, \delta_B)$ . Then distribution of recognition rate

after fusion is  $P_F \sim N(\hat{P}_F, \delta_F)$ . If  $\hat{P}_F = a\hat{P}_A + b\hat{P}_B$  such that  $a+b=1$ . It can be shown that standard deviation of recognition rate of fusion is

$$\delta_F = \sqrt{a^2 \delta_a^2 + b^2 \delta_b^2} \quad (6)$$

Equation 13 gives the condition of better fusion, in order to increase the reliability by increasing lowest recognition rate of best classifier ( $P_A-S_A$ ) to  $P_F-S_F$ . Four fusion methods namely Decision Templates (DT)[6], Majority Voting (MV)[9], Product and Average [9] are used for aggregation of classifier output when t-test decides for fusion.

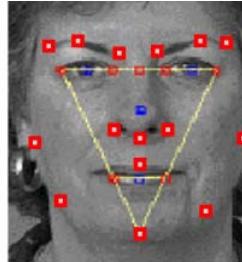
## 5 Experiment on Face Recognition

The proposed method was tested in the area of face recognition using standard FERET database. Its performance was evaluated through extensive experiments, and shown to be reliable and superior to those of most popular methods, especially under changing illumination. Face images are used as context data as well as action input data. The changes in image data under changing illumination are modeled as environmental contexts. Context aware Unit (CAU) clusters face data into several distinguishable contexts according to illumination variations. The model of illumination environmental contexts is constructed by considering light direction and brightness. The CAU is implemented by Kohonen's self-organizing map (SOM) [6,7] and Radial basis function (RBF). SOM has the capability of unsupervised learning. It models illumination environment as several context categories. The RBF neural network is trained using the clustered face data in order to identify the context category of an input image. In the AU, histogram equalization (HE) is used for preprocessing components and Gabor wavelet is used as feature representation. Gabor wavelet shows desirable characteristics in orientation selectivity and special locality.

### **Classifier Gabor3, Gabor13, Gabor 28, Gabor30**

The Gabor wavelet transform guided by an evolutionary approach has been employed to adapt the system for variations in illumination. The proposed approach employs Gabor feature vector, which is generated from the Gabor wavelet transform. Gabor wavelet is biologically motivated convolution kernels in the shape of plane waves restricted by gabor kernel. Gabor wavelet has shown to be particularly fit to image decomposition and representation. Gabor wavelet shows desirable characteristics in orientation selectivity and special locality. Face Gabor vector is generated as shown Fig. 3. The feature is extracted 19 feature points. We adopt four Gabor based classifiers: Gabor3, Gabor13, Gabor28, Gabor30. They are different from each other only in the number of feature points.

The knowledge of effective classifier structure for a data context is described by the pair of data context category and corresponding artificial chromosome representing the best classifiers for that context.



**Fig. 3.** An example of feature points for face recognition

### 5.1 Experimental Results

The proposed method is tested in the area of face recognition using four data sets: Inha, FERET, AR, Yale. Its performance is evaluated through extensive experiments to be superior to those of most popular methods, especially in changing illumination.

The first experiment is performed using the data set accumulated by our lab. The data set has 1000 face images from 100 people. We used 5 images for registration for each person. The remaining 500 images are used as the normal images. We used 99 registration images and 198 test images from the AR face image set excluding images wearing sunglasses. For the Yale data set, we used 15 registration face image and 45 test images. The FERET gallery images of 1196 people are used for registration and 1196 images of probe\_fafb\_expression dataset are used for test. The following individual classifiers and classifier combinations are evaluated and compared. Table 1 shows the result of fusion of each context for six and three context model. It shows that t-test decides one of best selection or fusion for each context for reliable fusion. Since it was stated that fusion method is not a primary factor when the ensemble contains diverse set of classifiers[1]. This gives insight that our classifier selection method has chosen optimal set of classifiers for each context.

Table 2 shows the result of six experiments and comparison our proposed method and other methods. Our proposed method is highest performance other method.

**Table 1.** Face recognition rate of the proposed classifier fusion and applied dynamic cluster

	YaleDB_All	YaleDB_Del		FERET	
Gabor28	90.90%	96.63%	oracle	90.98%	oracle
FitCorr[6] G28	90.60%	93.26%	100.00%	91.80%	93.44%
FitCorr[3] G28		98.88%	100.00%	90.98%	92.62%
FitCorr[2] G28		96.63%	97.75%	90.98%	91.80%
Gabor13		93.26%	Oracle	90.16%	oracle
FitCorr[6] G13		95.51%	100.00%	91.80%	94.26%
FitCorr[3] G13		96.63%	100.00%	90.98%	92.62%
FitCorr[2] G13		93.26%	94.38%	90.98%	91.80%

**Table 2.** Single Classifier Experimental Results

	<b>Ferret</b>	<b>Yale</b>	<b>Our Lab</b>
Eigenface	60.35	58.91	94.2
Gabor3	59.59	68.99	93.72
Gabor13	64.96	77.83	94.44
Gabor28	82.06	80.46	95.79
Gabor30	82.06	80.46	95.79
MV	84.58	82.17	98.41
MX	84.66	80.46	97.14
MN	8.38	10.07	8.73
NB	85.16	83.1	97.77
Product	78.62	82.63	90.94
Average	83.74	84.03	97.22
t-test (proposed Method)	92.11	94.2	99.52

## 6 Conclusion

Since no single classifier is best for all environmental contexts we choose the best classifiers for each context. The fusion of chosen best classifiers of a context may not always prove to be effective, so t-test is performed to check the effectiveness of the fusion in terms of reliability. This paper proposes a novel run-time classifier combination framework that can be used by classification systems by adopting the concept of context-awareness and classifier fusion t-test and classifier fusion. If expected fusion seems to be unreliable then best selection is chosen. The proposed method tested on face recognition system shows the reliable result across varying environmental context. Our context – based classifier is efficient for face recognition in varying environment data. Furthermore, it solves the time-consuming problem of the GA by taking advantage of the concept of context-aware architecture. The proposed framework has been tested in the area of face recognition where most approaches show vulnerability under dynamically changing environments. The superiority of the proposed scheme is shown using four data sets: Inha, FERET, and Yale.

## References

- [1] K. I. Diamantaras and S.Y. Kung, Principle Component Neural Networks: Theory and Application, Johj Wiley and Sons, (1996) .
- [2] M. Potzsch, N. Kruger, and C. Von der Malsburg, : Improving Object recognition by Transforming Gabor Filter reponses, Network: Computation in Neural Systems, 7(2), 341-347.

- [3] A. S. Georghiades, P. N. Belhumeur, and D. J. Kriegman, : From Few to Many: Illumination COne Models for face recognition under Variable Lighting and Pose, IEEE Trans. on PAMI, Vol. 23, No. 6, (2001) 643-660.
- [4] C. Liu and H. Wechsler : Evolutionary Pursuit and Its Application to Face recognition, IEEE Trans. on PAMI, vol. 22, no. 6, (2000) 570-582.
- [5] Ludmila I. Kuncheva, Switching Between Selection and Fusion in Combining Classifiers: An Experiment, IEEE Transactions on Systems, Man, and Cybernetics - part B: cybernetics, Vol.32, Vo.2, (2002) 146-156.
- [6] Ludmila I. Kuncheva, : A Theoretical Study on Six Classifier Fusion Strategies IEEE S on PAMI, Vol. 24, No. 2, (2002).
- [7] L. I. Kuncheva, J. C. Bezdek, and R. P. W. Duin, “Decision templates for multiple classifier fusion: An experimental comparison,” Pattern Recognit., Vol. 34, No. 2, (2001) 299–314.
- [8] S. Yau, F. Karim, Y. Wang, B. Wang, and S. Gupta, Reconfigurable Context-Sensitive Middleware for Pervasive Computing, IEEE Pervasive Computing, Vol.1, No.3, (2002) 33-40.
- [9] Kuncheva L.I. and L.C. Jain, Designing classifier fusion systems by genetic algorithms, IEEE Transactions on Evolutionary Computation, Vol.4, No.4, (2000) 327-336.

# Non-negative Matrix Factorization on Kernels

Daoqiang Zhang<sup>1,2</sup>, Zhi-Hua Zhou<sup>2</sup>, and Songcan Chen<sup>1</sup>

<sup>1</sup> Department of Computer Science and Engineering  
Nanjing University of Aeronautics and Astronautics,  
Nanjing 210016, China  
`{dqzhang, s.chen}@nuaa.edu.cn`

<sup>2</sup> National Laboratory for Novel Software Technology  
Nanjing University, Nanjing 210093, China  
`zhouzh@nju.edu.cn`

**Abstract.** In this paper, we extend the original non-negative matrix factorization (NMF) to kernel NMF (KNMF). The advantages of KNMF over NMF are: 1) it could extract more useful features hidden in the original data through some kernel-induced nonlinear mappings; 2) it can deal with data where only relationships (similarities or dissimilarities) between objects are known; 3) it can process data with negative values by using some specific kernel functions (e.g. Gaussian). Thus, KNMF is more general than NMF. To further improve the performance of KNMF, we also propose the SpKNMF, which performs KNMF on sub-patterns of the original data. The effectiveness of the proposed algorithms is validated by extensive experiments on UCI datasets and the FERET face database.

## 1 Introduction

Many data analysis tasks in machine learning require a suitable representation of the data. Typically, a useful representation can make the latent structure in the data more explicit, and often reduces the dimensionality of the data so that further computational methods can be applied [6]. Non-negative matrix factorization (NMF) [7] [8] is a recent method for finding such representation. NMF imposes the non-negativity constraints in its bases and coefficients. Due to its part-based representation property, NMF and its variations have been applied to image classification [2] [5], face expression recognition [3], face and object recognition [9] [10] [12], document clustering [13], etc.

However, NMF and many of its variants are essentially linear, and thus cannot disclose nonlinear structures hidden in the data. Besides, they can only deal with data with attribute values, while in many applications we do not know the detailed attribute values and only the relationships (similarities or dissimilarities) are available. NMF cannot be directly applied to such relational data. Furthermore, one requirement of NMF is that the values of data should be non-negative, while in many real-world problems the non-negative constraints can not be satisfied.

In this paper, we propose the kernel NMF (KNMF), which can overcome the above limitations of NMF. First, through using kernel-induced nonlinear mapping, KNMF

could extract more useful features hidden in the original data. Second, we develop a method for KNMF to deal with data where only relationships between objects are known. Third, by using some specific kernel functions (e.g. Gaussian), KNMF can process data with negative values. Thus, KNMF is more general than NMF. Moreover, inspired by successes of so many 2D pattern representation methods [4] [14] and to further improve the performance of KNMF, we also propose the SpKNMF, which performs KNMF on sub-patterns of the original data. The effectiveness of the proposed algorithms is validated by extensive experiments on several UCI datasets and the FERET database for face recognition.

The rest of the paper is organized as follows: Section 2 introduces NMF briefly. This is followed by the detailed description of the KNMF algorithm in Section 3. In Section 4, we present the SpKNMF algorithm. In Section 5, experimental results are reported. Finally, we conclude this paper and raise some issues for future research in Section 6.

## 2 Non-negative Matrix Factorization

The key ingredient of NMF is the non-negativity constraints imposed on matrix factors. Assume that the observed data of the objects are represented as an  $n \times m$  matrix  $V$ , each column of which contains  $n$  non-negative attribute values of one of the  $m$  objects. In order to represent the data or reduce the dimensionality, NMF finds two non-negative matrix factors  $W$  and  $H$  such that

$$V_{i\mu} \approx (WH)_{i\mu} = \sum_{a=1}^r W_{ia} H_{a\mu} \quad (1)$$

Here the  $r$  columns of  $W$  are called NMF bases, and the columns of  $H$  are its combining coefficients. The dimensions of  $W$  and  $H$  are  $n \times r$  and  $r \times m$ , respectively. The rank  $r$  of the factorization is usually chosen such that  $(n+m)r < nm$ , and hence the dimensionality reduction is achieved.

To find an approximate factorization  $V \approx WH$ , a cost function is needed to quantify the quality of the approximation. NMF uses the divergence measure as the objective function

$$D(V \parallel WH) = \sum_{i,j} \left( V_{ij} \log \frac{V_{ij}}{(WH)_{ij}} - V_{ij} + (WH)_{ij} \right) \quad (2)$$

NMF factorization is a solution to the following optimization problem: minimize  $D(V \parallel WH)$  with respect to  $W$  and  $H$ , subject to the constraints  $W, H \geq 0$ , i.e. all terms in the matrix are non-negative. In order to obtain  $W$  and  $H$ , a multiplicative update rule is given in [11] as follows

$$W_{ia} = W_{ia} \sum_{\mu=1}^m \frac{V_{i\mu}}{(WH)_{i\mu}} H_{a\mu} \quad (3a)$$

$$W_{ia} = \frac{W_{ia}}{\sum_{j=1}^n W_{ja}} \quad (3b)$$

$$H_{a\mu} = H_{a\mu} \sum_{i=1}^n W_{ia} \frac{V_{i\mu}}{(WH)_{i\mu}}. \quad (3c)$$

### 3 Kernel Non-negative Matrix Factorization

Given  $m$  objects  $\mathcal{O}_1, \mathcal{O}_2, \dots, \mathcal{O}_m$ , with attribute values represented as an  $n$  by  $m$  matrix  $V=[v_1, v_2, \dots, v_m]$ , each column of which represent one of the  $m$  objects. Define the nonlinear map from original input space  $V$  to a higher or infinite dimensional feature space  $F$  as follows

$$\phi: x \in V \rightarrow \phi(x) \in F \quad (4)$$

For the  $m$  objects, denote

$$\phi(V) = [\phi(v_1), \phi(v_2), \dots, \phi(v_m)] \quad (5)$$

Similar as NMF, KNMF finds two non-negative matrix factors  $W_\phi$  and  $H$  such that

$$\phi(V) = W_\phi H \quad (6)$$

Here,  $W_\phi$  is the bases in feature space and  $H$  is its combining coefficients, each column of which denotes now the dimension-reduced representation for the corresponding object. It is worth noting that since  $\phi(V)$  is unknown, it is impractical to directly factorize  $\phi(V)$ . In what follows, we will derive a practical method to solve this problem. From Eq. (6), we obtain

$$(\phi(V))^T \phi(V) = (\phi(V))^T W_\phi H \quad (7)$$

Before further explaining the meaning of Eq. (8), we first give the definition of kernels. According to [15], a kernel is a function in the input space and at the same time is the inner product in the feature space through the kernel-induced nonlinear mapping. More specifically, a kernel is defined as

$$k(x, y) = \langle \phi(x), \phi(y) \rangle = (\phi(x))^T \phi(y) \quad (8)$$

Some commonly-used kernels in literature are [15]:

(1) Gaussian kernel

$$k(x, y) = \exp\left(\frac{-\|x - y\|^2}{\sigma^2}\right) \quad (9)$$

(2) Polynomial kernel

$$k(x, y) = (1 + \langle x, y \rangle)^d \quad (10)$$

(3) Sigmoid kernel

$$K(x, y) = \tanh(\alpha \langle x, y \rangle + \beta) \quad (11)$$

From Eq. (8), the left side of Eq. (7) can be rewritten as

$$(\phi(V))^T \phi(V) = \left\{ (\phi(v_i))^T \phi(v_j) \right\}_{i,j=1}^m = \left\{ k(v_i, v_j) \right\}_{i,j=1}^m \triangleq K \quad (12)$$

Denote

$$Y = (\phi(V))^T W_\phi \quad (13)$$

From Eqs. (12) and (13), Eq. (7) changes to

$$K = YH \quad (14)$$

Comparing Eq. (14) with Eq. (6), it can be found that the combining coefficient  $H$  is the same. Since  $W_\phi$  is the learned bases of  $\phi(V)$ , similarly we call  $Y$  in Eq. (14) as the bases of the kernel matrix  $K$ . Eq.(14) provides a practical way for obtaining the dimension-reduced representation  $H$  by performing NMF on kernels.

For a new data point, the dimension-reduced representation is computed as follows

$$\begin{aligned} H_{new} &= (W_\phi)^+ \phi(v_{new}) \\ &= (W_\phi)^+ ((\phi(V))^T)^+ (\phi(V))^T \phi(v_{new}) \\ &= Y^+ K_{new} \end{aligned} \quad (15)$$

Here  $A^+$  denote the generalized (Moore-Penrose) inverse of matrix  $A$ , and  $K_{new} = (\phi(V))^T \phi(v_{new})$  is the kernel matrix between the  $m$  training instances and the new instance.

Eqs. (14) and (15) construct the key components of KNMF when used for classification. it is easy to see that, the computing of KNMF need not know the attribute values of objects, and only the kernel matrix  $K$  and  $K_{new}$  are required. Note that some

kernels (e.g. Gaussian) can be seen as similarity measures between objects. Thus, the classification problem which KNMF can deal with is formulated as:

Given  $m$  training objects, we do not know their detailed attribute values, but the pair wise relationship between them can be measured (recorded in  $\mathbf{K}$ ). Also, the attribute values of the test object is not known, but the relationship between it and the training objects can be computed (recorded in  $\mathbf{K}_{new}$ ). Then, classify the new object into one of the training objects given  $\mathbf{K}$  and  $\mathbf{K}_{new}$ .

Obviously, KNMF is more general than NMF because the former can deal with not only attribute-value data but also relational data. Another advantage of KNMF is that it is applicable to data with negative values since the kernel matrix in KNMF is always non-negative for some specific kernels (e.g. Gaussian).

## 4 Sub-pattern Based KNMF

Given  $m$  objects  $\mathbf{O}_1, \mathbf{O}_2, \dots, \mathbf{O}_m$ , with attribute values represented by an  $n$  by  $m$  matrix  $\mathbf{V}=[\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_m]$ , each column of which represents one of the  $m$  objects. Assume  $n$  is divisible by  $p$ , then reassemble the original matrix  $\mathbf{V}$  into  $n/p$  by  $mp$  matrix  $\mathbf{U}$  as follows

$$\mathbf{U} = \left[ \mathbf{u}_1, \dots, \mathbf{u}_p, \mathbf{u}_{p+1}, \dots, \mathbf{u}_{2p}, \dots, \mathbf{u}_{(m-1)p+1}, \dots, \mathbf{u}_{mp} \right] \quad (16)$$

Here

$$\mathbf{v}_i = \left[ \mathbf{u}_{(i-1)p+1}^T, \dots, \mathbf{u}_{ip}^T \right]^T, i=1, 2, \dots, m \quad (17)$$

From Eq. (16), compute the  $mp$  by  $mp$  kernel matrix as

$$\mathbf{K} = (\phi(\mathbf{U}))^T \phi(\mathbf{U}) \quad (18)$$

Factorizing Eq. (18) using Eq. (14), we obtain the dimension-reduced representation  $\mathbf{H}=\{\mathbf{h}_j\}$  with dimension of  $r$  by  $mp$ , where  $r$  is the number of reduced dimensions. Then reassemble the matrix  $\mathbf{H}$  into  $rp$  by  $m$  matrix  $\mathbf{R}$  as

$$\mathbf{R} = \left\{ \mathbf{r}_i \right\}_{i=1}^m = \left\{ \left[ \mathbf{h}_{(i-1)p+1}^T, \dots, \mathbf{h}_{ip}^T \right]^T \right\}_{i=1}^m \quad (19)$$

Similarly, for some new data  $\mathbf{v}_{new}$ , first reassemble it into  $n/p$  by  $p$  matrix  $\mathbf{U}_{new}$  as follows

$$\mathbf{U}_{new} = \left[ \mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_p \right] \quad (20)$$

Here  $\mathbf{v}_{new} = \left[ \mathbf{u}_1^T, \dots, \mathbf{u}_p^T \right]^T$ . From Eq. (20), compute the  $mp$  by  $p$  kernel matrix as

$$\mathbf{K}_{new} = (\phi(\mathbf{U}))^T \phi(\mathbf{U}_{new}) \quad (21)$$

From Eq. (15), we can obtain the dimension-reduced representation  $H_{new} = [h_1, h_2, \dots, h_p]$  with dimension of  $r$  by  $p$ , where  $r$  is the number of reduced dimensions. Then reassemble the matrix  $H_{new}$  into  $rp$  by 1 vector  $R_{new}$  as

$$R_{new} = \begin{bmatrix} h_1^T, \dots, h_p^T \end{bmatrix}^T \quad (22)$$

Finally, Eqs. (19) and (22) can be used for classification. For example, if the nearest neighborhood classifier is adopted, then classify the new data point to the same class of  $i$ -th column vector of  $R$  with minimum distance from  $R_{new}$ .

## 5 Experiments

In this section, we present a set of experiments to evaluate our proposed algorithms: KNMF and SpKNMF, compared with traditional NMF, on several UCI Machine Learning Repository datasets [1] and the FERET face database [11]. In our experiments Gaussian kernel is adopted and the kernel width is set to the standard variance  $\sigma = \text{sqrt}(\sum_{j=1}^n \|x_j - \bar{x}\|^2 / n)$ . The nearest neighborhood classifier (1-NN) is used for classification. It is worthy noting that NMF, KNMF and SpKNMF are unsupervised dimensionality methods, and hence it is not comparable between them and some supervised dimensionality techniques or supervised classifiers.

### 5.1 UCI Data Sets

Four UCI datasets are used. No extra criterion is adopted for the selection of datasets except that the datasets should have relatively more dimensions and only numeric attributes without missing values are considered. Table 1 gives the statistics of them. For each dataset 10 independent runs are carried out and the results are averaged. At each run, half of the data are randomly picked for training, and the rest for testing. For each data set, we test the accuracies of NMF, KNMF and SpKNMF under different dimensions.

**Table 1.** Statistics of the UCI data sets

Dataset	Size	Dimension	# of classes
<i>Ionosphere</i>	351	34	2
<i>Bupa</i>	345	6	2
<i>Glass</i>	214	9	6
<i>PID</i>	768	8	2

Fig. 1 depicts the accuracies of NMF, KNMF and SpKNMF under different dimensions. It can be found that on all these data sets, SpKNMF consistently outperforms KNMF and NMF no matter which dimension is considered. For *Ionosphere* and *Glass*, KNMF outperforms NMF greatly. While for *Bupa* and *PID*, the performances of KNMF and NMF are close, while KNMF is slightly better under bigger dimensions.

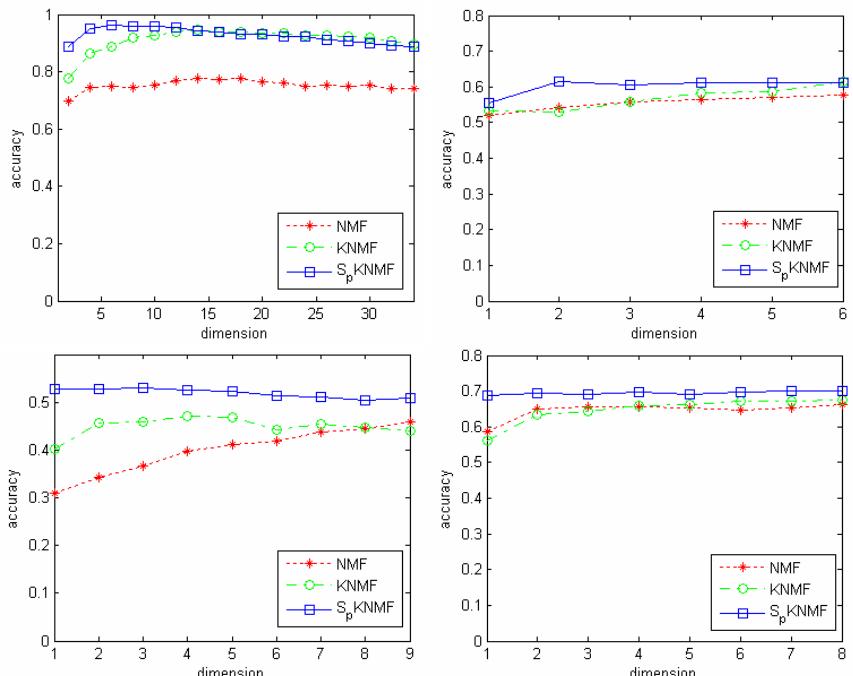
Table 2 shows the accuracies averaged across the dimensions shown in Fig. 1. From Table 2 it can be found that SpKNMF and KNMF outperform NMF on average, and SpKNMF always achieves the best performance (see the bold).

**Table 2.** Comparisons of averaged accuracies (%) under different dimensions (the values in the bracket are the sizes of the reassembled matrices)

Datasets	NMF	KNMF	SpKNMF
<i>Ionosphere</i>	75.24	91.24	<b>92.69(17x2)</b>
<i>Bupa</i>	55.58	56.79	<b>60.19(3x2)</b>
<i>Glass</i>	39.87	44.89	<b>51.88(3x3)</b>
<i>PID</i>	64.52	64.75	<b>69.51(4x2)</b>

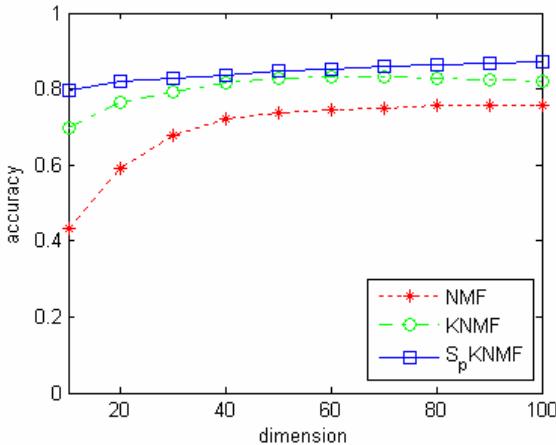
## 5.2 FERET Face Database

In this experiment, a partial FERET face database containing 400 gray-level frontal view face images from 200 persons are used, each of which is cropped with the size of  $60 \times 60$ . There are 71 females and 129 males; each person has two images (**fa** and **fb**) with different facial expressions. The **fa** images are used as gallery for training while the **fb** images as probes for test.



**Fig. 1.** Comparisons of accuracies under different dimensions on *Ionosphere* (top left), *Bupa* (top right), *Glass* (bottom left) and *PID* (bottom right)

Fig. 2 shows the accuracies of the three algorithms under different feature dimensions on the partial FERET face database. From Fig2 it can be found that SpKNMF and KNMF consistently outperform NMF no matter how many dimensions are used. SpKNMF is slightly superior to KNMF on this database. The accuracies of NMF, KNMF and SpKNMF averaged across all the dimensions shown in Fig. 2 are 69.23, 80.37 and 84.44, respectively. The performance of SpKNMF is still the best (the size of its reassembled matrix is 900x4). It is impressive that SpKNMF and KNMF achieve nearly 10% and 15% higher accuracies than NMF, respectively.



**Fig. 2.** Comparisons of accuracies under different dimensions on the partial FERET database

## 6 Conclusions

In this paper, KNMF is developed. Compared with conventional NMF, KNMF can: 1) extract more useful features hide in the original data using some kernel-induced nonlinear mapping; 2) deal with relational data where only the relationships between objects are known; 3) process data with negative values by using some specific kernel functions (e.g. Gaussian). Thus, KNMF is more general than NMF. Furthermore, another algorithm SpKNMF is proposed to further improve the performance of KNMF by performing KNMF on sub-patterns of the original data. Experimental results on UCI datasets and the FERET face database validated the effectiveness of the proposed algorithms.

There are several issues for future research. First, as in other kernel-based methods, the selection of kernels and their parameters is crucial for the performances of KNMF and SpKNMF. In this paper, we only consider the Gaussian kernel and set the kernel width to the standard variance. We will investigate how to adaptively choose the kernels and parameters in the future. Also, choosing the appropriate size for the reassembled matrix in SpKNMF is also an interesting issue for future research. Moreover, comparing KNMF and SpKNMF with other dimensionality reduction methods, such

as LDA and KFD, in particular, on their performances in classification, is left for future research.

## Acknowledgements

This work was supported by NSFC (Nos. 60505004 and 60473035), JiangsuSF (BK2005122), FANEDD (200343), the Jiangsu Postdoctoral Research Foundation, and funds from Shanghai Key Laboratory of Intelligent Information Processing. Portions of the research in this paper use the FERET database of facial images collected under the FERET program.

## References

1. Blake, C., Keogh, E., Merz, C.J.: UCI repository of machine learning databases [<http://www.ics.uci.edu/~mlearn/MLRepository.html>], Department of Information and Computer Science, University of California, Irvine, CA, 1998
2. Buchsbaum, G., Bloch, O.: Color categories revealed by non-negative matrix factorization of Munsell color spectra. *Vision Research* 42 (2002) 559-563
3. Buciu, I., Pitas,I.: Application of non-negative and local non-negative matrix factorization to facial expression recognition. In: ICPR, Cambridge, 2004
4. Chen, S.C., Zhu, Y.L.: Subpattern-based principle component analysis. *Pattern Recognition* 37 (5) 1081-1083
5. Guillamet, D., Bressan, M., Vitria, J.: A weighted non-negative matrix factorization for local representation. In: CVPR, Hawaii, 2001
6. Hoyer, P.O.: Non-negative matrix factorization with sparseness constraints. *Journal of machine Learning Research* 5 (2004) 1457-1469
7. Lee D.D., Seung, H.S.: Learning the parts of objects by non-negative matrix factorization. *Nature* 401(1999) 788-791
8. Lee D.D., Seung, H.S.: Algorithms for non-negative matrix factorization. In: NIPS 13 (2001) 556-562
9. Li, S.Z., Hou, X.W., Zhang, H.J., Cheng, Q.S.: Learning spatially localized, parts-based representation. In: CVPR, Hawaii, 2001.
10. Liu, W., Zheng, N.: Non-negative matrix factorization based methods for object recognition. *Pattern Recognition Letters* 25 (2004) 893-897
11. Phillips, P.J., Wechsler, H., Huang, J., Rauss, P.J.: The FERET database and evaluation procedure for face-recognition algorithms. *Image and Vision Computing* 16 (5) (1998) 295-306
12. Wild, S., Curry, J., Dougherty, A.: Improving non-negative matrix factorizations through structured initialization. *Pattern Recognition* 37 (11) (2004) 2217-2232
13. Xu, W., Liu, X., Gong, Y. Document clustering based on non-negative matrix factorization. In: SIGIR'03, Toronto, Canada, 2003
14. Zhang, D.Q., Chen, S.C., Liu, J.: Representing image matrices: Eigenimages vs. Eigenvectors. In: ISNN, Chongqing, China, 2005
15. Zhang, D.Q., Chen, S.C.: Clustering incomplete data using kernel-based fuzzy c-means algorithm. *Neural Processing Letters* 18(3) (2003) 155-162

# Modelling Citation Networks for Improving Scientific Paper Classification Performance

Mengjie Zhang<sup>1,2</sup>, Xiaoying Gao<sup>1,2</sup>, Minh Duc Cao<sup>1</sup>, and Yuejin Ma<sup>2</sup>

<sup>1</sup> School of Mathematics, Statistics and Computer Science  
Victoria University of Wellington, P.O. Box 600, Wellington, New Zealand  
<sup>2</sup> Artificial Intelligence Research Centre, College of Mechanical and Electrical Engineering, Agricultural University of Hebei, Baoding, China  
{mengjie, xgao, minducao}@mcs.vuw.ac.nz, {zmj, myj}@hebau.edu.cn

**Abstract.** This paper describes an approach to the use of citation links to improve the scientific paper classification performance. In this approach, we develop two refinement functions, a linear label refinement (LLR) and a probabilistic label refinement (PLR), to model the citation link structures of the scientific papers for refining the class labels of the documents obtained by the content-based Naive Bayes classification method. The approach with the two new refinement models is examined and compared with the content-based Naive Bayes method on a standard paper classification data set with increasing training set sizes. The results suggest that both refinement models can significantly improve the system performance over the content-based method for all the training set sizes and that PLR is better than LLR when the training examples are sufficient.

## 1 Introduction

As more and more scientific papers are published online these days, scientific paper classification has been becoming an important research area. It is useful for researchers, librarians, publishers to search and organise papers into necessary categories according to their need.

Several scientific paper search engines such as Google Scholar and CiteSeer provide very good tools for researchers for searching for scientific papers. However, searching for papers on those engines is primarily based on keyword match and therefore, a search would often result in a large number of hits, majority of which may not be relevant to what users look for. A suggested solution to improving the search results is to categorise indexed documents into predefined topics. Users can then select the topics they are interested in when performing a paper search and the search engines need to search for documents in the specified topics only. This mechanism can not only reduce search time, but also produce a more accurate hit list.

Since the late 1990s, many machine learning approaches for general document classification have been developed. The most common methods are naive Bayes [1], k-nearest neighbours [2], decision trees [3], support vector machines [4],

maximum entropy [5], and neural networks [6]. However, these methods usually extract features from document *contents* only.

Scientific papers, different from general documents, do not exist in isolation but are linked together by a citation network. A paper normally cites other related published papers which are likely to have similar topics. Hence, in addition to the information from documents' own contents, the citation structure provides another source of clue that could be exploited for better classification.

Link analysis has been researched intensively since the birth of the world wide Web. Brin and Page exploited hypertext mining for PageRank, the technology behind the success of Google [7]. Chakrabarti et al. [8] and Oh et al. [9] combined words from connected documents into the feature vector of a document for classification. Lu and Getoor et al. [10,11,12,13] applied the relational model [14] for link mining and Craven and Slattery [15] proposed combining statistical text analysis with a relational learner such as FOIL [16]. Most of these approaches suggest that the naive use of text from neighbouring documents even degrades accuracy of the classification. One explanation is that link information is noisy and the distribution of terms from neighbouring documents is not sufficiently similar to the distribution of the "true" class. We applied a simple approach using the class information of the neighbouring documents to update the labels of the papers obtained with the contents and initial results were promising [17].

### 1.1 Goals

This paper aims to investigate an effective approach to the use of citation link structures for further improving scientific document classification performance. In this approach, we will first use the Naive Bayes method and the features extracted from document contents to determine the class categories of the documents, then use the citation link structures to refine the class labels. We will consider both a simple linear model and a complex probabilistic model. The two models will be examined and compared with the content-based Naive Bayes classifier on a standard data set with an increasing number of training examples. We will also investigate how to develop the two models, whether this approach can improve the system performance of the content-based classifier, and which model is better for scientific paper classification.

## 2 Content Based Document/Paper Classification

The problem of document classification can be stated formally as follows. Given a corpus of  $n$  documents  $\mathcal{D} = \{d_1, d_2, \dots, d_n\}$  and a set of predefined  $m$  categories or classes  $C = \{c_1, c_2, \dots, c_m\}$ , assign each document  $d_j$  to one of the classes  $c_i$ .

In the content based document classification systems, each document is converted to a numeric feature vector  $d_i = (w_1^i, w_2^i, \dots, w_l^i)$ , where  $l$  is the number of features and  $w_j^i$  is the weight of feature  $j$  for document  $i$ . The most common document representation is the *bag of words* model, where each unique word used in the corpus corresponds to a feature and the weight  $w_j^i$  reflects the number of occurrences of word  $j$  in document  $i$ .

A classifier is a function  $\phi(d_j) = (s_{j1}, s_{j2}, \dots, s_{jm})$  and  $s_{ji}$  is the category score of document  $d_j$  assigning to class  $c_i$ . The document  $d_j$  will be assigned to the category which has the highest score:

$$\mathcal{C}(d_j) = \arg \max_i \{s_{ji}\} \quad (1)$$

In this approach, we used the naive Bayes method to train the classifier. The naive Bayes approach [18] applies the Bayes theorem to estimate the probability of a class given a test document based on the assumption that words in the documents are independent given the category. The method learns the parameters of a probabilistic model from the training data.

The classifier is trained by a set of examples  $S$ . A good training set should contain sufficient examples from all categories, so that the characteristics of all categories could be extracted by the classifier trainer.

In the scientific paper domain, citations are clearly a rich source of information to identify topic of documents. There is a correlation of topics between two citing documents. It is observable that the topic of a paper is related to that of papers it links to. This suggests that, information in the citation structure could help the classification of scientific papers. The next section describes the two new models for refining the results achieved by the content based classifier.

### 3 Citation Links for Document Classification Refinement

In this approach, the naive Bayes method uses the *contents* of the documents themselves as features to predict the initial class labels of the documents. A citation link based model is then trained and used to refine the initial class labels (the results) obtained by the naive Bayes method.

The class label of each document  $d_j$  predicted by the naive Bayes classifier is a vector  $V_j = (s_{j1}, s_{j2}, \dots, s_{jm})$ , where  $s_{jk}$  is the likelihood of document  $d_j$  for class  $c_k$ . We need to design a refinement model/function  $\mathcal{F}$  to refine the label of the paper/document using the class labels of its neighbours predicted. Letting  $N_j$  be the set of documents that connect with document  $d_j$ , then we have:

$$(V_j)_{new} = \mathcal{F}(V_j, V_{j1}, V_{j2}, \dots, V_{j|N_j|}) \quad (2)$$

or

$$(V_j)_{new} = \mathcal{F}(V_j, V_k) \quad (3)$$

for refining the initial label by an individual neighbouring document  $d_k$ .

We developed two models, *linear label refinement* and *probabilistic label refinement*, to refine the class labels, which are described in the rest of this section.

#### 3.1 Linear Label Refinement (LLR)

LLR defines a linear function using the initial class labels of the neighbours of a paper to refine the initial class labels of the paper. Assuming each neighbouring document  $d_k$  has an equal influence on the document  $d_j$ , we add a fraction of

those labels into the document's own class label to form the new refinement function.

$$(V_j)_{new} = V_j + \eta \sum_{k:d_k \rightarrow d_j} V_{jk} \quad (4)$$

where the refinement rate  $\eta$  is a parameter that reflects the influence of linked papers, which is determined by an empirical search. The higher  $\eta$ , the more influence of the categories of neighbouring documents on the target document.

The refinement process is performed iteratively until the class labels of all documents in the corpus become stable.

### 3.2 Probabilistic Label Refinement (PLR)

There are several limitations of the LLR approach above. Firstly, the model is static and the inference is dependent on the parameter  $\eta$ . Deriving a good parameter value requires an empirical search and the parameter value found is unlikely to be optimal. Secondly, the model is rather “naive” and may not capture the dependencies between the current paper and the neighbours well.

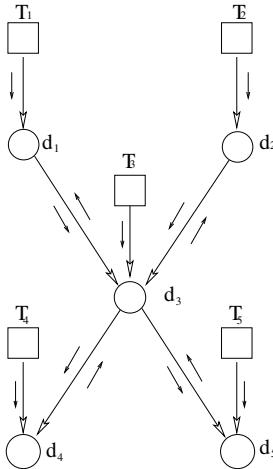
The dependencies among documents in the corpus is presented by citation links. The citation structure can be modelled as a directed graph in which vertices are documents and edges are citation links. As a paper can only cite other papers already published, there should not be a circle in the graph. The graph is similar to the Bayesian network or the belief network [19]. Therefore, we can use a method derived from the Bayesian network to model the citation link structure and call it *probabilistic label refinement*, or PLR for short.

**Modelling Citation Links.** A Bayesian network is a directed graph representing dependencies among random variables [20]. Each node in the graph represents a random variable. An edge from node  $X$  to node  $Y$  represents dependency in the form of conditional probability  $P(X|Y)$ . In this case, variable  $X$  is said to be a parent of variable  $Y$ . Each node is associated with a conditional probability distribution  $P(X|\text{parents}(X))$  that represents the effects of the parents on the node.

A Bayesian network allows calculation of the probability of a node given that some other nodes are observed. The well known inference algorithm in the Bayesian network is *probability propagation* or *message passing* [19,21,22,23].

If the graph is a poly-tree, there is at most one indirect path, and the complexity of exact inference in the Bayesian network is linear to the size of the network. However, in most cases, the network is multiply-connected and inference is NP-hard [24]. In this case, the Monte Carlo method [25] can be used to generate samples from the distributions in the network.

From the above observation, this approach can model the citation links by a variant Bayesian network. An example of the network is shown in Figure 1. There are two types of nodes in the network. Each round node represents a document, whose category is directly influenced by its text and other neighbouring documents. A square node represents the text of a document. Parent nodes of a document are its text node and other documents that are cited by the document.



**Fig. 1.** An example citation network

For an example, document  $d_3$  cites documents  $d_1$  and  $d_2$ , so its parents are its text node  $T_3$  and two document nodes  $d_1$  and  $d_2$ . Document  $d_3$  in turn, is the parent of documents  $d_4$  and  $d_5$  as it is cited by them.

In the citation network described above, the text nodes are observed. As a text node has no parents and only a child node, it sends the probability message to its child node, which is the document with the text content. Each document, upon receiving the belief probability from its text node, sends messages to its parent and child document nodes. The detail of inference in the citation network is described in next sub section.

**Inference in Citation Network.** Assume we need to find the probability of a document  $d_j$  belonging to class  $c_i$ ,  $P(d_j = c_i)$  (a short notation for  $P(\mathcal{C}(d_j) = c_i)$ ) based on available information from its text node  $T_j$  and citation information  $N_j$ . We apply the Bayesian rule for that probability as:

$$P(d_j = c_i | T_j, N_j) = \frac{P(T_j, N_j | d_j = c_i) P(d_j = c_i)}{P(T_j, N_j)} \quad (5)$$

Assuming that in a document, the content  $T$  and the neighbouring  $N$  are statistically independent, we have:

$$P(T_j, N_j | d_j = c_i) = P(T_j | d_j = c_i) P(N_j | d_j = c_i) \quad (6)$$

By cancelling out the normalisation factor  $\frac{1}{P(T_j, N_j)}$ , equation 5 becomes:

$$\begin{aligned} P(d_j = c_i | T_j, N_j) &\propto P(T_j | d_j = c_i) P(N_j | d_j = c_i) P(d_j = c_i) \\ &= P(T_j | d_j = c_i) P(d_j = c_i) P(N_j | d_j = c_i) \\ &\propto P(d_j = c_i | T_j) P(N_j | d_j = c_i) \end{aligned} \quad (7)$$

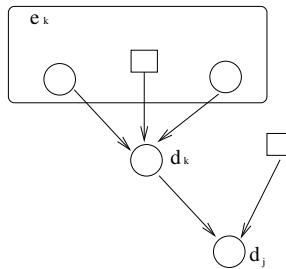
where  $P(d_j = c_i | T_j)$  is the probability of document  $d_j$  in class  $c_i$  given its content. The probability is exactly the category score computed by the content-based classifier. Therefore the improvement would be obtained by multiplying with  $P(N_j | d_j = c_i)$ . Further assuming each neighbouring document  $n_k$  is independent for  $d_j$ , we can write  $P(N_j | d_j = c_i)$  as

$$P(N_j | d_j = c_i) = \prod_{k: d_k \rightarrow d_j} P(n_k | d_j = c_i) = \prod_{k: d_k \rightarrow d_j} \frac{P(d_j = c_i | n_k) P(n_k)}{P(d_j = c_i)} = \prod_{k: d_k \rightarrow d_j} \frac{P(d_j = c_i, n_k)}{P(d_j = c_i)} \quad (8)$$

where  $d_k$  is the set of neighbouring documents of  $d_j$  and  $n_k$  is the information for  $d_j$  obtained from  $d_k$ .

The node  $d_k$  is actually not observed. In fact, it can be computed by information from its text node, which is equivalent to  $P(d_k = c_i | T_k)$ , and the information from its neighbours  $P(d_k = c_i | N_k)$ . In other words, the category information of  $d_k$  can be obtained from the content based classification and label refinement from its neighbours.  $d_k$  itself does not generate information but encapsulates all messages it receives and passes on to  $d_j$ . Suppose messages received by  $d_k$  are from some sources of evidence  $e_k$  (which are  $T_k$  and  $N_k$  as shown on Figure 2), message  $n_k$  from  $d_k$  to  $d_j$  is actually  $e_k$ . Therefore we have:

$$P(d_k = c_i | e_k) = P(d_k = c_i | T_k, N_k), \text{ and } P(d_j = c_i, n_k) = P(d_j = c_i, e_k) \quad (9)$$



**Fig. 2.** Information from  $d_k$  to  $d_j$

As can be seen from Figure 2, information from  $e_k$  is sent to  $d_j$  via  $d_k$ . Therefore, the joint probability  $P(d_j = c_i, e_k)$  can be factored as

$$\begin{aligned} P(d_j = c_i, e_k) &= \sum_l P(d_j = c_i, d_k = c_l, e_k) \\ &= \sum_l P(d_j = c_i | d_k = c_l) P(d_k = c_l | e_k) P(e_k) \\ &= \sum_l P(d_j = c_i | d_k = c_l) P(d_k = c_l | T_k, N_k) P(e_k) \end{aligned} \quad (10)$$

where  $l$  is all possible values of class  $c_l$ .

$P(e_k)$  is again the common factor for all categories  $c_i$  and thus can be normalised. Plug equation 10 into equation 8 and equation 7, we obtain:

$$\begin{aligned}
& P(d_j = c_i | T_j, N_j) \\
& \propto P(d_j = c_i | T_j) \prod_{k: d_k \rightarrow d_j} \frac{\sum_l P(d_j = c_i | d_k = c_l) P(d_k = c_l | T_k, N_k) P(e_k)}{P(d_j = c_i)} \\
& \propto P(d_j = c_i | T_j) \prod_{k: d_k \rightarrow d_j} \frac{\sum_l P(d_j = c_i | d_k = c_l) P(d_k = c_l | T_k, N_k)}{P(d_j = c_i)}
\end{aligned} \tag{11}$$

In the above formula,  $P(d_j = c_i | T_j)$  is obtained from the content based classifier.  $P(d_k = c_l | T_k, N_k)$  is the previous calculation on  $d_k$ . The term  $P(d_j = c_i | d_k = c_l)$  and  $P(d_j = c_i)$  can be learned from the training set.

**Learning the Model.** The independence assumption we made above makes learning the network easier. Instead of learning the full conditional probability distribution  $P(d_j | \text{parent}(d_j))$ , the model needs only to learn the distribution  $P(d_j = c_i | d_k = c_l)$  that can be applied to all the citation dependencies.

As a citation link can be either in-link (cited) or out-link (citing), we need to estimate two conditional probability distributions, one for the case  $d_j$  cites  $d_k$  and the other for the case  $d_j$  is cited by  $d_k$ . Each distribution is a matrix  $|C| \times |C|$ . The entry  $(i, l)$  of the in-link matrix is estimated from the training set by the maximum likelihood:

$$\text{in-link}_{il} = P(d_j = c_i | d_k = c_l) = \frac{L_{li}}{\sum_k L_{lk}} \tag{12}$$

where  $L_{lk}$  is the number of citation links from a document in class  $c_l$  to  $c_k$ . Similarly, the matrix out-link is also estimated as:

$$\text{out-link}_{il} = P(d_j = c_i | d_k = c_l) = \frac{L_{il}}{\sum_k L_{kl}} \tag{13}$$

Finally, the prior probability is estimated as:

$$P(d_j = c_i) = \frac{|S_i|}{|S|} \tag{14}$$

where  $S$  is the examples in the training set and  $S_i$  is the examples for class  $c_i$ .

## 4 Experiments and Results

### 4.1 Experiment Design and Configuration

We used a subset of Cora, a real world scientific paper corpus [26] as test bed in the experiments. The test bed contains 3098 papers in seven subjects of machine learning: *case based, probabilistic methods, learning theory, genetic algorithms, reinforcement learning, neural networks, and rule learning*. The proportion of papers in each topic ranges from 7% in the rule learning topic to 32% in the neural network topic. These papers are connected by 11713 citation links. 76% of citations are between papers of same class while 24% are cross topics.

For each experiment, we randomly select an equal portion of the papers from each category to train a classifier, and use the rest as the test set to evaluate the

classifier. We carry out experiments on the cora by varying training set sizes, 20%, 30%, 40%, 50% and 60% of the cora, to examine the effect of each method. For the ease of notation, we name each configuration of the data set as CORAXX where XX is the percentage of training set. For example, the set CORA40 refers to the use of 40% of the collection for training and 60% for testing.

The training set is first used to train the naive Bayes content-based classifier, and then used to learn the Bayesian citation network. For testing, we first apply the content based naive Bayes classifier to compute the initial class labels of the documents, then use each of two models to refine the classification results.

Each experiment is run 10 times. Note that the training set size for the 10 runs is the same but the actual documents in the training set are different. For example, for each run using the set CORA20, the system randomly selects 20% of the documents in each class to make up the training set.

## 4.2 Results

Table 1 shows the average classification accuracy on the test sets over 10 runs of the naive Bayes content based method, and the new approach with the LLR and the PLR models using different sizes of the training set.

**Table 1.** Results (%) of the PLR, LLR and the content based classification

Method \ Data set	CORA20	CORA30	CORA40	CORA50	CORA60
Content Based Method	67.79	74.57	77.25	83.35	85.05
Content+citation1 (LLR)	84.40	85.39	85.96	86.51	86.75
Content+citation2 (PLR)	78.86	82.95	85.43	87.82	88.08

As can be seen from table 1, the systems with both the LLR and the PLR models achieved significantly better classification results than the basic content based Naive Bayes method for all the data sets, suggesting that the citation link refinement for classification does improve the system performance and that the citation link information is important for scientific paper classification.

As expected, for all the three methods, as the training set size increased, the system performance also got improved. However, the improvements over different training set sizes were quite different.

When there is only a small set of training examples such as CORA20 and CORA30, LLR outperformed PLR. As the training set increased, the PLR method showed a clear advantage over LLR. This is mainly due to the nature of the two methods: the LLR is only an approximated modelling of the citation structure, but the PLR method can learn high level hidden patterns between the citation links and the contents of the scientific papers when the training examples are sufficient. Because of this nature, the PLR spent a bit longer time than the LLR method for the training process.

Accordingly, whenever there are sufficient scientific papers/documents that can be used for training, the PLR method is recommended. Only when is there very few existing documents for training, LLR is a good choice.

## 5 Conclusions

The goal of this paper was to investigate a new approach to the use of the citation links to refine the initial results obtained from the content-based classifier. The goal was successfully achieved by developing two refinement models, the linear label refinement (LLR) and the probabilistic label refinement (PLR), to refine the class labels of the documents obtained by the content-based naive Bayes classification method. The system with the two refinement models was examined and compared with the content based Naive Bayes method on a standard paper classification data set with increasing training set sizes. The results suggest that both refinement models significantly improved the system performance.

The LLR method refined the initial label of a document by adding the labelling information of its neighbours multiplied by a constant. The PLU applied a variant Bayesian network to model the citation structures and used a message passing algorithm for propagating class label information around the citation network. Our experiments suggest that whenever there are sufficient scientific papers/documents that can be used for training, the PLR method is recommended. Only when are there very few existing documents for training, LLR will be a good choice.

While PLR has made good improvement over the other methods, we believe that it can be further improved by embedding some exact inference algorithms such as junction tree or clustering. We will also investigate some temporal approaches such as hidden Markov models and dynamic Bayesian networks for modelling the citation structure in the future.

## References

1. Lewis, D.D.: Naive (bayes) at forty: The independence assumption in information retrieval. In Proceedings of the 10th European Conference on Machine Learning, Chemnitz, DE, Springer Verlag, Heidelberg, DE (1998) 4–15
2. Yang, Y.: Expert network: effective and efficient learning from human decisions in text categorisation and retrieval. In Proceedings of the 17th ACM International Conference on Research and Development in Information Retrieval, Dublin, IE, Springer Verlag (1994) 13–22
3. Lewis, D.D., Ringuelette, M.: A comparison of two learning algorithms for text categorization. In: Proceedings of the 3rd Annual Symposium on Document Analysis and Information Retrieval, Las Vegas, US (1994) 81–93
4. Joachims, T.: Text categorization with support vector machines: learning with many relevant features. In Proceedings of the 10th European Conference on Machine Learning, Springer, (1998) 137–142
5. Nigam, K., Lafferty, J., McCallum, A.: Using Maximum Entropy for Text Classification. In: IJCAI-99 Workshop on Machine Learning for Information Filtering. (1999) 61–67
6. Ruiz, M.E., Srinivasan, P.: Hierarchical neural networks for text categorization. In Proceedings of the 22nd ACM International Conference on Research and Development in Information Retrieval, ACM Press (1999) 281–282
7. Brin, S., Page, L.: The anatomy of a Large-scale Hypertextual Web search Engine. Computer Networks and ISDN Systems **30**(1–7) (1998) 107–117

8. Chakrabarti, S., Dom, B.E., Indyk, P.: Enhanced hypertext categorization using hyperlinks. In Proceedings of the ACM International Conference on Management of Data, ACM Press (1998) 307–318
9. Oh, H.J., Myaeng, S.H., Lee, M.H.: A practical hypertext categorization method using links and incrementally available class information. In Proceedings of the ACM International Conference on Research and Development in Information Retrieval, ACM Press (2000) 264–271
10. Getoor, L., Segal, E., Taskar, B., Koller, D.: Probabilistic Models of Text and Link Structure for Hypertext Classification. In IJCAI Workshop on Text Learning: Beyond Supervision (2001)
11. Lu, Q., Getoor, L.: Link-based classification using labeled and unlabeled data. In: ICML Workshop on the Continuum from Labeled to Unlabeled Data in Machine Learning and Data Mining, Washington DC. (2003)
12. Lu, Q., Getoor, L.: Link-based text classification. In: IJCAI Workshop on Text Mining and Link Analysis. Acapulco, MX. (2003)
13. Lu, Q., Getoor, L.: Link-based classification. In: International Conference on Machine Learning, Washington, DC. (2003)
14. Taskar, B., Segal, E., Koller, D.: Probabilistic Classification and Clustering in Relational Data. In Proceeding of the 17th International Joint Conference on Artificial Intelligence (2001) 870–878
15. Craven, M., Slattery, S.: Relational Learning with Statistical Predicate Invention: Better Models for Hypertext. *Mach. Learn.* **43**(1-2) (2001) 97–119
16. Quinlan, J.R.: Learning logical definitions from relations. *Mach. Learn.* **5**(3) (1990) 239–266
17. Cao, M.D., Gao, X.: Combining contents and citations for scientific document classification. In: Proceedings of 18th Australian Joint Conference on Artificial Intelligence, Springer (2005) 143–152
18. Lewis, D.D.: Naive (Bayes) at forty: The independence assumption in information retrieval. In: Proceedings of the 10th European Conference on Machine Learning. Springer, (1998) 4–15
19. Pearl, J.: Probabilistic Reasoning in Intelligent Systems: Network of Plausible Inference. Morgan Kaufmann publishers (1988)
20. Ghahramani, Z.: Graphical Models: Parameter Learning. In Handbook of Brain Theory and Neural Networks. 2 edn. MIT Press (2003) 486–490
21. Jordan, M.I., Weiss, Y.: Graphical Models: Probabilistic Inference. In Handbook of Brain Theory and Neural Networks. 2 edn. MIT Press (2003) 490–496
22. Yedidia, J.S., Freeman, W.T., Weiss, Y.: Understanding Belief Propagation and its Generalizations. Technical Report TR-2001-22, Mitsubishi Electric Research Laboratories, Inc. (2002)
23. Cowell, R.: Introduction in Inference in Bayesian Networks. MIT Press, Cambridge, MA, USA (1999) 9–26
24. Russell, S., Novig, P.: Artificial Intelligence: A Modern Approach. 2 edn. Prentice Hall (2005)
25. MacKay, D.J.C.: Introduction to Monte Carlo Methods. In: Learning in graphical models. MIT Press, Cambridge, MA, USA (1999) 175–204
26. McCallum, A.K., Nigam, K., Rennie, J., Seymore, K.: Automating the Construction of Internet Portals with Machine Learning. *Information Retrieval* **3**(2) (2000) 127–163

# Analysis on Classification Performance of Rough Set Based Reducts

Qinghua Hu, Xiaodong Li, and Daren Yu

Harbin Institute of Technology, Harbin 150001, P.R. China  
huqinghua@hcms.hit.edu.cn

**Abstract.** Feature subset selection and data reduction is a fundamental and most explored area in machine learning and data mining. Rough set theory has been witnessed great success in attribute reduction. A series of reduction algorithms were constructed for all kinds of applications based on rough set models. There is usually more than one reduct for some real world data sets. It is not very clear which one or which subset of the reducts should be selected for learning. Neither experimental comparison nor theoretic analysis was reported so far. In this paper, we will review the proposed attribute reduction algorithms and reduction selection strategies. Then a series of numeric experiments are presented. The results show that, statistically speaking, the classification systems trained with the reduct with the least features get the best generalization power in terms of single classifiers. Furthermore, Good performance is observed from combining the classifiers constructed with multiple reducts compared with Bagging and random subspace ensembles.

## 1 Introduction

Rough set methodology has been witnessed great success in attribute reduction, rule extraction, granular computing, and reasoning with imprecise and vague information in recent years. A great number of researchers from mathematics, artificial intelligence and engineering societies are evolved with the theoretic and application investigation. Since Pawlak introduced the concepts of lower approximation, upper approximation, rough set based on indiscernibility relations, a series of non-standard rough set models were introduced based on similarity relation [1, 2], preference relation [3], neighborhood relation [4, 5] and general relation [6]. From the other aspect, fuzzy mathematics, probability and statistics were introduced into intelligent computation with rough set techniques. To deal with fuzziness in data, some generalizations of Pawlak's model were proposed; the theories on rough and fuzzy sets were put together. Rough-fuzzy sets and fuzzy-rough sets were introduced in [7, 8] and analyzed in detail [9-11]. Statistics tools were introduced into rough set based data analysis in [12], Pawlak explored the relation between rough set data analysis and Bayes reasoning and give a new view of Bayes reasoning [13]. Wu and Leung presented the connections between rough set theory and Dempster-Shafer theory of evidence. Wei and Zhang extended the variable precision rough set model [14] into probabilistic rough set model [15]. Furthermore, in [16] Hu and Yu et al gave a fuzzy probabilistic rough set model, where a probability distribution is assigned on the

fuzzy sets, which are used to approximate the fuzzy concepts in approximation spaces. After more than 20 year development, the foundation of rough set theory is becoming firmer and firmer, and its applications have grown in number and variety. It has become one of the most important tools for modeling and reasoning with imperfect information.

Attribute selection and data reduction is one of the main applications of rough set theory. In the view of rough sets, reducts are the attribute sets which keep the representation power of the original data and have no information redundancy. Given a data set with  $N$  attributes, there are  $2^N$  subsets of the attribute set. Some of the  $2^N$  subsets satisfy the conditions of reducts. Rough set theory tells us there are a number of reducts for some data set, however it doesn't present methods for finding the best or all reducts. In fact, it is not an easy thing to get it. Exhaustive search of reducts in the attribute space is infeasible even the data is with a moderate size of attributes.

Reduction is the process to find one or several reducts in the attribute subsets. As there are many irrelevant and redundant features in the real-world databases, which will decrease the classification performance of a pattern recognition system. Therefore, attribute selection and data reduction plays an important role in machine learning and pattern recognition. A series of rough set based feature selection and reduction algorithms are developed. In order to find a more efficient reduction algorithm, Skowron introduced the discernibility function and discernibility matrix [17], the elements in the matrix is the set of attributes which can discern the two objects. Skowron further developed a reduction with the Boolean function in the conjunctive form based on the distribution law and the absorptive law. Hu and Cercone introduced the ratio of the positive region with respect to the selected attributes to that with respect to the whole attributes as the significance of the attributes [18]. Based on the measure of significance, a greedy algorithm was constructed to find a reduct from the information system. Different from Hu's measure, Jelonek et al used the ratio of the positive region with respect to the selected attributes to the universe as the significance of the attributes [19]. In fact, there is no difference between the two algorithms substantially. Wang and Miao introduce the information gain as the significance measure of attributes [20]. In order to improve the stability of reduction algorithm, Bazan et al presented a dynamic reduct technique for extracting laws from decision tables [21]. Wroblewski introduced genetic algorithms to find the minimal reducts [22].

In order to deal with numeric attribute and fuzzy attributes, the fuzzy version of reduction algorithm was proposed in [23] and improved in [24, 25]. In this approach, the cardinality of positive regions was extended into the fuzzy case based on the extension law. Then the significance of attributes is defined as [19]. Bhatt found the algorithm proposed by Shen and Jensen is not convergent in a number of real-world databases [26], therefore an improved approach was presented. Hu and Yu presented an information measure for fuzzy indiscernibility relations [27] and used it to compute the significance of attributes [28]. A greedy algorithm was constructed to find minimal reducts from hybrid data.

As we know, there is usually more than one minimal reduct for some real-world databases. In particular, tens of, even hundreds of reducts are found for some data with many candidate attributes. As a given machine learning task, which reduct

should be selected? According to Occam's razor principle, the minimal is preferred. However, some literatures show the principle doesn't work in some cases. To our knowledge, there has been neither theoretic result nor empirical comparison so far. In this paper, we will review the reduction algorithms and conduct a detailed experiment analysis of classification performance based on rough set reducts. The conclusion from the experiments will form a guideline for reduct selection in real-world applications.

The rest of the paper is structured as follows. Section 2 will present the basic notions on rough set theory. A series of rough set based reductions are given in section 3. Then section 4 presents the experimental results and analysis. The conclusion and future work are shown in section 5.

## 2 Notions on Rough Sets

Rough set theory, which was introduced to deal with imperfect and vague concepts by Pawlak [29], has extracted a lot of attention from theory and application research areas. Data sets are usually given as the form of tables, we call a data table as an information system, formulated as a four-tuple  $IS = \langle U, A, V, f \rangle$ , where  $U = \{x_1, x_2, \dots, x_n\}$  is a set of finite and nonempty objects, called the universe,  $A$  is the set of attributes characterizing the objects,  $V$  is the domain of attribute value and  $f$  is the information function  $f : U \times A \rightarrow V$ . If the attribute set is divided into condition attributes  $C$  and decision attributes  $D$ , the information system is also called a decision table.

With arbitrary attributes  $B \subseteq A$ , there is an indiscernibility relation  $IND(B)$ :

$$IND(B) = \{ \langle x, y \rangle \in U \times U \mid \forall a \in B, a(x) = a(y) \}.$$

$\langle x, y \rangle \in IND(B)$  means objects  $x$  and  $y$  are indiscernible with respect to attributes  $B$ . Obviously, indiscernibility relation is an equivalent relation which satisfies reflexivity, symmetry and transitivity. The equivalence class induced by the attributes  $B$  is denoted by

$$[x_i]_B = \{x \mid \langle x, x \rangle \in IND(B), y \in U\}.$$

Equivalence classes generated by  $B$  are also called  $B$ -elemental granules,  $B$ -information granules. The set of elemental granules forms a concept system, which is used to characterize the imperfect concepts in the information system. The partition of the universe by the attributes  $B$  is defined as

$$U / B = \{[x_i]_B \mid x_i \in U\}$$

Given arbitrary concept  $X$  in the information system, two unions of elemental granules are associated with

$$\begin{cases} \underline{BX} = \{[x]_B \mid [x]_B \subseteq X, x \in U\} \\ \overline{BX} = \{[x]_B \mid [x]_B \cap X \neq \emptyset, x \in U\} \end{cases}.$$

The concept  $X$  is approximated by the two sets of elemental granules.  $\underline{BX}$  and  $\overline{BX}$  are called lower and upper approximations of  $X$  in terms of attributes  $B$ .  $\underline{BX}$  is also called the positive.  $X$  is a definable set if  $\underline{BX} = \overline{BX}$ , which means the concept  $X$  can be perfectly characterized with the knowledge  $B$ , otherwise,  $X$  is indefinable. An indefinable set is called a rough set.  $BND(X) = \overline{BX} - \underline{BX}$  is called the boundary of the approximations. As a definable set, the boundary is empty. In this context, attributes are the knowledge to discern the objects and form concept systems for classification and reasoning.

$C$  is the condition attribute set and  $D$  is the decision in a given nonempty and finite universe  $U$ , then  $C$  and  $D$  will generate two partitions of the universe. Machine learning is usually involved to using condition knowledge to approximate the decision and finding the mapping from the conditions to decisions. Approximating  $U / D$  with  $U / C$ , the positive and boundary regions are defined as:

$$\begin{aligned} POS_C(D) &= \bigcup_{X \in U / D} \underline{CX}, \\ BND_C(D) &= \bigcup_{X \in U / D} \overline{CX} - \bigcup_{X \in U / D} \underline{CX}. \end{aligned}$$

The boundary region is the set of elemental granules which can not be perfectly described by the knowledge  $C$  and the positive region is the set of  $C$ -elemental granules which completely belong to one of the decision concepts. The size of positive or boundary regions reflects the approximation power of the condition attributes. Given a decision table, for any  $B \subseteq C$ , it is said the decision attribute set  $D$  depends on the condition attributes with the degree  $k$ , denoted as  $B \Rightarrow_k D$ , where

$$k = \gamma_B(D) = \frac{|POS_B(D)|}{|U|}.$$

The dependence coefficient  $k$  measures the approximation power of a condition attribute set with respect to the decision  $D$ . In data mining, especially in feature selection, it is important to find the dependence relations between attribute sets and to find a concise and efficient representation of the data.

Given a decision table  $DT = \langle U, C \cup D, V, f \rangle$ , if  $P \subseteq Q \subseteq C$ , we have

$$\gamma_Q(D) \geq \gamma_P(D)$$

Given a decision table  $DT = \langle U, C \cup D, V, f \rangle$ ,  $B \subseteq C$ ,  $a \in B$ , we say condition attribute  $a$  is indispensable if  $\gamma_{(B-a)}(D) < \gamma_B(D)$ , otherwise we say  $a$  is redundant. We say  $B \subseteq C$  is independent if any  $a$  in  $B$  is indispensable. Attribute subset  $B$  is a reduct of the decision table if

- 1)  $\gamma_B(D) = \gamma_C(D)$ ;
- 2)  $\forall a \in B : \gamma_B(D) > \gamma_{B-a}(D)$ .

A reduct of a decision table is the condition attribute subset which keeps the approximation power of the condition attributes and has no redundant attribute. The

definition of “reduct” is perfect for feature selection, where most of the aims are to find a most compact representation of the problem. “Reduct” present a concise and profound definition for feature subset selection.

$\{B_i \mid i = 1, \dots, N\}$  is the set of reducts, then the core of the information system is defined as

$$\text{Core} = \bigcap_{i=1}^N B_i .$$

### 3 Reduct Selection Strategies

It is quite often that an information system or a decision table has more than one reduct. Sometimes tens, even hundreds of reducts are produced. Each reduct can be used instead of the whole set of condition attributes in the original system in learning and decision making procedure because the dependency relation between condition and decision don't vary. So a natural question is the best one for a given learning and classification task. There are some proposed principles for reduct selection strategy [30, 31]. Roughly speaking, they can be divided into two kinds. One is based on a single reduct, and the other is combining multiple reducts.

As to a single reduct strategy, there are also some different ideas. The selection depends on the optimization criterion and prior knowledge at hand.

**Random selection criterion:** Shen and Jensen in [24, 25, 32] considered since any reduct in the reduct set has the same representation power as the original decision table, randomly selection one of the reducts. Thus a single minimal subset of the attributes is all that is required for data reduction. Hassanien takes the same argument in analyzing medical data and generating classification rules from a set of observed samples of breast cancer data. [33].

**Cost minimizing criterion:** As to a real-world application, the cost to acquire the attributes are different. For example, in the medical and fault recognition of machinery domains, some diagnostic procedures are much more expensive than others. If a cheap procedure has the same recognition accuracy as the expensive one, one will select the reduct with the minimal cost for considerable saving will be accomplished without decreasing the quality of the diagnosis [30].

**Occam's Razor criterion:** Occam's razor is often considered one of the fundamental tenets of modern science. it states that “Entities should not be multiplied beyond necessity”. It was formulated by William of Occam in the late Middle Ages as a criticism of scholastic philosophy, whose theories grew ever more elaborate without any corresponding improvement in predictive power. Today it is often invoked by learning theorists and KDD practitioners as a justification for preferring simpler models over more complex ones [34]. Within KDD Occam's razor is often used in a sense that can be stated as: Given two models with the same training-set error, the simpler one should be preferred because it is likely to have lower generalization error. In this meaning, the reducts with the least attributes are preferred. Based on this criterion, a procedure to find the minimal reduct should be constructed. Unfortunately, in [35], Skowron showed that finding all of the reducts or the minimal

reduct for an information system is a NP-complete problem. In order to deal with the problem, a series of greedy algorithms have been proposed based on some significance measures of attributes, such as dependence function, mutual information and so on. These algorithms cannot guarantee the reducts found are the minimal and also can guarantee the found attribute subset is a reduct of the information system. However, they will find an approximately minimal reduct.

There are also some researchers constructing classifier system based on multiple reducts, instead of a single reduct. Namely, they train a series of classifiers with a set of reducts, and then combine the decisions from the classifier set with a fusion algorithm in the predicting stage.

In [36], Hu introduced two algorithms to search multiple reducts from an information system, and then generate a set of maximal generalization rules for each reduct. The predictions from all of the trained classifiers are combined with simple voting strategy. Good performance was observed from a series of numeric experiments with some UCI machine learning data sets.

Wu et al constructed Bayes net ensemble with multiple reducts [37]. Hu proposed a new idea to combine multiple reducts—selective ensemble [38], where a part of reducts are selected to construct classifiers, instead of using all the reducts. They took decision tree as the learning machine, and combined the trained multiple decision trees with plurality voting method. Experiments showed the classification performance is greatly improved.

## 4 Empirical Analysis

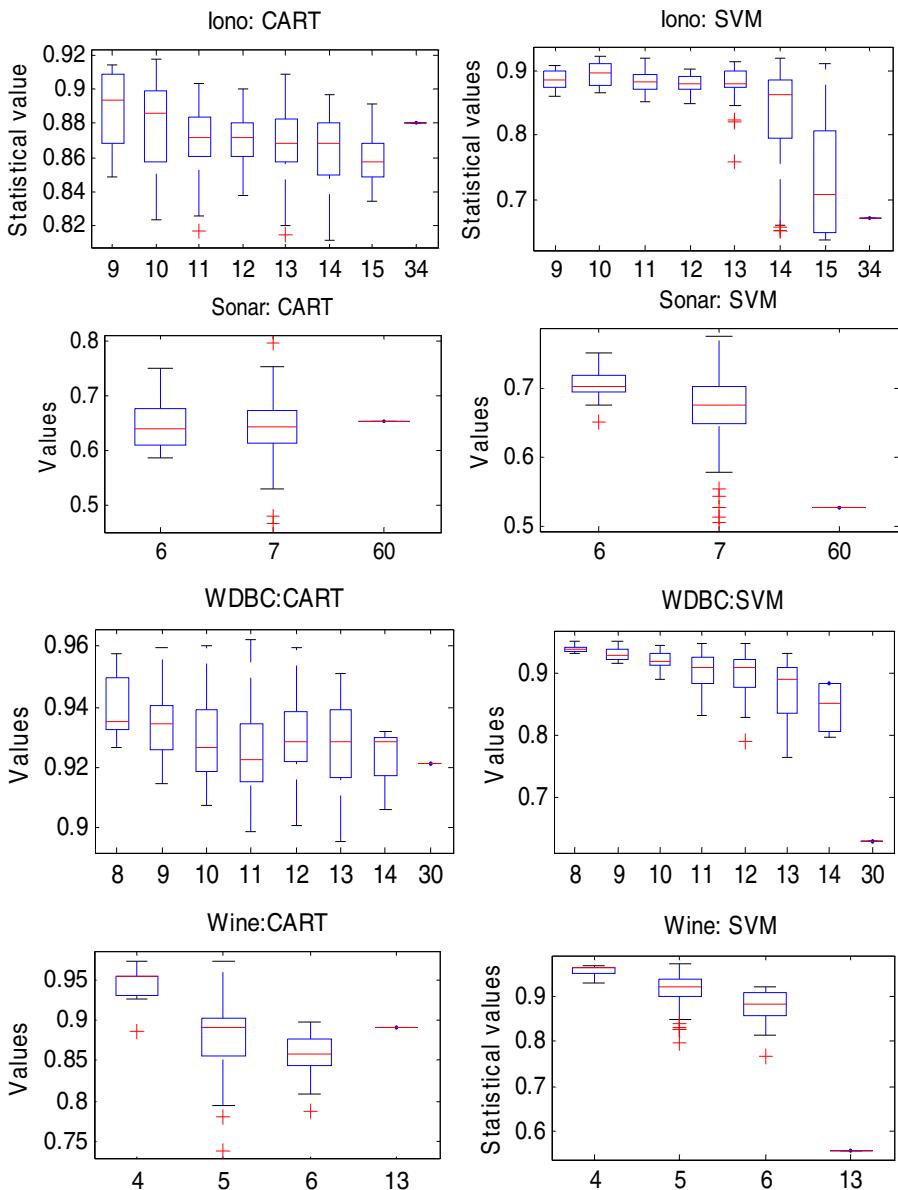
The aims of the numeric experiments is not to compare the performance of different reduction algorithms, but to compare the classification performance of all the reducts, then try to find some guidelines for selecting reducts in learning and classification.

Four UCI machine learning data sets are selected and two popular learning algorithms, CART and SVM, are introduced in the experiments. The descriptions of the data are presented in table 1. There are some numeric attributes in the data. We discretize numeric attributes with FCM. When conducting multiple-reduct algorithm, 195, 248, 212 and 136 reducts are found from the four data sets, respectively.

We train two classifiers (a decision tree and a SVM) with each reduct. So a series of classifiers can be constructed. As to CART algorithm, Gini's diversity index is chose as the split criterion; as to SVM, linear SVM is employed. We compute the average classification accuracy with 10-fold cross validation to evaluate performances of reducts. The statistical results are shown with the box plots in figure 1. The x-axis is the number of attributes in the reducts.

**Table 1.** Data description

	Data	Sample size	Features	Reducts
1	Iono	351	34	195
2	Sonar	208	60	248
3	WDBC	569	30	212
4	Wine	118	13	136

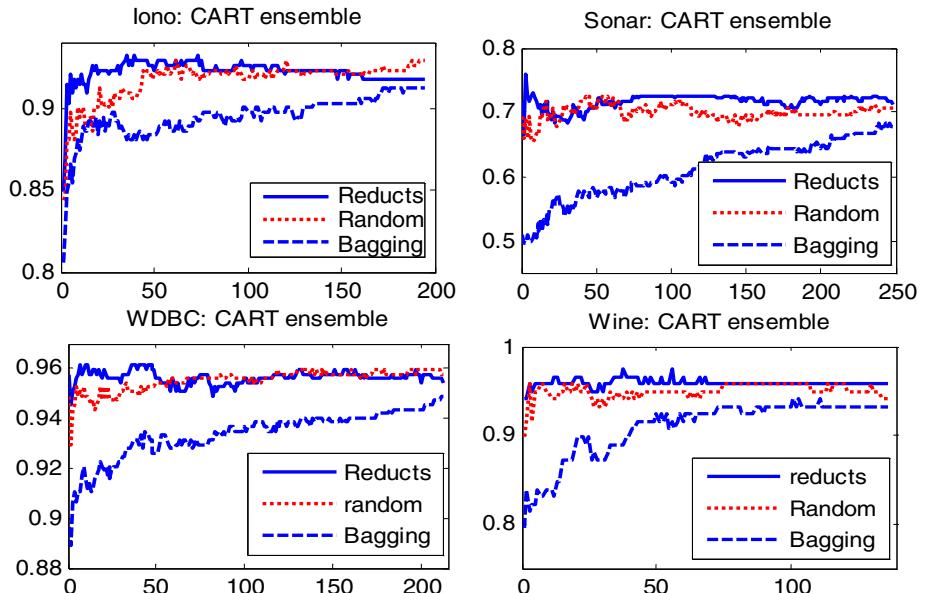


**Fig. 1.** Statistical Box-plot of accuracies of four data sets with CART and SVM, respectively

Note that the last lines in the box plots are the performance accuracies of the original data. From the above plots, it's easy to find that the classification performance of original data is comparative with the reducts for CART algorithm. However, the performance is greatly improved by reduction as to SVM algorithm. It shows that CART algorithm is not sensitive to attribute reduction. Statistically speaking, the reducts with fewer features get high classification accuracies both for

CART and SVM. However, some reducts with more features get good performances and some reducts with fewer features get bad performances. The experimental results tell us it is not absolutely credible in depending on a single reduct.

Hu [37] and Wu [38] suggested that combining a set of classifier trained with multiple reducts will improve the classification performance. Here we will show some experimental results in figure 2. The x-axis is the number of classifiers in the ensemble systems; y-axis is the average classification accuracies of the ensemble systems. The curve marked reducts is the accuracies of the ensemble systems which base classifiers are trained with rough set based reducts; the curve marked random is the accuracies of ensemble systems with the random subspace method [39], and the curve marked bagging is the accuracies of ensemble systems with Bagging.



**Fig. 2.** The classification accuracies of three ensembles vary with number of fused classifiers

**Table 2.** Classification accuracies with different ensemble systems

	Raw data	Reduced ensemble	Random ensemble	Bagging
Iono	88.04%	93.16%	92.88%	91.17%
Sonar	65.50%	75.96%	72.60%	67.79%
WDBC	92.09%	96.13%	95.96%	94.90%
Wine	89.09%	97.46%	95.76%	94.92%
<b>Average</b>	<b>83.68%</b>	<b>90.68%</b>	<b>89.30%</b>	<b>87.20%</b>

We can find the reduct ensemble get good performance comparative with random subspace method and Bagging in most of cases. And it is also easy to see that the accuracies of the ensemble systems firstly increase with the number of the classifiers, after a peak, and then the accuracies are reducing. It shows that the classification

accuracies don't monotonously increase with the number of base classifiers. A stop criterion should be constructed. Therefore a selective ensemble is more advisable.

Table 2 shows the maximal classification accuracies with different classification systems. Reduct ensembles on the data sets consistently get the best performance.

## 5 Conclusion

Rough set based attribute reduction is widely analyzed and a series of reduction algorithms and software were developed. However, as a lot of reducts can be found for some real-world data sets, there is no general criterion for selecting reducts to constructs classifiers. In this paper, after reviewing the existing selection methods and reduction algorithms, we conduct a set of numeric experiments with UCI machine learning data sets. The experimental results show that although a reduct with fewer attributes cannot get absolutely good classification performance compared with the reducts with more attributes, statistically speaking, the reducts with fewer attributes are better than the reducts with more attributes. Therefore, Occam's razor principle works in the view of statistics; combining multiple reducts can improve the classification performance, what's more, a selective ensemble is more advisable.

## References

1. Kim D., Bang S. Y.: A handwritten numeral character classification using tolerant rough set. *IEEE transactions on PAMI*. 22 (2000) 923-937
2. Kim D.: Data classification based on tolerant rough set. *Pattern recognition*. 34 (2001) 1613-1624
3. Greco S., Matarazzo B., Slowinski R.: Rough sets theory for multicriteria decision analysis. *European journal of operational research*. 129 (2001) 1-47
4. Yao Y.: Relational interpretations of neighborhood operators and rough set approximation operators. *Information sciences*. 111 (1998) 239-259
5. Wu W. Z., Zhang W. X.: Neighborhood operator systems and approximations. *Information sciences*. 144 (2002) 201-217
6. Slowinski R., Vanderpooten D.: A generalized definition of rough approximations based on similarity. *IEEE transactions on knowledge and data engineering*. 12 (2000) 331-336
7. Dubois D., Prade H.: Rough fuzzy sets and fuzzy rough sets. *International Journal of general systems*. 17 (1990) 191-209
8. Dubois D., Prade H.: Putting fuzzy sets and rough sets together, in: R. Slowinski (Ed.), *Intelligent Decision support*, Kluwer Academic, Dordrecht, 1992, 203-232
9. Boixader D., Jacas J., Recasens J.: Upper and lower approximations of fuzzy sets. *International journal of general systems*. 29 (2000) 555-568
10. Wu W., Mi J., Zhang W.: Generalized fuzzy rough sets. *Information sciences*. 151 (2003) 263-282
11. Yeung D. S., Chen D. G., Tsang ECC, et al.: On the generalization of fuzzy rough sets. *IEEE transactions on fuzzy systems*. 13 (2005) 343-361
12. Duntsch I., Gediga G.: Uncertainty measures of rough set prediction. *Artificial intelligence*. 106 (1998) 109-137
13. Pawlak Z.: Rough sets, decision algorithms and Bayes' theorem. *European Journal of Operational Research*. 136 (2002) 181-189

14. Ziarko W.: Variable Precision Rough Set Model. *J. Computer and System Sciences.* 46 (1993) 39-59
15. Wei L., Zhang W.: Probabilistic rough sets characterized by fuzzy sets. *International journal of uncertainty, fuzziness and knowledge based systems.* 12 (2004) 47-60
16. Hu Q., Yu D., and Xie Z.: Fuzzy probabilistic approximation spaces and their information measures. *IEEE transactions on fuzzy systems.* 14 (2006) 191-201
17. Skowron R., Rauszer C., 1991. The discernibility matrices and functions in information systems. *Intelligent decision Support: Handbook of Applications and Advances of Rough Set Theory,* 331-362
18. Hu X., Cercone N.: Learning in Relational Databases: A Rough Set Approach. *Computational Intelligence.* 11(1995) 323-338
19. Jelonek J., Krawiec K., Slowinski R.: Rough set reduction of attributes and their domains for neural networks. *Computational Intelligence.* 11 (1995) 339-347
20. Wang J., Miao D.: Analysis of attribute reduction strategies of rough set. *Journal of computer science and technology.* 13 (1998) 189-193
21. Bazan J. G., Skowron A., and Synak P.: Dynamic reducts as a tool for extracting laws from decision tables. *LNAI, vol. 869, pp.346-355.* Springer-Verlag, 1994
22. Wroblewski J.: Finding minimal reducts using genetic algorithms. In Proc. Second International Joint Conference on Information Sciences, pages 186-189, Sept. 1995
23. Jensen R., Shen Q.: Fuzzy-rough sets for descriptive dimensionality reduction. *FUZZ-IEEE'02. Vol.1, 12-17, pp: 29 -34*
24. Jensen R., Shen Q.: Fuzzy-rough attribute reduction with application to web categorization. *Fuzzy sets and systems.* 141 (2004) 469-485
25. Shen Q., Jensen R.: Selecting informative features with fuzzy-rough sets and its application for complex systems monitoring. *Pattern recognition.* 37 (2004) 1351-1363
26. Bhatt R. B., Gopal M.: On fuzzy-rough sets approach to feature selection. *Pattern recognition letters.* 26 (2005) 965-975
27. Hu Q. H., Yu D. R.: Entropies of fuzzy indiscernibility relation and its operations. *International Journal of uncertainty, fuzziness and knowledge-based systems.* 12 (2004) 575-589
28. Hu Q. H., Yu D. R., Xie Z. X.: Information-preserving hybrid data reduction based on fuzzy rough techniques. *Pattern recognition letters.* 27 (2006) 414-423
29. Pawlak Z.: *Rough Sets—Theoretical Aspects of Reasoning about Data.* Kluwer Academic, Dordrecht. 1991
30. Hu X.: 1995, knowledge discovery in database: an attribute-oriented rough set method. Ph.D. Thesis. University of Regina.
31. Slezak D.: 2001, Approximate decision reducts. Ph.D. Thesis. Warsaw University.
32. Jensen R., Shen Q.: Semantics-preserving dimensionality reduction: rough and fuzzy-rough based approaches. *IEEE trans. on knowl. and data engin.* 16 (2004) 1457-1471
33. Hassanien E.: Rough Set Approach for Attribute Reduction and Rule Generation: A Case of Patients With Suspected Breast Cancer. *Journal of the American society for information science and technology.* 55 (2004) 954-962
34. Domingos P.: The role of Occam's razor in knowledge discovery. *Data mining and knowledge discovery.* 3 (1999) 409-425
35. Skowron R., Rauszer C.: The discernibility matrices and functions in information systems. *Intelligent decision support—Handbook of applications and advances of the rough sets theory.* R. Slowinski (edit) 1991, 331-362.
36. Hu X.: Using rough set theory and database operations to construct a good ensemble of classifiers for data mining applications. 2001 ICDM. 233-240.

37. Wu Q., David B., Martin M.: Multiknowledge for decision making. *Knowledge and information systems.* 7 (2005) 246-266
38. Hu Q. H., Yu D. R., Wang M. Y.: Constructing rough decision forests. D. Slezak et al. (Eds.): RSFDGrC 2005, LNAI. 3642 (2005) 147-156
39. Ho T. K.: The Random Subspace Method for Constructing Decision Forests. *IEEE Transactions on pattern analysis and machine intelligence.* 20 (1998) 832-844

# Parameter Optimization of Kernel-Based One-Class Classifier on Imbalance Text Learning

Ling Zhuang and Honghua Dai

School of Engineering and Information Technology, Deakin University,  
221 Burwood Highway, VIC 3125, Australia  
[lzhu@deakin.edu.au](mailto:lzhu@deakin.edu.au), [hdai@deakin.edu.au](mailto:hdai@deakin.edu.au)

**Abstract.** Applying one-class classification to the minorities in an imbalance data has been shown to have the potential to achieve better performance than conventional learning schemes. Parameter optimization is a significant issue when the one-class classifier is sensitive to the parameters. For one-class learning scheme with the kernel function as one-class SVM and SVDD, besides the parameters involved in the kernel, the rejection rate is another one-class specific parameter. In this paper, we proposed an improved framework in which the minority target class is used first for learning in the classification stage; then both minority and majority class are employed for estimating the generalization performance. This performance is set as the optimization criteria. Experiments on UCI and Reuters text data show that both of the parameter optimized one-class classifiers outperform the other standard one-class learning schemes.

## 1 Introduction

Recent research in machine learning has revealed that one-class classification is of special importance in a number of real world application areas. For example, in order to classify sites of “interest” to a web surfer, the sites that are of interest are regarded as the positive instances. However, those “non-interest” ones are normally difficult or expensive to obtain and define. In such cases, one-class classification is the better solution. One essential difference between one-class and conventional classification is that in one-class learning, it is assumed that only the target class information is available. In other words, in the classifier training process, instances from the target class are used and there is no information about its counter examples.

Normal classifiers generally perform poorly on imbalance data sets because most classifiers are designed to generalize from training data and output the simplest hypothesis. This is based on the principle of Occam’s razor. With imbalance data, the simplest hypothesis is often the one that classifies almost all instances as the majority class. One-class learning strategy is an alternative choice to overcome this problem. This is relatively a new research area, although some work has already been done [4,5]. The advantage of the one-class classifier is that by discarding the distractive majorities, the “space” where minority data resides could be better determined.

However, one-class learning may cause another problem on imbalance data: the classifier may overfit the training minority class. Another issue we consider is the parameter selection. The goal of this paper is to investigate the possibilities of applying one-class learning to imbalance data and optimizing the classifiers with all available information. We propose a one-class classification framework to solve these problems. The majority instances are not involved in the classifier training procedure. But they are utilized to help in optimizing the parameters of the kernel-based one-class classifiers: one-class SVM and Support Vector Data Description(SVDD). Experimental results show that with the assistance of the majority class, the overall performance is improved and the accuracy rates on minority and majority class are more balanced.

## 2 Parameter Estimation in Kernel-Based One-Class Classifiers

One-class SVM was first proposed in [6] to estimate the probability density function where the data set is drawn from. In other words, the learning purpose of the algorithm is to return a function  $f$  that takes the value +1 in a “small” region capturing most of the training data points and -1 elsewhere. The learning strategy is firstly to map the data into a feature space with a certain kernel function; then separate the mapped vectors from origin with maximum margin using the standard SVM.

SVDD, which is first proposed in [1], aims to find a sphere with minimum volume containing all the data objects. It is allowed to ignore some remote objects outside the sphere to represent the data better. Let  $x_i (i \in [1, l])$  denote the training examples labeled as the positive class, this learning problem becomes a quadratic optimization problem eventually as follows:

$$\begin{aligned} & \max \sum_i \alpha_i (x_i \cdot x_i) - \sum_{i,j} \alpha_i \alpha_j (x_i \cdot x_j) \\ & \text{subject to: } \alpha_i \geq 0 \\ & \sum_i \alpha_i = 1 \end{aligned} \tag{1}$$

Note that it is stated completely in terms of inner products between vectors. Hence, inner products of objects ( $x_i \cdot x_j$ ) can be replaced by a kernel function  $K(x_i, x_j)$ , similar as in one-class SVM.

Some common kernel functions include linear, sigmoid, polynomial and Gaussian kernels. The Gaussian kernel, which is given by

$$k(x, y) = e^{-||x-y||^2/s} \tag{2}$$

has been popular for practical use. We employ the Gaussian kernel in the paper only for simplicity and efficiency.

The width parameter  $s$  is the first parameter needs to be taken into account. In SVDD, the width parameter  $s$  regulates the number of support vectors. When increasing  $s$ , the volume of the closed region is enlarged and the number of support vector is decreased. The aim of SVDD is to reduce the volume covered by the boundary, but this comes at the price of overfitting. These are contradictory and a trade-off is definitely necessary. The width parameter  $s$  plays an important role to control the balance.

Another parameter in both learning approaches is the rejection rate  $v \in (0, 1]$ . This user specified parameter determines the ratio of points considered to be “outliers” in the target class. In one-class SVM, it is an upper bound on the fraction of training points outside the estimated region. It is also a lower bound on the fraction of support vectors. This value of  $v$  indicates the size of region the learning scheme covers in both one-class SVM and SVDD. The smaller value  $v$  has, the bigger size the estimated region will be. The ideal solution is to find a smaller region covers more fraction of the training points. Hence, similar as the width parameter  $s$ ,  $v$  also determines the trade-off between the covered region and possible overfitting problem.

### 3 Related Work

One-Class SVM has been applied to document classification in [4]. They did extensive tests on various one-class classification algorithms with different document representations. In particular, they tested the one-class SVM algorithms with several kernel functions, including linear, polynomial, sigmoid and Gaussian kernels. The results achieved on the Reuters data set show that the one-class SVM is more robust to smaller categories. However, it is very sensitive to the parameters and choice of kernel.

The outstanding performance of one-class SVM applied to data with heavily unbalanced class proportions is reported in [5]. Experiments on one high-dimensional real world data and low-dimensional synthetic data with noise prove that there is a consistent pattern of decreasing performance with increasing proportion of negative class instances. On the Reuters data set, when removing the most frequent features, the drop in performance for 2-class SVM models is much larger than the one-class one. All these indicate that one-class classification does have potential to be superior to the normal classifier on imbalance data. However, in their work, they only considered the linear kernel for SVMs and the parameter selection issue is hardly mentioned.

Parameter selection has been an important research issue appeared in many works. In [9], a generalization performance estimation method for one-class SVM is proposed. This method is an extended version of the  $\xi\alpha$ -estimate approach, which is used to estimate the generalization performance of standard SVM. Generic algorithm is employed to optimize the training model. For SVDD, an estimation method based on the consistency measurement is proposed in [8]. This consistency measurement is heavily related to the rejection rate  $v$ , which is user-specified. Thus, the parameter selection is still not solved completely.

## 4 One-Class Classification Framework

The advantage of one-class classification on imbalance data is that by discarding the majority information during training procedure, it is no longer a distraction for the classifier. However, when only minorities are participating in the training, there is a high probability that the classifier will overfit the target class so that the performance on majorities deteriorates. Parameter selection is another issue which needs to be taken into consideration, especially for classifiers sensitive to parameters. The parameter optimization needs to be based on a certain criteria. This criteria should not only reflect the estimated classification performance on target class, but also consider its accuracy on the “outliers”. Therefore, although the majorities are excluded in the training procedure to avoid their distractions on minorities, they could be employed when estimating the performance of constructed classifier. By further tuning the parameters with the assistance of majority class, the hypothesis is optimized.

The one-class classification framework is based upon the above idea and is composed of three stages:

1. Learning Stage;
2. Evaluation Stage;
3. Optimization Stage.

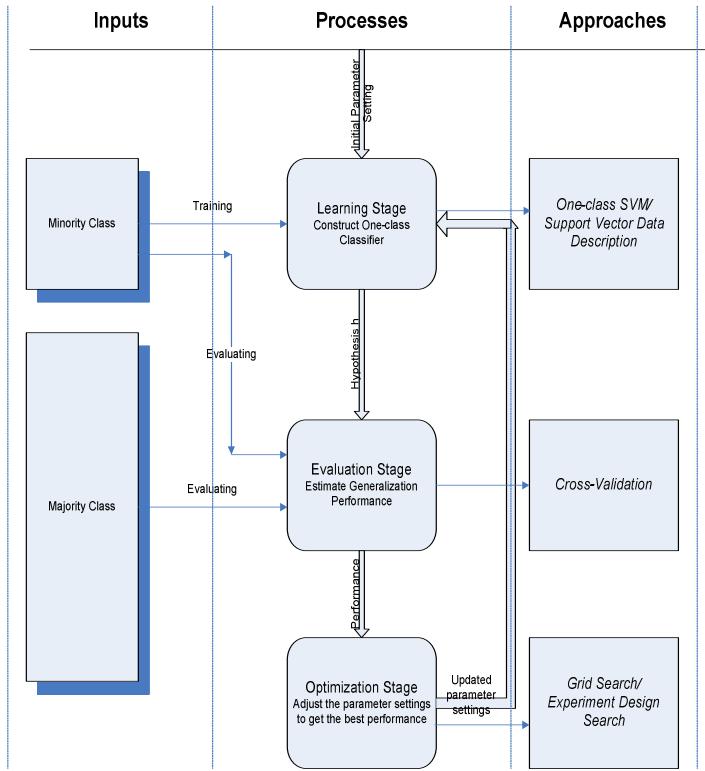
Figure 1 illustrates the detailed procedure. In the first stage, to construct the classifier, normally an initial parameter setting needs to be given. Generally this initial setting is chosen randomly. The classifier is trained from the minorities, i.e., the target class in the training set. Based upon the hypothesis induced in the first step, the performance is evaluated utilizing BOTH minority and majority data. This is the evaluation stage. Finally at the optimization stage, with the generalization performance as the objective function, the parameter settings are adjusted to achieve better results. The whole process is iterating until stopping at a certain criteria where the best parameter setting is selected.

### 4.1 Learning Stage

This is the step where the one-class classifier is constructed using the minority class. The purpose of our research is to improve the classification accuracy on minorities in imbalance learning. Thus, classifier which is able to describe the minority data is needed. One-class classifier is the most suitable learning methods for this purpose since it neglects the distracting majorities. As shown in Figure 1., after this stage, a hypothesis is produced and will be estimated in the next step.

### 4.2 Evaluation Stage

Generalization performance estimation of a classifier is one of the important tasks in learning theory. Cross-validation is the most popular approach. Among the various versions of cross-validation estimator, the leave-one-out estimator is shown to be almost unbiased [3]. The leave-one-out estimator is described



**Fig. 1.** Framework of Applying One-Class Classifiers to Imbalance Data

as follows: From the training instances  $S = ((x_1, y_1), \dots, (x_n, y_n))$ , the first instance is removed. The resulting instances  $S^1 = ((x_2, y_2), \dots, (x_n, y_n))$  are used for training, producing a classification rule  $h_L^1$ . This rule is tested on the held out instance  $(x_1, y_1)$ . This process is repeated for all training examples. The generalization performance is measured by these test results.

However, leave-one-out is very expensive to run, especially in cases with huge amount of training data. Cross-validation is another option. An  $k$ -fold cross validation works as follows: firstly, we divide the training data into  $k$  equal-sized sections  $S_1, S_2, \dots, S_k$ . For each section  $S_i$ , the training model is built based upon the  $k - 1$  sections ( $S_1, \dots, S_{i-1}, S_{i+1}, \dots, S_k$ ) and tested on the left-out  $S_i$ . Apparently, leave-one-out estimator is the situation when  $k$  equals to  $n$ .

We employed both 10-fold cross-validation and leave-one-out estimator when estimating the generalization performance. However, to estimate the performance on majority outlier class, there is no need to use these strategies. Note that majority instances are not involved in the training procedure. Thus, the one-class classifier is first constructed on the entire minorities. The majority class then is regarded as the validation set. All the instances in this set are “outliers” for this classifier. Accordingly the accuracy rate on negative class could be calculated.

This estimation is a fast process since the classifier only needs to be constructed once.

The overall performance is measured upon the accuracy of both positive and negative classes. In our experiment, we employed g-metric means as introduced in Section 5.1.

### 4.3 Optimization Stage

Grid search is a straightforward and popular parameter search method. In grid search, a uniform grid is defined in the parameter space. Then points in each of the grid are evaluated and the global optimum is found in this space. The coarseness of the grid determines the quality of the solution and the efficiency of the search. Although lots of effort has been spent in parameter search for SVM, grid search is still regarded as the most reliable approach. The only problem is its high computational demands. Even moderately high resolution searches can result in a large number of evaluations and unacceptably long run times.

The idea of parameter selection based on design of experiments(DOE) was first proposed in [7]. This approach basically is to start with a very coarse grid covering the whole search space and iteratively refine both the grid resolution and search boundaries, keeping the number of samples at each iteration roughly constant.

A combination of three-level experiment design with the two-level experiment design constitutes the sampling pattern in our experiment design search method. The three-level design is also written as a  $3^k$  factorial design. It means that  $k$  factors are considered, each at 3 levels. Similarly, the two-level design is to consider  $k$  factors at 2 levels. In a two parameter space, if each parameter is considered as one factor, this approach will produce thirteen solutions( $3^2 + 2^2 = 13$ ). Please note that when we select the points, we first discretize each parameter space by dividing it into three or two equal-length sections. In this paper, the middle point of each section is chosen as the representative for each level.

In each search iteration, the system evaluates the classifier performance at the sampled points, i.e., with the selected parameter settings. The one with the best performance will be chosen and the search space is refined around it. Currently in our experiment, we half the parameter range after each iteration. However, this could be adjusted accordingly. If the new search space could center around the best point without going outside the original boundary, that is the best choice. Otherwise, the new search range will start from or end to the closest original bound and then extend to the other half section. This process is repeated as many times desired or once the points in the refined search space could not improve the previous performance any more.

## 5 Experimental Results

### 5.1 Data Sets and Evaluation Measurement

Four UCI datasets: abalone(19), letter(26), glass(7) and segment(1) are used first to evaluate the performance of different search algorithms. The label in the

parentheses indicates the target class we chose as the positive and the other classes are regarded as negative. Table 1 summarizes the details of each data set with respect to the number of positive, negative instances, positive-to-negative ratio and the number of attributes. All the attributes in these four data sets are continuous variables and scaled to either  $[-1, 1]$  or  $[0, 1]$ .

Another real world data set we used in our experiments is the Reuters-21578 Modified Apte Split. This benchmark text collection contains 9603 documents in the training set and 3299 documents in the test set. We preprocessed the documents using the standard stop word removing, stemming and converted the documents to high-dimensional vectors using TFIDF weighting scheme. In order to reduce the size of the term set, we discarded terms which appear in less than 5 documents and the total number of terms extracted finally is 6362. We have chosen 8 frequent topic categories which are all mildly imbalanced. Table 2 summarizes their details. It lists, for each specific topic, the number of positive documents (#+Training) and the positive-to-negative instance ratio in the training set.

**Table 1.** UCI Dataset Description

Dataset	Positive Insts.	Negative Insts.	Ratio	Attribute
Segment1	330	1980	1:6	19
Glass7	29	185	1:6	9
Letter26	734	19266	1:26	16
Abalone19	32	4145	1:130	8

**Table 2.** Reuters-21578 ModApte Dataset Description

Data set	Money-fx	Grain	Crude	Trade	Interest	Ship	Wheat	Corn
#+Training	475	371	330	369	347	197	212	181
Ratio(+:-)	1:19.2	1:24.9	1:28.1	1:25.2	1:26.7	1:47.8	1:44.3	1:52.1

The g-metric means suggested by Kubat et. al. [2] is employed to evaluate classifiers on highly imbalanced datasets. The calculation of g-metric means is as follows:

$$g = \sqrt{acc_P * acc_N}$$

where  $acc_P$  indicates the sensitivity metrics and  $acc_N$  the specificity. Sensitivity is defined as the accuracy on the positive instances(true positives/(true positives + false negatives)), and specificity is defined as the accuracy on the negative instances(true negatives/(true negatives + false positives)).

$F_1$  measure is also employed which is given as:

$$F_1 = \frac{2 \times recall \times precision}{recall + precision}.$$

## 5.2 Results

**Search Algorithm Comparison.** First of all, we compared the accuracy of parameter search algorithms on the small UCI data sets. Part of these results are also reported in [10]. Table 3 lists the g-metric mean accuracy rates. Table 4 gives more details of the search performance by listing the accuracy rate on positive and negative class respectively. From Table 3, we could see that the experiment design search achieves the best overall performance. Now look at the more detailed statistics in Table 4, we find that apparently, experiment design search did improve the accuracy rate on positive minority class compared to grid search. However, the accuracy rate on negative majority class of experiment design search is slightly worse than the grid search. From the definition of experiment design search, we could see that the advantage of it is that it could sample the parameter space in a better way. By increasing the grid resolution in each iteration, the search space is refined.

**Table 3.** Search Algorithm Comparison(G-metric)

Dataset	Grid Search	EDS
Segment	0.799	0.921
Glass	0.722	0.781
Letter	0.866	0.909
Abalone	0.583	0.635

**Table 4.** Search Algorithm Comparison(Recall on P/N class)

Dataset	Grid Search		EDS	
	P	N	P	N
Segment	0.667	0.957	0.94	0.903
Glass	0.552	0.946	0.724	0.843
Letter	0.869	0.863	0.895	0.924
Abalone	0.438	0.777	0.594	0.679

**Comparison with Other One-Class Classifiers.** We compared the parameter optimized one-class SVM and SVDD on Reuters21578 text collection with the results published in [4]. We select the experiment design search due to its better performance. In addition, for one-class SVM, leave-one-out estimation is employed; for SVDD, 10-fold cross-validation is used.

To the best of our knowledge, [4] is the first work to apply one-class SVM on document classification. However, they discarded the negative class information and in their experiments, they only considered the impact of various document representation on classification performance, nor the parameter selection in one-class SVM. Table 5 shows the comparison details. The evaluation measurement is F1 measure. The algorithms we compared with include the standard One-class SVM(RBF kernel, binary document representation), Outlier-SVM(Linear kernel, binary document representation), Neural Networks, one-class Naive Bayes,

**Table 5.** Comparison of standard One-class SVM(OC), Outlier-SVM(OS), Neural Networks(NN), Naive Bayes(NB), Nearest Neighbour(NN2), Prototype Algorithm(P) and Parameter Optimized One-Class SVM(P-OC) and Parameter Optimized SVDD(P-S)

Dataset	OC	OS	NN	NB	NN2	P	P-OC	P-S
Money	0.514	0.563	0.642	0.493	0.468	0.484	0.550	0.585
Grain	0.585	0.523	0.473	0.382	0.333	0.402	0.742	0.48
Crude	0.544	0.474	0.534	0.457	0.392	0.398	0.715	0.7
Trade	0.597	0.423	0.569	0.483	0.441	0.557	0.634	0.709
Interest	0.485	0.465	0.487	0.394	0.295	0.454	0.609	0.592
Ship	0.539	0.402	0.361	0.288	0.389	0.370	0.427	0.445
Wheat	0.474	0.389	0.404	0.288	0.566	0.262	0.647	0.545
Corn	0.298	0.356	0.324	0.254	0.168	0.230	0.542	0.571
Average	0.505	0.449	0.474	0.380	0.380	0.395	0.608	0.578

one-class Nearest Neighbour(Hadamard document representation) and Prototype Algorithm(tf-idf document representation). According to [4], the listed F1 results are the best among various document representations and parameter settings.

As shown in the table, in most cases, the parameter optimized approaches, either one-class SVM or SVDD, achieve much better results than the stand-alone one-class classifiers. The last row of Table 6 presents the average F1 accuracy for each learning method on these eight data sets. The parameter optimized one-class SVM beats all the other schemes. The optimized SVDD is slightly worse. The standard one-class SVM with the default settings performs the best compared with the other one-class learning approaches. However, by further optimizing its parameters, the average accuracy is increased about 10%. This further indicates the significance of parameter selection.

## 6 Conclusion

In this paper, we propose a general parameter optimization framework for one-class classification on imbalance data. In this framework, only minority class is engaged in inducing the classifier but both majority and minority instances are utilized in evaluating the generalization performance of the constructed classifier. Using the generalization performance as optimization criteria , the classifier is constructed with the best parameter settings. Empirical results prove that this framework with the one-class SVM and SVDD both achieve much better accuracy results than the standard OC-SVM and other one-class learning schemes. In particular, the parameter optimized one-class SVM ranked the top among all the learning schemes.

One possible direction of the future work is to investigate more one-class classification algorithms within this framework. Another important issue is the parameter search algorithms. A better way to search for the best setting is to let the classification performance converge to a optimal value rather than do a exhaustive search.

## References

1. Robert P.W. Duin David M.J. Tax. Support vector domain description. *Pattern Recognition Letters*, 20:1191–1199, 1999.
2. Miroslav Kubat and Stan Matwin. Addressing the curse of imbalanced training sets: one-sided selection. In *Proc. 14th International Conference on Machine Learning*, pages 179–186. Morgan Kaufmann, 1997.
3. A. Lunts and V. Brailovskiy. Evaluation of attributes obtained in statistical decision rules. *Engineering Cybernetics*, pages 98–109, 1967.
4. Larry M. Manevitz and Malik Yousef. One-class svms for document classification. *Journal of Machine Learning Research*, 2:139–154, 2001.
5. Bhavani Raskutti and Adam Kowalczyk. Extreme re-balancing for svms: a case study. *SIGKDD Explorations*, 6:60–69, 2004.
6. Bernhard Scholkopf, John C. Platt, John Shawe-Taylor, Alex J. Smola, and Robert C. Williamson. Estimating the support of a high-dimensional distribution. *Neural Computation*, 13:1443–1471, 2001.
7. Carl Staelin. Parameter selection for support vector machines. Technical Report HPL-2002-354R1, Hewlett-Packard Company, 2003.
8. D.M.J. Tax and K.R. Müller. A consistency-based model selection for one-class classification. In *Proceedings of the International Conference on Pattern Recognition 2004*, pages 363–366. IEEE Computer Society, Los Alamitos, CA, 2004.
9. Quang-Anh Tran, Xing Li, and Haixi Duan. Efficient performance estimate for one-class support vector machine. *Pattern Recognition Letters*, 26:1174–1182, 2005.
10. Ling Zhuang and Honghua Dai. Parameter estimation of one-class svm on imbalance text classification. In *Proceedings of the 19th Canadian conference on artificial intelligence*, 2005.

# Clustering-Based Nonlinear Dimensionality Reduction on Manifold

Guihua Wen<sup>1,3</sup>, Lijun Jiang<sup>1</sup>, Jun Wen<sup>2</sup>, and Nigel R. Shadbolt<sup>3</sup>

<sup>1</sup> South China University of Technology, Guangzhou 510641, China  
`crghwen@scut.edu.cn, cssturat@sohu.com`

<sup>2</sup> Hubei Institute for Nationalities, HuBei Ensi 445000, China  
`wenj_64@163.com`

<sup>3</sup> University of Southampton, Southampton SO17 1BJ, United Kingdom  
`nrs@ecs.soton.ac.uk`

**Abstract.** This paper proposes a clustering-based nonlinear dimensionality reduction approach. It utilizes the clustering approaches to form the clustering structure by which the distance between any two data points are rescaled to make data points from different clusters separated more easily. This rescaled distance matrix is then provided to improve the nonlinear dimensionality reduction approaches such as Isomap to achieve the better performance. Furthermore, the proposed approach also decreases the time complexity on the large data sets, as it provides good neighborhood structure that can speed up the subsequent dimensionality reducing process. Unlike the supervised approaches, this approach does not take the labelled data set as prerequisite, so that it is unsupervised. This makes it applicable to the broader domains. The conducted experiments by classification on benchmark data sets have validated the proposed approach.

## 1 Introduction

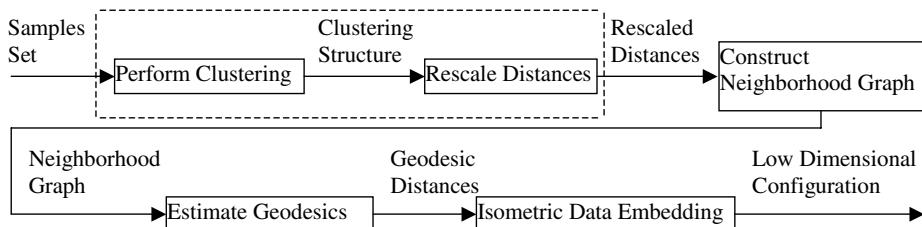
Dimensionality reduction is often required to find the meaningful low-dimensional structures hidden in the high-dimensional observations for exploratory data analysis such as classification and visualization. Because classic linear approaches can be reliably applied only when the data manifolds are linear subspaces, two new approaches have recently been developed to deal with nonlinear data manifold. One is locally linear embedding (LLE) that is based on local approximation of the geometry of the manifold[4]. LLE has many variants, such as Laplacian eigenmaps [5]and Hessian eigenmaps(HLLE) [6], incremental LLE [19], supervised LLE [13,19], integrated LLE with classic PCA or LDA [22,21], integrated LLE with SOM [14] etc. The other new approach is Isomap that preserves the manifold geometry at all scales and has better ability to deal with nonlinear subspaces [1]. It also has many variants, such as Landmark Isomap [2], supervised Isomap [9], spatio-temporal extension of Isomap [16], incremental extension of Isomap [11], integrated Isomap with fuzzy LDA [23] etc. The other researches are also investigated such as the selection of the optimal parameter value for LLE and Isomap [24,12], integration of LLE and Isomap [25] etc. These two approaches have their

own superiorities so that they have been being developed simultaneously for various context applications. Generally Isomap outperforms LLE in performance and is well theoretically studied such as in convergence proof and the conditions for successful recovery of co-ordinates [11]. LLE is computationally simpler and can give useful results on a broader range of manifolds [7].

Despite Isomap performs well in many cases, it often fails to nicely deal with noisy data or sparsely sampled data [9]. This is because in these cases the local neighborhood structure which Isomap largely depends on is critically distorted. This makes Isomap vulnerable to short-circuit errors that some neighbors of the current point come from other different folds so that these neighbors are not nearest ones on manifold[3,15]. This can in turn lead to drastically incorrect low-dimensional embedding. In order to solve these problems, some supervised Isomap are proposed [8,9], which employ the class labels of the input data to guide the manifold learning. They obviously improve the Isomap in classification and visualization. However, they need a large number of labelled training data. When sometimes these labelled data are not available, the supervised approaches can not work. This paper utilizes the automatic clustering techniques to partition the data into clusters so as to form the clustering structure, on which rescaling of Euclidean distance can be realized to improve the Isomap.

## 2 Proposed Approach

Our proposed approach is structurally illustrated as Figure 1, where the difference between our approach and the classic Isomap is the added two steps in dotted rectangle. The first step applies clustering approach to form the clustering structure for the input data, on which the distances between any two points are rescaled in the second step. The rescaled distance matrix instead of original Euclidean distance matrix will be fed into the remaining steps which are the same as that of the classic Isomap.



**Fig. 1.** Framework of Clustering-based Isometric Embedding

### 2.1 Clustering Approaches

Clustering is the process of grouping a set of objects into classes of similar objects. Recognized as an unsupervised learning method in that the class label of each object is unknown and the data are grouped together simply based on the

similarity between them, clustering is well fitted in many problems. Among available clustering approaches,  $k$ -mean approach is computationally efficient for the large data sets with both numeric and categorical attributes [10]. It is formulated by minimizing a formal objective function, mean-squared-error distortion:

$$\min \sum_{i=1}^N \|x_i - c_{p(i)}\|$$

where  $N$  is the number of data samples,  $x_i (i = 1 \dots N)$  is a data sample,  $p(i)$  is the cluster label of  $x_i$ ,  $c_j (j = 1 \dots k)$  is a cluster centroid, and  $k$  is the number of clusters.

The classic  $k$ -means approach firstly chooses  $k$  samples at random from the whole sample space to approximate centroids of initial clusters, and then iteratively updates the centers until no reassignment of patterns to new cluster centers occurs. In every step, each sample is allocated to its closest cluster center and cluster centers are reevaluated based on current cluster memberships. Generally it performs well, but it also quite depends on the initial setting of centers. Many works have been investigated to overcome this trouble [10]. Its most remarkable advantage is its conceptual simpleness and lower computational complexity. It is preferable if efficiency is a consideration or data sets are very large. This approach will be applied to the later experiments.

## 2.2 Clustering-Based Isomap

Now we can apply the  $k$ -mean approach to partition the input data into many clusters. If we assign all samples in the same cluster the same class label, the idea of supervised approaches such as their rescaling functions can be utilized. The WeightIso approach rescales the Euclidean distance  $d_e$  between two data points with a constant factor less than 1 if their class labels are the same, while the distance between two data points from different classes keeps unchanged [8].

$$d_c(x_i, x_j) = \begin{cases} \lambda d_e(x_i, x_j) : x_i \in c_k, x_j \in c_k, 0 < \lambda < 1 \\ d_e(x_i, x_j) : x_i \in c_k, x_j \in c_t, k \neq t \end{cases}$$

This makes the two points closer to each other in the feature space if they belong to the same class, so that it can make classification in the feature space easier. This approach is simpler, but easily remove the structure of intra-cluster, so that it may not be suitable for visualization.

The S-Isomap rescales the Euclidean distance as follows [9], by changing both intra-class distance and interclass distance

$$d_c(x_i, x_j) = \begin{cases} \sqrt{1 - e^{-\frac{d_e(x_i, x_j)}{\beta}}} : x_i \in c_k, x_j \in c_k \\ \sqrt{e^{\frac{d_e(x_i, x_j)}{\beta}} - \alpha} : x_i \in c_k, x_j \in c_t, k \neq t \end{cases}$$

This approach is more efficient in visualization and classification. Here two parameters need to be tuned carefully. They easily make the neighborhood graph

of the input data disconnected. This is a case that Isomap can not handle. Some approaches have been proposed to deal with this problem [15].

To simplify the problem, based on the proper combination of above two approaches, we design a simple rescaling function for classification with attempt to prove that clustering approaches can be applied to improve the nonlinear dimensionality reduction.

$$d_c(x_i, x_j) = \begin{cases} e^{-\frac{d_e(x_i, x_j)}{c\lambda}} d_e(x_i, x_j) : x_i \in c_k, x_j \in c_k \\ d_e(x_i, x_j) : x_i \in c_k, x_j \in c_t, k \neq t \end{cases}$$

The parameter  $c\lambda$  is applied to prevent the  $d_c(x_i, x_j)$  decreases too fast when  $d_e(x_i, x_j)$  is relatively large, where  $c$  takes the average Euclidean distance between all pairs of data points and  $0 < \lambda < 1$ . The distance between points in the same cluster decreases exponentially. The further the points within the same cluster, the faster their distance decreases so as to relatively enlarge the gap between clusters and keep the structure between near points. It can be easily observed that when the Euclidean distance is equal, the inter-cluster distance is larger than the intra-cluster distance. This is suitable for classification tasks.

Now we can apply  $k$ -mean clustering approach and rescaling function to improve the basic Isomap. The complete clustering-based Isomap is illustrated as follows

**Algorithm.** ClustIsomap( $X, k_g, k_c, d$ )

Input parameters:  $X = \{x_i\}$  is the input data set,  $k_g$  is the neighborhood size,  $k_c$  is the number of the clusters, and  $d$  is the dimension of the projected space, and the output is the dimensionally reduced data set  $Y = \{y_i\}$ .

1. Utilize any clustering approach such as  $k$ -means( $X, k_c$ ) to separate the whole data set into  $k_c$  clusters  $\{c_i, i = 1..k_c\}$ .
2. Calculate the rescaled distance matrix  $\{d_c(x_i, x_j)\}$  based on the clustering structure .
3. Construct the weighted neighbor graph  $G_c = (V, E)$  for the data set  $X$ , by connecting each point to all its  $k_g$ -nearest neighbors, where  $(v_i, v_{i+1}) \in E$ , if  $x_j$  is a member of  $k_g$  neighbors of  $x_i$  in terms of distance metric  $d_c$ . This edge has weight  $d_c(x_i, x_j)$ .
4. Employ the  $G_c$  to approximately estimate the geodesic distance  $d_g(x_i, x_j)$  between any pair of points as the shortest path through graph that connects them.

- 1) Initialize

$$d_g(x_i, x_j) = \begin{cases} d_c(x_i, x_j) : (x_i, x_j) \in E \\ \infty : (x_i, x_j) \notin E \end{cases}$$

- 2) Iterate the following formula using all  $k$

$$d_g(x_i, x_j) = \min_k \{d_g(x_i, x_j), d_g(x_i, x_k) + d_g(x_k, x_j)\}$$

- 3) Final matrix  $D_G = \{d_g(x_i, x_j)\}$  is composed of geodesic distances between any two points.

5. Construct d-dimensional embedding: Let  $\lambda_p$  be the  $p$ -th eigenvalue (in decreasing order) of the matrix  $\tau D_G$  (The operator  $\tau$  is defined by  $\tau D_G = -HSH/2$ , where  $S$  is the matrix of squared distances  $S_{ij} = D_{ij}^2$ , and  $H$  is the "centering matrix"  $H_{ij} = \delta_{ij} - 1/N$ ,  $\delta_{ij}$  is the Kronecker delta function, and  $v_p^i$  be the  $i$ -th component of the  $p$ -th eigenvector. Then set the  $p$ -th component of the  $d$ -dimensional coordinate vector  $y_i$  equal to  $\sqrt{\lambda_p} v_p^i$  (This is actually a procedure of applying classical MDS to the matrix of graph distances).

Compared with the basic Isomap, ClustIsomap adds step 1 and modify step 2. It should be noted in step3 that the edges of neighborhood graph are weighted by rescaled distance instead of Euclidean distance. The step 4 and step 5 remains the same as that of the basic Isomap. Therefore the time complexity increases from step 1 where the additional complexity is  $O(n)$ . On performance, except for sharing the advantages with the basic Isomap such as asymptotic convergence guarantees, ClustIsomap performs better in classification depending on its rescaled distance which leads to the optimized neighborhood structure.

### 3 Experimental Results

We make comparison between Isomap and the proposed ClustIsomap by classification through several experiments on benchmark data sets. Isomap uses the published matlab code [1]. The ClustIsomap is also implemented in Matlab7.0.

#### 3.1 Experiment Data Set

Experimental benchmark data sets are listed as Table 1, which are available from UCI Machine Learning Repository. On the frontal seven data sets, three experiments are performed to compare ClustIsomap with Isomap in terms of classification error. Each data set is split into the training set (70%) and test set(30%). All samples in data set, including the test sample, are projected into the reduced space for classification, where classification is performed using KNN classifier. KNN often results in the very good classification performances in many practical applications and also extremely simple to be implemented [17]. On the last data set, the experiments are performed to compare both approaches in time complexity.

**Table 1.** Datasets used in our experiments

No. Dataset	Size	Attributes	Classes
1 Dematology	358	34	6
2 Diabetes	768	8	2
3 Iris	150	4	3
4 Glass	214	9	6
5 Segmentation	210	19	7
6 Ionosphere	351	34	2
7 Wine	178	13	3
8 Yeast	1484	8	10

### 3.2 Performance Evaluation by Classification

When classification performed on the high dimensional large data sets, the curse of dimensionality causes highly biased estimates, thereby reducing the classification accuracy[9]. Dimensionality reduction can be applied to solve this problem as a preprocess for classification [8,9,13]. The better dimensionality reduction approach can remove the irrelevant noise but can keep necessary information so as to get better classification results.

We perform three experiments on the given data sets, where parameter  $k_c=2\ldots 6$  for clustering approach,  $k_g=10\ldots 40$  for estimating geodesic distance,  $\lambda=0.4\ldots 1.2$  with step 0.2 for the rescaling function, and  $k_{nn}=3\ldots 40$  for KNN classifier. These parameters are chosen by exhaustive grid search in the range. To investigate how much information in the reduced space is preserved, KNN in full dimensional space is also compared.

#### **Experiment 1:** Classification errors on the different data sets

This experiment aims to evaluate the ability of Isomap and ClustIsomap to remove the irrelevant noise but keep necessary information for classification. Both Isomap and ClustIsomap project each data set to its half dimensional space, where KNN classifier is applied to perform the classification on the projected data set.

**Table 2.** Classification errors (%)on data sets with half of the original dimensionality

DatasetNo	1	2	3	4	5	6	7
Full	11.67	23.44	14.29	32.39	17.14	8.55	16.67
Isomap	10.83	21.48	12.24	30.99	17.14	5.13	16.67
ClustIsomap	<b>10.00</b>	<b>19.92</b>	<b>00.00</b>	<b>29.58</b>	<b>14.29</b>	<b>1.71</b>	<b>13.33</b>

It can be seen from Table 2 that ClustIsomap performs better than Isomap on all data sets. The maximum gap on data set no.3 is up to 12.24%. It can be also observed that KNN always performs better on the projected data sets than on the data sets with full dimensionality. This illustrates that dimensionality reduction techniques can be applied to improve the performance of data classification.

#### **Experiment 2:** Classification errors against the dimensions.

To decreases the computational burden for classification or to visualize the data, it is very important to reduce the high dimensional data set to the very lower dimensional space while maintaining as much information as possible that is useful for classification. To make comparison between two approaches in this aspect, we perform experiments to project two data sets to the lower dimensional space with various dimensions, where Dematology and Ionosphere are applied, because they are high dimensional.

It can be seen from Table 3 that on both data sets ClustIsomap performs better than Isomap at any lower dimension. This illustrates that ClustIsomap is more robust to the lower dimensions. On Ionosphere data set, both approaches

**Table 3.** Classification error (%)on different low dimensions

Dermatology (No.1)										
Dimension	2	5	8	11	14	17	20	23	26	29
Isomap	25.00	16.67	15.00	10.83	10.83	10.83	<b>10.00</b>	10.00	10.83	10.83
ClustIsomap	20.00	14.17	12.50	10.83	9.17	9.17	9.17	<b>8.33</b>	8.33	8.33
Ionosphere (No.6)										
Dimension	2	5	8	11	14	17	20	23	26	29
Isomap	4.27	<b>2.56</b>	4.27	3.42	5.13	5.13	5.13	5.98	5.13	5.13
ClustIsomap	2.56	<b>0.85</b>	0.85	0.85	1.71	1.71	0.85	2.56	2.56	1.71

reach the least classification errors at very low dimension,namely, 5, where ClustIsomap still outperforms Isomap by 1.71%.

### Experiment 3: Classification errors against the number of clusters.

ClustIsomap depends on the  $k$ -mean clustering approach that depends on the number of clusters. To investigate the importance of the selection of the number of clusters, we perform experiments on two data sets: Dermatology and Ionosphere.

It can be observed from Figure 2 that the classification error of ClustIsomap decreases in the non-monotonic way against the number of clusters. On Dermatology data set which has 6 classes, ClustIsomap performs best using 20 clusters. On Ionosphere data set which has 2 classes, ClustIsomap performs best using 5 clusters. Obviously, the number of clusters to acquire the optimal classification performance is always larger than the number of prelabelled classes. This illustrates that to reach the better performance, the larger number of clusters should be tested firstly. Because the supervised approaches only apply the labelled classes, they can not discover the smaller granule clustering structure of the data set. This means that the clustering-based Isomap may also exceed the supervised Isomap under the same rescaling function. However this needs to be investigated more by experiments in the future.

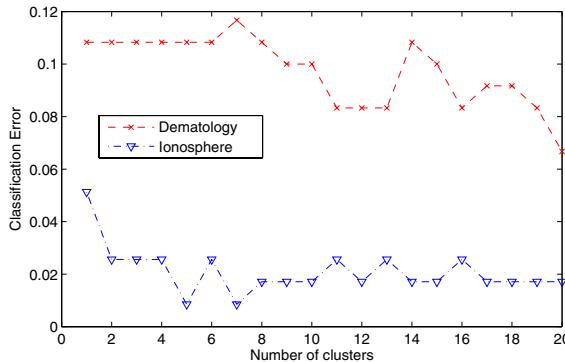
### 3.3 Time Complexity Comparison

The ClustIsomap needs additional time to perform clustering so as to form the optimized neighborhood structure. However the optimized neighborhood structure can speed up the subsequent dimensionality reducing process. This leads to the whole time complexity of the approach does not increase. This can be empirically illustrated by experiments on data set No.8, because it has more samples.

We generate 10 data sets from data set No.8 with different size, and then test Isomap and ClustIsomap on these data sets respectively, where  $k=10$  for

**Table 4.** Comparison of time complexity (seconds)

Datasize →	500	600	700	800	900	1000	1100	1200	1300	1400
Isomap	10.10	17.50	27.00	40.30	56.40	78.30	101.20	133.40	167.80	211.80
ClustIsomap	10.00	17.30	26.90	40.10	56.30	78.30	100.90	133.40	167.40	212.10



**Fig. 2.** Classification errors against the number of clusters

estimating the geodesic distance,  $m=6$  for the number of the clusters, and  $\lambda=1$ . It can be observed from Table 4 that ClustIsomap still exceeds the Isomap on many data sets in time complexity.

## 4 Conclusion

The main contribution of this paper is to utilize the clustering techniques to automatically rescale the distance among points within the same cluster so as to improve the nonlinear dimensionality reduction approach. Unlike the supervised approaches, our approach does not take any prelabelled data set as prerequisite, so that it has a broader range of applications. Furthermore from the small scale experiments, it does not increase the time complexity, as it provides good neighborhood structure that can speed up the subsequent dimensionality reducing process. This means that the more complicated or even time-consuming clustering approaches can be applied to construct the better neighborhood structure whilst the whole time complexity will not raise.

In the future, we will apply more efficient clustering techniques to improve the Isomap, such as modified k-mean clustering approach [10], and do experiments on the large data sets. Another work is to redefine the rescaling function to consider both classification and visualization. This includes to make detailed comparison between our approach with supervised approaches. Finally due to the idea behind our approach is general, we will directly apply it to improve locally linear embedding approaches.

## Acknowledgement

This work is supported by Natural Science Foundation of China(60003019), Hubei Science and Technology Project (2005AA101C17),and UK Engineering and Physical Sciences Research Council under grant number GR/N15764/01.

## References

1. J.B. Tenenbaum and V. de Silva and J.C. Langford: A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science*, **290**(2000)2319–2323
2. V.D. Silva and J. B. Tenenbaum: Global versus local methods in nonlinear dimensionality reduction. *Neural Information Processing Systems*, **15**(2003)705–712
3. M. Balasubramanian and E.L. Schwartz: The Isomap Algorithm and Topological Stability. *Science*, **295**(2002)7–7
4. S.T. Roweis and L.K. Saul: Nonlinear Dimensionality Reduction by Locally Linear Embedding. *Science*, **290**(2000)2323–2326
5. M. Belkin and P. Niyogi: Laplacian Eigenmaps for Dimensionality Reduction and Data Representation. *Neural Computing*, **15**(2003)1373–1396
6. D.L. Donoho and C. Grimes: Hessian eigenmaps: Locally linear embedding, techniques for high-dimensional data. *Proc.Natl.Acad. Sci. U. S. A.*, **100**(2003) 5591-5596
7. L. K. Saul and S.T. Roweis: Think Globally, Fit Locally: Unsupervised Learning of Low Dimensional Manifolds. *Journal of Machine Learning Research*, **4**(2003) 119–155
8. M.Vlachos and C.Domeniconi and D.Gunopoulos et al.: Non-Linear Dimensionality Reduction Techniques for Classification and Visualization. *Knowledge Discovery and Data Mining*. Edmonton, Canada(2002).
9. X.Geng and D.C.Zhan and Z.H.Zhou: Supervised Nonlinear Dimensionality Reduction for Visualization and Classification. *IEEE Transactions on Systems, Man and Cybernetics*, **35**(2005)1098–1107
10. D.Charalampidis: A modified k-means algorithm for circular invariant clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **27**(2005) 1856 - 1865
11. Law MHC and Jain AK: Incremental nonlinear dimensionality reduction by manifold learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **28**(2006) 377-391
12. Shao C and Huang HK: Selection of the optimal parameter value for the ISOMAP algorithm. *Lecture Notes in Artificial Intelligence*, **3789**(2005) 396-404
13. de Ridder D, Kouropeteva O, Okun O, et al.: Supervised locally linear embedding. *Lecture Notes in Artificial Intelligence*, **2714**(2003) 333-341
14. Xiao J, Zhou ZT, Hu DW, et al. Self-organized locally linear embedding for non-linear dimensionality reduction. *Lecture Notes in Computer Science*, **3610**(2005) 101-109
15. Yang L: Building k edge-disjoint spanning trees of minimum total length for isometric data embedding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **27**(2005) 1680-1683
16. O. Jenkins and M. Mataric: A Spatio-temporal Extension to Isomap Nonlinear Dimension Reduction. *Proceedings of the Twenty-First International Conference on Machine Learning*, Alberta, Canada(2004) 1680-1683
17. F.Sebastiani: Machine Learning in Automated Text Categorization. *ACM Computing Surveys*, **34**(2002) 1-47
18. Kouropeteva, Olga, Okun, Oleg; Pietikainen, Matti: Incremental locally linear embedding. *Pattern Recognition*, **38**(2005) 1764-1767
19. de Ridder, D.; Loog, M.; Reinders, M.J.T.: Local Fisher embedding. *Proceedings of the 17th International Conference on Pattern Recognition*, **2**(2004) 295 - 298

20. Saxena A, Gupta A, Mukerjee A: Non-linear dimensionality reduction by locally linear isomaps. Lecture Notes in Computer Science,**3316**(2004) 1038-1043
21. Chang JP, Shen HX, Zhou ZH: Unified locally linear embedding and linear discriminant analysis algorithm (ULLELDA) for face recognition. Lecture Notes in Computer Science,**3338**(2004) 296-304
22. Abusham EE, Ngo D, Teoh A: Fusion of locally linear embedding and principal component analysis for face recognition (FLLEPCA). Lecture Notes in Computer Science,**3687**(2005) 326-333
23. ShifengWeng, Changshui Zhang, Zhonglin Lin: Exploring the structure of supervised data by Discriminant Isometric Mapping. Pattern Recognition, **38**(2005) 599- 601
24. O.Kouropteva and O.Okun and M.Pietikainen: Selection of the optimal parameter value for the locally linear embedding algorithm. Proceedings of 1th International Conference on Fuzzy Systems and Knowledge Discovery,Singapore(2002)359-363
25. A. Saxena and A. Gupta and A. Mukerjee: Non-linear dimensionality reduction by locally Linear Isomaps. Proceedings of 11th International Conference on Neural Information Processing, India(2004)1038-1043

# Sparse Kernel PCA by Kernel K-Means and Preimage Reconstruction Algorithms

Sanparith Marukatat

Information Research and Development Division,  
National Electronics and Computer Technology Center (NECTEC),  
Thailand Science Park, Phahon Yothin Road,  
Pathumthani 12120, Thailand  
[sanparith.marukatat@nectec.or.th](mailto:sanparith.marukatat@nectec.or.th)

**Abstract.** Kernel PCA, like other kernel-based techniques, is suffered from memory requirement and computational problems as well as from a tedious training procedure. This work shows that the objective function of Kernel PCA, *i.e.* the reconstruction error can be upper bounded by the distortion of K-means algorithm in the feature space. From this relation, we propose a simplification of Kernel PCA's training procedure by Kernel K-means algorithm. The application of preimage reconstruction algorithm allows further simplification and leads to a more computational economic solution.

## 1 Introduction

Non-linear analysis via kernel function has received increasing attentions in recent years. Support Vector Machine (SVM) [1], Kernel Principal Component Analysis (KPCA) [2] and Kernel Fisher Discriminant Analysis (KFDA) [3] are examples of such techniques. These techniques consist in replacing the dot product by a kernel function  $\kappa$ , therefore conducting the same analysis on the implicit feature space induced by the kernel function  $\kappa$ . Generally, there is a need to retain all or a large part of training data in memory even after the training step. This leads to a memory requirement and a computational complexity problem. As an example, the memory requirement for Kernel PCA algorithm is an order of  $\mathcal{O}(l^2)$  and the complexity of the training algorithm is an order of  $\mathcal{O}(l^3)$  where  $l$  is the number of training samples. Therefore the application of these kernel techniques to large scale problem is still problematic.

There are several works that address these problems in kernel-based literature, e.g. [4,5,6,7,8,9]. In [4] the author first select a set of linearly independent vectors among the training data and then approximate the found solution with the combination of these vectors. In [8], a greedy approach is used to incrementally select new basis vector as the one with maximal reconstruction error. The selected basis vectors approximate the PCA in the feature space. This procedure does not only reduce the memory requirement but also the complexity of the training procedure. In [6], the author consider the probabilistic formulation of PCA in the feature space. The maximum-likelihood approach is used to approximate the covariance

matrix by a reduced number of examples. The highly sparse form of KPCA is obtained without loss of effectiveness but with an increasing complexity of training procedure. In [5], the authors construct the basis vectors by solving the so-called *preimage problem*, i.e. by searching for a vector in the original input space which best approximates the combination of vectors in the feature space. Like [4], this approach aims at post-processing the solution found by classical approach, thus does not remove the tedious training step.

In order to simplify the training step, Kim et al. [9] adapt the classical generalized Hebbian algorithm by incorporating the kernel trick. This allows computing KPCA without storing the intermediate kernel matrix, so that large dataset can be processed.

A very different solution is proposed in [7]. Indeed, in this paper, the authors consider replacing each single kernel evaluation with independent random variables that behave like the original kernel in expectation. They showed that this randomized kernel evaluation leads to a sparse kernel matrix.

In this work, we focus on the simplification of the KPCA using a Gaussian kernel, not only its solution but also the training procedure. In order to simplify the solution outputs by the KPCA, we follow [5] by using the preimage reconstruction algorithm. To simplify the training procedure, we proposed an algorithm based on the K-means clustering algorithm. This relation between KPCA and K-means will be discussed in Section 2. The resulting training algorithm is presented in Section 3. Sections 4 and 5 present some experimental results and conclusions.

*Notation.* Let  $x_1, \dots, x_l$ ,  $x_i \in \mathcal{I}$  be the set of training vectors and let  $\kappa$  be a Gaussian kernel, i.e.

$$\kappa(x, y) = \exp\left(-\frac{\|x - y\|^2}{\sigma^2}\right) \quad (1)$$

Let  $\phi : \mathcal{I} \rightarrow \mathcal{F}$  be the mapping function associated with  $\kappa$  that maps any vector from the *input space*  $\mathcal{I}$  to the *feature space*  $\mathcal{F}$ . Hence we have  $\langle \phi(x); \phi(y) \rangle = \kappa(x, y)$ . The mapped training data on the feature space is then given by  $\phi(x_1), \dots, \phi(x_l)$ .

## 2 Kernel K-Means and Principal Space

In recent years, there are many works relating the K-means algorithm to the spectral structure of the covariance matrix or the kernel matrix used in PCA, e.g. [10,11,12]. For example, in [12] a lower bound on the objective function of K-means is proposed in term of eigenvalues of the covariance matrix used in PCA. From this relation, the authors build a K-means algorithm driven by the PCA. In this work, we are interested in the reverse process namely the PCA driven by the K-means algorithm, in the implicit feature space induced by a kernel function. Working in the implicit feature space, the covariance matrix of the data cannot be directly computed. In this work, we propose another derivation of the lower bound of the objective function of the K-means algorithm in this implicit feature space in term of the objective function of the KPCA. Indeed,

the KPCA aims at selecting the subspace of the feature space which maximizes the variance of data or equivalently which minimizes the reconstruction error of data. In next 2 subsections, we discuss the relation between the K-means' objective function and the reconstruction error of some subspace. This is the basic idea behind the training algorithm presented in next section (Section 3).

## 2.1 K-Means' Distortion and Reconstruction Error

We consider the clustering of data in the feature space induced by  $\kappa$ , i.e. the clustering of  $X = \phi(x_1), \dots, \phi(x_l)$ . Suppose that  $V = \phi(v_1), \dots, \phi(v_K)$  are  $K$  centers obtained by K-means clustering. These vectors minimizes the following distortion measure:

$$\mathcal{D}(X, V) = \sum_{i=1}^l \|\phi(x_i) - \phi(v_{i_k})\|^2 \quad (2)$$

with  $i_k = \arg \min_{k=1, \dots, K} \|\phi(x_i) - \phi(v_k)\|^2$ . Using the fact that  $\kappa(x, x) = 1, \forall x$ , by a straight calculation, this distortion can be rewritten as

$$\mathcal{D}(X, V) = \sum_{i=1}^l (2 - 2\kappa(x_i, v_{i_k})) \quad (3)$$

Furthermore, suppose that  $V$  is also an orthonormal basis of some subspace in the feature space, i.e.  $\kappa(v_i, v_j) = 0$  for  $i \neq j$  and  $\kappa(v_i, v_i) = 1$ . The reconstruction error associated with  $V$  is given by

$$\mathcal{E}(X, V) = \sum_{i=1}^l \|\phi(x_i) - \sum_{k=1}^K \langle \phi(x_i); \phi(v_k) \rangle \phi(v_k)\|^2 \quad (4)$$

$$= \sum_{i=1}^l \|\phi(x_i) - \sum_{k=1}^K \kappa(x_i, v_k) \phi(v_k)\|^2 \quad (5)$$

$$= \sum_{i=1}^l \left( 1 - \sum_{k=1}^K \kappa(x_i, v_k)^2 \right) \quad (6)$$

The last equality uses the fact that  $\kappa(x, x) = 1, \forall x$  and  $V$  is an orthonormal basis.

Comparing Eq. (3) to Eq. (6) we have

$$\mathcal{D}(X, V) - \mathcal{E}(X, V) = \sum_{i=1}^l \left( 1 - 2\kappa(x_i, v_{i_k}) + \sum_{k=1}^K \kappa(x_i, v_k)^2 \right) \quad (7)$$

$$= \sum_{i=1}^l \left( 1 - 2\kappa(x_i, v_{i_k}) + \kappa(x_i, v_{i_k})^2 + \sum_{k \neq i_k} \kappa(x_i, v_k)^2 \right) \quad (8)$$

$$= \sum_{i=1}^l \left( (1 - \kappa(x_i, v_{i_k}))^2 + \sum_{k \neq i_k} \kappa(x_i, v_k)^2 \right) \quad (9)$$

The two terms in summation of the right-hand-side of Eq. (9) are non negative, so we conclude that

$$\mathcal{D}(X, V) \geq \mathcal{E}(X, V) \quad (10)$$

This means that for the feature space induced by a Gaussian kernel, the reconstruction error of an orthonormal basis  $V$  is upper bounded by the K-means' distortion having these basis vector as centers. As K-means algorithm minimizes the distortion measure, it will also minimize the reconstruction error. This bound suggests another method for finding such basis for KPCA, by using centers resulting from K-means clustering under constraint that  $\kappa(v_i, v_j) = 0$  for  $i \neq j$ .

## 2.2 Constructing Basis from K-Means' Centers

While the result from previous subsection suggests an easy way of selecting principal space, it requires some modification of K-means algorithm. In this subsection, we ask whether we can decouple the orthogonality constraint from K-means algorithm to further simplify the training procedure. It turned out that using the orthonormal basis which spans the K-means' centers as projection axis, the similar bound can also be obtained.

Indeed, let  $V = \phi(v_1), \dots, \phi(v_K)$  be  $K$  centers obtained by K-means without any constraint and let  $W = \phi(w_1), \dots, \phi(w_K)$  be an orthonormal basis which spans  $V$ . The basis  $W$  can be obtained by eigen-decomposing  $VV^T$ . We have

$$\phi(v_k) = \sum_{j=1}^K \langle \phi(v_k); \phi(w_j) \rangle \phi(w_j) \quad , \forall k = 1, \dots, K \quad (11)$$

$$= \sum_{j=1}^K \kappa(v_k, w_j) \phi(w_j) \quad , \forall k = 1, \dots, K \quad (12)$$

Hence, the K-means' distortion can be rewritten as

$$\mathcal{D}(X, V) = \sum_{i=1}^l \|\phi(x_i) - \phi(v_{i_k})\|^2 \quad (13)$$

$$= \sum_{i=1}^l \|\phi(x_i) - \sum_{j=1}^K \kappa(v_{i_k}, w_j) \phi(w_j)\| \quad (14)$$

$$= \sum_{i=1}^l \left( 1 - 2 \sum_{j=1}^K \kappa(v_{i_k}, w_j) \kappa(x_i, w_j) + \sum_{j=1}^K \kappa(v_{i_k}, w_j)^2 \right) \quad (15)$$

and using the Eq.(6) the reconstruction error using the basis  $W$  is given by

$$\mathcal{E}(X, W) = \sum_{i=1}^l \left( 1 - \sum_{j=1}^K \kappa(x_i, w_j)^2 \right) \quad (16)$$

Comparing Eq.(15) to Eq.(16) we have

$$\mathcal{D}(X, V) - \mathcal{E}(X, W) = \sum_{i=1}^l \sum_{j=1}^K (\kappa(x_i, w_j)^2 - 2\kappa(x_i, w_j)\kappa(v_{i_k}, w_j) - \kappa(v_{i_k}, w_j)^2) \quad (17)$$

$$= \sum_{i=1}^l \sum_{j=1}^K (\kappa(x_i, w_j) - \kappa(v_{i_k}, w_j))^2 \quad (18)$$

The right hand side of the last equality is clearly non negative, leading to the conclusion that

$$\mathcal{D}(X, V) \geq \mathcal{E}(X, W) \quad (19)$$

This inequality means that the reconstruction error of the subspace formed by the basis  $W$  which span the  $K$  centers  $V$  is upper bounded by the distortion of the K-means objective function.

### 2.3 Kernel K-Means and KPCA

Recall that the KPCA algorithm search for a basis  $W^*$  which minimizes the reconstruction error, that is:

$$W^* = \arg \min_W \mathcal{E}(X, W) \quad (20)$$

The result from last section suggests that instead of searching for such basis directly, we might as well search for centers  $\hat{V}$  minimizing the K-means distortion:

$$\hat{V} = \arg \min_V \mathcal{D}(X, V) \quad (21)$$

then form the subspace with the orthonormal basis  $\hat{W}$  which span these centers  $\hat{V}$ . Indeed,  $\mathcal{E}(X, \hat{W}) > \mathcal{E}(X, W^*)$  from the Eq.(20) and using the Eq.(19) we have

$$\mathcal{D}(X, \hat{V}) \geq \mathcal{E}(X, W^*) \quad (22)$$

This is the basic idea of the training algorithm for KPCA proposed in next section.

## 3 Training Algorithm for Sparse KPCA

From the result presented in previous section, we will train the KPCA based on K-means clustering's result on the feature space. While the idea of this algorithm is quite simple, the classical kernel K-means requires storage of all training data in memory. In order to simplify the result of K-means in this feature space we will use the preimage reconstruction algorithm. Section 3.1 presents the application of the reconstruction algorithm for a Gaussian kernel as proposed by Scholkopf et al. in [5] to the kernel K-means. Section 3.2 presents the overall procedure of our training method for KPCA.

### 3.1 Kernel K-Means and Preimage Algorithm

The kernel K-means algorithm, like classical K-means, is an iterative procedure which repeatedly reassigns each point to the closest cluster and recomputes clusters' center. The cluster assignment is based on the Euclidean distance. On the feature space, the Euclidean distance between cluster center  $\phi(v_k)$  and a mapped training data  $\phi(x_i)$  is given by:

$$\|\phi(x_i) - \phi(v_k)\|^2 = \kappa(x_i, x_i) - 2\kappa(x_i, v_k) + \kappa(v_k, v_k) \quad (23)$$

$$= 2 - 2\kappa(x_i, v_k) \quad (24)$$

In order to recompute the new cluster's center, we will use the preimage reconstruction algorithm for a Gaussian kernel [5]. Indeed, the objective of preimage reconstruction algorithm is to find the vector  $z$  in the input space such that  $\phi(z)$  best approximate the vector  $\Psi \in \mathcal{F}$  of the form

$$\Psi = \sum_{i=1}^l \alpha_i \phi(x_i) \quad (25)$$

For Gaussian kernel,  $z$  can be found by an iterative procedure whose update equation is given by:

$$z^{t+1} = \frac{\sum_{i=1}^l \alpha_i \kappa(x_i, z^t) x_i}{\sum_{i=1}^l \alpha_i \kappa(x_i, z^t)} \quad (26)$$

where  $z^t$  is the current estimate of solution  $z$  and  $z^{t+1}$  is its new value. This iterative update is conducted until convergence. For our problem, the updating rule for recompute the new cluster's center is given by:

$$v_k^{t+1} = \frac{\sum_{i: i_k=k} \kappa(x_i, v_k^t) x_i}{\sum_{i: i_k=k} \kappa(x_i, v_k^t)}, \quad k = 1, \dots, K \quad (27)$$

The overall algorithm is depicted in Algorithm 1. This algorithm enjoys both the non-linearity of the kernel function and the simplicity of K-means algorithm. Moreover, it should be noted that, even if the preimage does only approximate the vector in the feature space, it does not deteriorate the performance of the kernel-based technique in practice. For example, experimentally we have found that by replacing the data centering in the feature space via averaging the kernel evaluation [2] by using the preimage approximation of center, the resulting KPCA allows better reconstruction error with faster computation.

### 3.2 K-Means Based KPCA Training Algorithm

The proposed training algorithm for sparse KPCA is then

1. Run preimage-based kernel K-means clustering to produce  $K$  centers,  $\phi(v_1), \dots, \phi(v_K)$
2. Form a kernel matrix  $\mathbf{K}$  with these  $K$  vectors:  $\mathbf{K}_{ij} = \kappa(v_i, v_j)$

**Algorithm 1.** Preimage-based Kernel K-means algorithm.

- 
1. Set  $t = 0$  and initialize the  $K$  centers  $\phi(v_1^t), \dots, \phi(v_K^t)$
  2. For each point  $\phi(x_i)$ ,  $i = 1, \dots, l$  find  $i_k = \arg \min \|\phi(x_i) - \phi(v_k^t)\|^2$  using Eq.(23)
  3. Recompute clusters' center by applying the preimage algorithm. For our problem, the update rule is given by:

$$v_k^{t+1} = \frac{\sum_{i_k=k} \kappa(x_i, v_k^t) x_i}{\sum_{i_k=k} \kappa(x_i, v_k^t)}$$

4. If not converged, set  $t = t + 1$  and go to Step 2; Otherwise, stop.
- 

3. Compute an orthonormal basis by eigen-decompose  $\mathbf{K}$  and normalize them;

$$\phi(w_j) = \sum_{k=1}^K \frac{u_{jk}}{\sqrt{(\lambda_j)}} \phi(v_k) \quad j = 1, \dots, K$$

- where  $u_j, \lambda_j$ ,  $j = 1, \dots, K$  are eigenvector and eigenvalue of  $\mathbf{K}$ . In fact, the step 2 and 3 realize the classical KPCA on the data  $v_1, \dots, v_K$ .
4.  $\phi(w_1), \dots, \phi(w_K)$  form the desired principal space. We may also run preimage algorithm to explicitly reconstruct  $w_j$   $j = 1, \dots, K$ . Noted that it is also possible to select fewer principal axes than the number of clusters.

## 4 Experimental Results

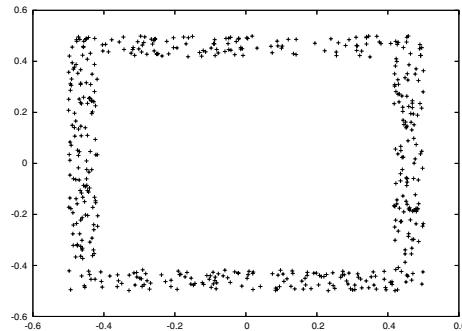
In this section, we present 2 experimental results using the proposed sparse KPCA algorithm. The first one is a toy problem which aims at comparing this sparse KPCA to the classical KPCA algorithm. The second experiment concerns the well-known MNIST database. This is to illustrate the use of our method in large scale database.

### 4.1 Toy Problem

For the first experiment, we compared the proposed sparse KPCA to classical KPCA on an artificial dataset of 800 points shown in Figure 1. In this experiment, we project all data onto a principal subspace of the feature space. The average reconstruction error from both algorithm is shown in Table 1.

**Table 1.** Reconstruction error from classical KPCA and the proposed K-means based sparse KPCA algorithm

	$\sigma^2 = 10$	$\sigma^2 = 20$	$\sigma^2 = 100$
Classical KPCA	0.00679351	0.00348327	0.000711866
K-means based KPCA	0.00698684	0.00355881	0.000726138



**Fig. 1.** Artificial dataset used in the first experiment

In this experiment, we use 4 first principal axes. Noted that for the classical KPCA, we did use the preimage approximation of data center to speed up the training and test time. Experimentally, this does also slightly improved the reconstruction error compared to the original data centering in feature space [2]. From these results, we can see that the K-means based KPCA achieve almost the same accuracy as the classical KPCA but with very smaller memory requirement. Indeed we need only 4 vectors to represent 4 principal axes for the proposed KPCA while the classical KPCA needs 800 vectors of the training set plus other 3200 real coefficients.

## 4.2 MNIST Database

MNIST is a well known offline handwritten database. This database contains 60000 training samples and 10000 testing samples of 10 digits. These samples were size-normalized and centered in a 28x28 gray scale image. For our experiment, we resize all image samples to 16x16 pixels. It should be noted that this database has been thoroughly evaluated by many researchers and PCA-based features is clearly not the best features in comparison to other features like the gradient-based features or chain-code for example [13]. Our objective here is only to illustrate the application of the proposed training algorithm on a large scale real world database.

The 1-NN classifier was used to evaluate the features obtained from the proposed sparse KPCA as well as the features from classical PCA. In this experiment, the KPCA was trained by first applying the kernel K-means with 200 clusters, then the first 50 principal axis which span these centers were selected. For the classical PCA, the first 50 principal axis were also selected.

For this test, we obtain 94.19% accuracy using PCA-based features. For KPCA, the variance of Gaussian kernel was experimentally set to 10x16x16 which allows reaching 94.32% accuracy. Even if this is only a slight improvement, it does confirm the applicability of our approach to a very large database. In this experiment, the training time of the proposed sparse KPCA is not very higher than that of classical PCA. On a 2 GHz Pentium 4, the PCA's running time

is about 5.76 min while the proposed sparse KPCA's running time is about 6.90 min. We believe that this running time could be improved for example by using the kernel cache like in SVMlight [14] or libsvm [15].

## 5 Conclusions and Future Works

This work concerns the simplification of KPCA algorithm both in the training procedure and the resulting model. We have studied the link between K-means algorithm and the reconstruction error which is the objective function of the PCA in the feature space induced by Gaussian kernel. This relation is similar to the one proposed in [12], but obtained with different derivation. Future works include the study of the compactness of the proposed bound (Eq.(22)). From this bound, we proposed a KPCA's training algorithm based on kernel K-means clustering result. A disadvantage of the approach is that the result depends on the initialization setting needed for the kernel K-means algorithm and there is still an open question about the number of clusters needed. However we believe this could be solved from clustering perspective, for example by using some split-and-merge clustering strategy. Furthermore, since clustering scales better to large database than the eigen-decomposition, the relation between these 2 apparently different problems could lead to other efficient training algorithms for KPCA. In addition, the application of preimage algorithm does not only simplify the kernel K-means algorithm used in the training procedure but also reduce the computational cost during the running time and the storage memory. The comparison between the proposed method and other techniques like the greedy selection [8] or the kernel Hebbian algorithm [9] is also currently under investigation.

## References

1. Boser, B., Guyon, I., Vapnik, V.: An training algorithm for optimal margin classifiers. In: In Fifth Annual Workshop on Computational Learning Theory. (1992) 144–152
2. Scholkopf, B., Smola, A., Muller, K.R.: Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation* **10** (1998) 1299–1319
3. Mika, S., Ratsch, G., Weston, J., Scholkopf, B., Muller, K.R.: Fisher discriminant analysis with kernels. *Neural Networks for Signal Processing IX* (1999) 41–48
4. Burges, C.J.: Simplified support vector decision rule. In: International Conference on Machine Learning. (1996) 71–77
5. Schölkopf, B., Mika, S., Burges, C.J., Knirsch, P., Müller, K.R., Rätsch, G., Smola, A.J.: Input space versus feature space in kernel-based method. *IEEE Transaction on Neural Networks* **10**(5) (1999) 1000–1017
6. Tipping, M.E.: Sparse kernel principal component analysis. In: Neural Information Processing Systems (NIPS). (2000) 633–639
7. Achlioptas, D., Frank McSherry, B.S.: Sampling techniques for kernel methods. In: Neural Information Processing Systems (NIPS). (2002)

8. Franc, V., V.Hlavac: Greedy algorithm for a training set reduction in the kernel methods. In: Computer Analysis of Images and Patterns: Proceedings of the 10th International Conference, Groningen, The Netherlands, Springer (2003) 426–433
9. Kim, K.I., Franz, M.O., Schölkopf, B.: Iterative kernel principal component analysis for image modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **27**(9) (2005) 1351–1366
10. Zha, H., He, X., Ding, C., Simon, H., Gu, M.: Spectral relaxation for k-means clustering. In: Neural Information Processing Systems (NIPS). (2001)
11. Dhillon, I.S., Guan, Y., Kulis, B.: Kernel k-means, spectral clustering and normalized cuts. In: International Conference on Knowledge Discovery and Data Mining (KDD). (2004)
12. Ding, C., He, X.: K-means clustering via principal component analysis. In: International Conference on Machine Learning, Banff, Canada (2004)
13. Liu, C.L., Nakashima, K., Sako, H., Fujisawa, H.: Handwritten digit recognition: benchmarking of state-of-the-art techniques. *Pattern Recognition* **36**(10) (2003) 2271–2285
14. Joachims, T.: Making large-scale svm learning practical. In Schölkopf, B., Burges, C., Smola, A., eds.: *Advances in Kernel Methods - Support Vector Learning*. MIT Press (1999)
15. Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines. (2001) Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.

# Clustering-Based Relevance Feedback for Web Pages

Seung Yeol Yoo and Achim Hoffmann

School of Computer Science and Engineering  
University of New South Wales Sydney 2052, NSW, Australia  
`{syy, achim}@cse.unsw.edu.au`

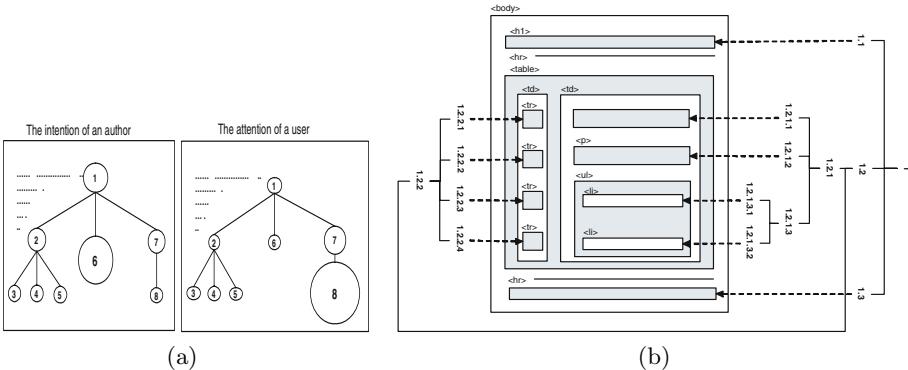
**Abstract.** Most traditional relevance feedback systems simply choose the top ranked Web pages as the source of providing the weights of candidate query expansion terms. However, the contents of such top-ranked Web pages is often composed of heterogeneous sub-topics which can be and should be recognized and distinguished. However, current approaches treat retrieved Web pages as one unit and often fail to extract good quality candidate query expansion terms.

In this paper, our basic idea is that the Web pages properly clustered into a sub-topic cluster can be used as a better source than whole given Web pages, to provide more topically coherent relevance feedback for that specific sub-topic. Thus, we propose **Clustering-Based Relevance Feedback for Web Pages**, which utilizes three methods to cluster retrieved Web pages into several subtopic-clusters. These three methods cooperate to construct good quality clusters by respectively supporting Web page Segmentation, Term Selection,  $k$  Seed Centroid Selection. Here, the automatically selected terms indicate the relevance feedback to construct all sub-topic clusters and assign the given Web pages to proper clusters. Each subset of the selected terms, which occurs in the Web pages assigned into a sub-topic cluster, indicates the relevance feedback to expand a query over that sub-topic cluster. Our experimental results showed that the clustering performances based on two traditional term-weighting methods (i.e., an unsupervised method and a supervised method) can be significantly improved with our methods.

## 1 Introduction

While the Web represents a vast information resource, the entire content in each Web page might not always be valuable to support a user's specific information needs. On the same Web page, the author of the Web page usually puts a lot of contents, which belong to different subtopics [1]. Thus, the content should be disassembled into smaller elements (e.g., segments) than Web pages. Such finer-grained elements should be well organized to be properly accessible according to a user's specific information needs.

To achieve such purposes, we need to consider the commonality and the difference between two views; the author's view (i.e., intention) described in a Web page and a user's view (i.e., attention) of the Web page. For example, in



**Fig. 1.** (a) Two different views on the same Web page: the intention of an author and the attention of a user. Each segment is illustrated by a node, and the size of a node indicates the relative importance value against the other segments in the Web page. The hierarchically connected edges indicate the topical connections between segments (i.e., the narration structure. We call it ‘segment-topic-paths’). (b) An example of discovered multiple topic-paths of hierarchically connected segments.

(a) of Figure 1, a user similarly recognized the topical streams intended by the author of a Web page, but the topical importance value of each segment was different to the user from that intended by the author. We call the author’s view ‘objective basic view’ and a user’s view ‘subjective query view’. The ‘subjective query view’ can change the importance value of each segment/segment-topic-path according to the user’s topical interest over the Web page.

To identify ‘objective basic view’ of each Web page, we developed a segmentation algorithm ‘Layout-based Web Page Segmentation Algorithm’ which can run as pre-processing of any query session and partition each Web page into several segment-topic-paths. It is independent of the semantics of not only the contents (excluding html-tags) of the Web page but also the query-words provided by users, as this algorithm is based on generic structure-patterns among html-tags.

Figure 1. (b) shows an example of discovered multiple segment-topic-paths from a Web page. The discovered subtopic-paths implicate that the story in the segment {1.2.1.3} might be more relevant to the story of its parent-segment {1.2.1} or sub-segments {1.2.3.1} and {1.2.3.2} than to that of the segment {1.2.2} which is not hierarchically connected with the segment {1.2.1.3} through any segment-topic-paths.

To reflect a user’s information needs, we also developed a novel user interface that can support the user through a query-session. However, in this paper, we omitted the details. The goal of this paper is to show that the ‘objective query view’ can contribute to discover more coherent query expansion terms. Thus, in this paper, we would have two assumptions as follows: 1) user’s topic interests are considered as targeted categories which our system is expected to generate with given Web pages, and 2) the user-selections of expansion terms are simulated by an unsupervised or a supervised term selection method.

The rest of this paper is organized as follows: Related works are presented in Section 2. Section 3 describes our approach utilizing three methods which select more coherent expansion terms from the given Web pages and then utilize the selected expansion terms for constructing Web pages clusters. Section 4 describes the set up of our experiments. Section 5 presents our experimental results showing that the average quality of clusters constructed by our approach is better than that of clusters constructed by  $k$ -means clustering based on two traditional terms selection methods. Section 6 presents the conclusions.

## 2 Related Works

As noise terms and multi-topics are two major negative factors for expansion terms selection in a relevance feedback technique, it is necessary to *segment a web page into semantically related units* so that irrelevant information and misleading terms can be filtered out and multiple topics can be distinguished. For example, [1,2,3,4,5,6] using segmentation method showed that it is necessary to filter out noisy segments and detect important segments from Web pages for the improvement of web information retrieval. However, their works did not touch the second negative factor to provide focusing on specific subtopics within Web pages by considering topically distinguishable boundaries within Web pages.

The Vision-based Page Segmentation (VIPS) method [7] is suggested to supplement the Dom-based segmentation method<sup>1</sup> e.g., [4] with visual cues. For example, [6] works in this way: Firstly they calculated the relevance values among content-elements according to their visual adjacency and/or appearance similarity. And then, they applied such relevance values to a few heuristic rules (i.e., a set of rules identifying the neighborhood relationship among content-element) to group relevant content-elements in a Web page into several segments.

[6] expected that some visual cues can aggregate more semantic information which is difficult to directly extract from html-tags. However, visual cues have some problems such as 1) how to choose the proper size of the unit content-element and decide generic adjacency-conditions between two content-elements. For example, the adjacency between two content-elements can be affected by the size of intermediate content-element(s), and 2) the decision about the total number of visually distinguishable segments has significant effects on the sizes of the chosen content-elements and the interpretation of those content-elements' semantic relationships. However such decisions cannot be predefined without considering the content of each Web page, and no satisfactory solution has been found as yet.

Another criticism of the VIPS method for discovering segments and their structure are as following: it does not distinguish two segments which describing different subtopics in similar layouts (e.g., similar size, length and position). According to our observations on Web pages, segments enumerating different subtopics are frequently presented with similar layouts in a Web page (e.g., in a portal site).

---

<sup>1</sup> Extracting a tag tree based on the html tags.

### 3 Relevance Feedback Based on Web Pages Clustering

#### 3.1 Layout-Based Segmentation

The goal of layout-based segmentation of a Web page is to discover more precise narration structure intended by the author of the Web page.

For the reasons mentioned above, we have selected a way to extract Generic Nested-Structure Patterns (GNSPs) rather than utilizing frail visual cues from a html-source to interpret the structural semantics in the html-source. Our GNSPs extraction algorithm supplements the limitations [6] of the Dom-based segmentation method with nest-based hierarchy relationships between html-tags. We believe that the narration structure intended by an author can be more precisely extracted with the help of GNSPs rather than visual cues, because GNSPs can naturally: 1) bound each content-element according to the html-tags formating it, 2) construct segment-topic-paths identifying the adjacency between content-elements, and 3) determine the number of distinguishable segments and segment-topic-paths of a Web page.

The GNSPs extraction algorithm was performed as a query pre-processing task not to obstruct the run-time performance of an information retrieval system as follows: 1) remove noisy contents such as script-codes, 2) analyze the generic hierarchy structure relationships among html-tags. For example, the listing information (e.g., `<ul>`, `<ol>`, `<li>`, `<select>`, `<option>`, etc.), the block information (e.g., `<table>`, `<tr>`, `<td>`, etc.), the boundary information (e.g., `<hr>`, `<h#>`, `<thead>`, `<tbody>`, `<tfoot>`, `<div>`, `<p>`, `<br>`, etc.), and the priority information (i.e., some tags should be considered with higher priority than the other tags. E.g., a `<div>` tag occurring between `<table>` and `</table>` tags should separate only the content between `<table>` tag and `</table>` tag rather than the whole contents of the Web page). Further details of GNSPs extraction algorithm are omitted due to limited space.

#### 3.2 Term Selection Methods

The performance of clustering task depends on the quality of terms extracted with a terms selection method. In this subsection, we would introduce two traditional methods i.e., “ $\chi^2$  statistic method (CHI)” and “Term Contribution method (TC)”, and propose a new method “Term Context Contribution based on Segments-Topic-Paths (TCC-STP)”.

**$\chi^2$  Statistic (CHI).** The  $\chi^2$  statistic is a supervised method which measures the statistical significance of association between a term and a category by using known class label information. The statistical significance can be defined to be  $\chi^2(t, c) = \frac{N \times (AD - CB)^2}{(A+C) \times (B+D) \times (A+B) \times (C+D)}$  and  $\chi^2(t) = \sum_{i=1}^m Pr(c_i) \chi^2(t, c_i)$ , where  $A$  is the number times  $t$  and  $c$  co-occur,  $B$  is the number of time the  $t$  occurs without  $c$ ,  $C$  is the number of times  $c$  occurs without  $t$ ,  $D$  is the number of times neither  $c$  nor  $t$  occurs, and  $N$  is the total number of documents [8].

**Term Contribution (TC).** Term Contribution method (TC), proposed as an unsupervised method in [9], considers the different importance of a term in different documents. They define the contribution of a term in a dataset as its overall contribution to the documents' similarities. The definition for TC is  $TC(t) = \sum_{i,j \cap i \neq j} f(t, w_i) \times f(t, w_j)$ , where  $f(t, w)$  represents the  $tf * idf$  [10]. However TC does not consider different contexts of each term in different documents. For example, if a term “java” is used with terms “island” and “travel” in a document, and is used with terms “software” and “applet” in another different document, then the contribution of “java” should be separately considered for two different contexts (i.e., two different word-senses “island” and “programming language” of “java”).

**Term Context Contribution Based on Segments-Topic-Paths (TCC-  
STP).** To consider different contexts of a term in different Web pages, we introduce a new unsupervised feature selection method called “Term Context Contribution based on Segments-Topic-Paths” (TCC-STP). It gives higher weights to the terms sharing more common context-words within the same segment-topic-paths. The equation for TCC-STP is:

$$TCC - STP(t) = \sum_{i,j \cap i \neq j} f(t, w_i) \times f(t, w_j) \times \\ f(CT(i, j), STP(t, w_i)) \times f(CT(i, j), STP(t, w_j)),$$

where  $STP(t, w)$  represents segment-topic-paths, in which  $t$  is occurred, of Web page  $w$ .  $CT(i, j)$  denotes context-words co-occurring in  $STP(t, w_i)$  and  $STP(t, w_j)$ .  $f(CT(i, j), STP(t, w_i))$  represents the frequency of  $CT(i, j)$  in  $STP(t, w_i)$ . We simply set  $f(CT(i, j), STP(t, w_i))=1$  and  $f(CT(i, j), STP(t, w_j))=1$ , when  $|CT(i, j)|=0$

### 3.3 Applied K-Means Clustering (AKC)

To select more coherent expansion terms, we modified the original  $k$ -means clustering (**KC**) with the following two algorithms.

**Initial  $k$  Centroids Selection.** In  $k$ -means clustering, the number of iterations and the quality of the final clusters depend on choosing a proper number of  $k$  centroids and selecting good initial centroids (i.e., seed centroids). For experimental purposes, we assumed that  $k$  is the same as the number of original categories within the test-dataset. We selected  $k$  seed centroids in a way which maximizes the total distance of the selected seed centroids.

We define the following maximization function of total inter-centroids distance D:

$$Max D = \sum_{i=1}^{k-1} \sum_{j=i+1}^k \overline{w_{c_i} w_{c_j}}$$

where  $w_{c_i}$  and  $w_{c_j}$  represent a Web page selected as the seed centroid for cluster  $c_i$  and  $c_j$ , respectively. The distance  $\overline{w_{c_i}w_{c_j}}$  between two seed centroids can be calculated by one of the following two equations:

1)  $\overline{w_{c_i}w_{c_j}} = -\sum_{t \in T} \left\{ \frac{(f(t, w_{c_i}) + f(t, w_{c_j}))}{2} \times \frac{|T_m|}{|T|} \right\}$  if  $f(t, w_{c_i}) \neq 0$  and  $f(t, w_{c_j}) \neq 0$ , and 2)  $\overline{w_{c_i}w_{c_j}} = \sum_{t \in T} \left\{ \max(f(t, w_{c_i}), f(t, w_{c_j})) \times \frac{|T_n|}{|T|} \right\}$  if  $(f(t, w_{c_i}) = 0 \text{ and } f(t, w_{c_j}) \neq 0) \text{ or } (f(t, w_{c_i}) \neq 0 \text{ and } f(t, w_{c_j}) = 0)$ .  $T$  denotes all terms occurred in  $w_{c_i}$  or  $w_{c_j}$ .  $f(t, w_{c_i})$  is the occurrence frequency of term  $t$  in  $w_{c_i}$ .  $T_m \subset T$  and  $T_n \subset T$  is the terms respectively satisfying the if condition of those equations.  $\frac{|T_m|}{|T|}$  and  $\frac{|T_n|}{|T|}$  are used to normalize the frequency weight of each term  $t$  over the total number  $|T|$  of terms occurred within two Web pages, and give more weight to common terms and distinguishable terms, respectively. The former  $\overline{w_{c_i}w_{c_j}}$  equation reduces the distance between two seeds when they share a term  $t$ , but the later  $\overline{w_{c_i}w_{c_j}}$  equation increases the distance between two seed centroids when a term  $t$  uniquely occurs in only one centroid  $w_{c_i}$  or  $w_{c_j}$ .

Our centroid selection method achieves earlier convergence and the quality of the final clusters is higher.

**Similarity with a Seed Centroid.** When  $k$  seed centroids are decided, the clustering problem can be considered as a classification problem which assigns each Web page to the most similar seed centroid among the  $k$  seed centroids.

The given Web pages  $W \ni w_i$ , the terms  $T \ni t$  occurred in  $W$  and the occurrence frequencies  $f(w_i, t)$  form a vector space. With that vector space, traditional  $k$ -means clustering method tried to minimize total intra-cluster distance. Each intra-cluster distance is measured by the sum of terms' occurrence differences between a centroid and assigned Web pages to that centroid. Thus, total intra-cluster distance can be denoted as a following equation:  $\text{Min Distance} = \sum_{j=1}^k \sum_{w_i \in W_{c_j}} |w_i - c_j|$ , where there are  $k$  clusters  $W_{c_j}$ ,  $j = 1, \dots, k$ .  $c_j$  denotes the centroid (i.e., mean point) of the Web pages  $W_{c_j}$ .

However, for Web pages, the traditionally used euclidian-distance measure is not efficient to cluster relevant Web pages. One reason is that common terms are usually much less often than different terms between a Web page  $w_i$  and a seed centroid  $c_j$ , even if they talk about common subtopic(s). Thus, there is always a chance that a Web page sharing a few common terms and having many different terms would have a large distance value than another Web page not sharing any common terms but having very few different terms with the centroid.

To address the above issue in Web page clustering, we reformulated the distance minimization problem between centroids and given Web pages as a similarity  $S$  maximization problem.

$$\text{Max } S = \sum_{c_j=1}^k \sum_{w_i \in W_{c_j}} \{S'(w_i, c_j) - D'(w_i, c_j)\}$$

We define the similarity between a centroid  $c_j$  and a Web page  $w_i$  as the difference between Dissimilarity and Similarity. Similarity and Dissimilarity can

be denoted as following two functions:  $S'(w_i, c_j) = \sum_{t \in T} \{ (f(t, w_i) + f(t, c_j)) \times \frac{|NZ(c_j, T)|}{|T|} \}$ , if  $f(t, c_j) \neq 0$  and  $t \notin NZ(\bar{c}_j, T)$ .  $D'(w_i, c_j) = \sum_{t \in T} \{ f(t, w_i) \times \frac{|Z(c_j, T)|}{\max(1, |NZ(\bar{c}_j, T)|)} \}$ , if  $f(t, c_j) = 0$  and  $t \in NZ(\bar{c}_j, T)$ . Where  $f(t, w_i)$  and  $f(t, c_j)$  respectively denotes the occurrence frequency of term  $t$  in the Web page  $w_i$  and the centroid  $c_j$ .  $T$  denotes a set of terms occurred in a Web page  $w_i$ .  $NZ(c_j, T) \subset T$  denotes the terms appearing in both  $w_i$  and  $c_j$ .  $Z(c_j, T) \subset T$  denotes the terms appearing in  $w_i$  but not in  $c_j$ .  $NZ(\bar{c}_j, T) \subset Z(c_j, T)$  denotes the terms appearing in  $w_i$  and at least one of the other centroids excluding  $c_j$ .

## 4 Evaluation Configuration

Through the experiments, our goal is to show that TCC-STP and AKC methods mentioned in Section 3 can extract good quality query expansion terms which can indicate relevant Web pages. For this purpose, the most intuitive evaluation method might be to compare the expansion terms extracted by a system with those extracted by human subjects. However, it has some issues like that the consistency problem between different human subjects, the limited human analysis capacity and etc. Thus, we followed a reverse but automatic way comparing the qualities of obtained clusters by different query expansion term extraction methods. Better clustering performance can be achieved by using better query expansion terms and their weights as the clustering features and feature values.

**Table 1.** The dataset properties. *Dis.words* means stemmed words (i.e., lemmas). DF means the number of documents in which a *Dis.word* occurred.

DataSet	Topic Num	Docs Num	Dis. Words	Avg. Dis. Words	Avg. DF per Dis. Word
A	5	96	6,010	63	3
B	21	333	13,833	42	6
C	12	369	13,031	35	6

### 4.1 Standard Clusters for Web Pages

To test the clustering performance, we collected three datasets from Yahoo! Directory (Downloading prohibited, ASP, link-broken or multi-framed Web pages are removed from each category). The information about the datasets is shown in Table 1. Dataset A, B and C is respectively relevant to “Robotic”, “Computer Science” and “Computer and Internet” category in the Yahoo! Directory, and respectively includes 5, 21 and 12 subcategories. We considered each subcategory as one standard cluster.

### 4.2 Evaluation Measures

Entropy and F-measure were used to evaluate the clustering performance. Those measures of a cluster respectively indicates the uniformity and the weighted harmonic mean of precision and recall of assigned Web pages to the cluster.

The Entropy of all obtained clusters is defined by the weighted sum of the entropy of each obtained cluster, as following:  $Entropy = \sum_{k=1}^{C'} \frac{|W_k|}{N} \sum_{j=1}^C p_{jk} \times \log(p_{jk})$ , where  $p_{jk} = \frac{1}{|W_k|} |\{w_i | label(w_i) = c_j\}|$  is the entropy of a obtained cluster.  $N$  is the total number of Web pages in the tested original clusters.  $C'$ ,  $C$  denote the number of obtained clusters and the number of original clusters respectively.  $W_k$  is the set of Web pages in a obtained cluster.  $label(w_i)$  returns the cluster-label of the Web page  $w_i$ . The cluster-label of a generated cluster is decided by the original cluster-label of the most shared Web pages in the generated cluster.

The Precision of a obtained cluster  $W (\ni w_i, i = 1, \dots, |W|)$  is defined as following:  $Precision(W) = \frac{1}{|W|} \max(|\{w_i | label(w_i) = c_j\}|)$ . The Recall of  $W$  is defined as following:  $Recall(W) = \frac{1}{|O|} \max(|\{w_i | label(w_i) = c_j\}|)$ , where  $label(O) = c_j$ .  $|O|$  denotes the number of Web pages which have cluster-label  $c_j$ , where  $O \subset W$ . The traditional F-measure (i.e., F1 measure; recall and precision are evenly weighted) of a obtained cluster is defined as following:  $F(W) = \frac{2 \times Precision(W) \times Recall(W)}{Precision(W) + Recall(W)}$ . The sum of F-measures of all obtained clusters  $C'$  were averaged by the number of all obtained clusters.

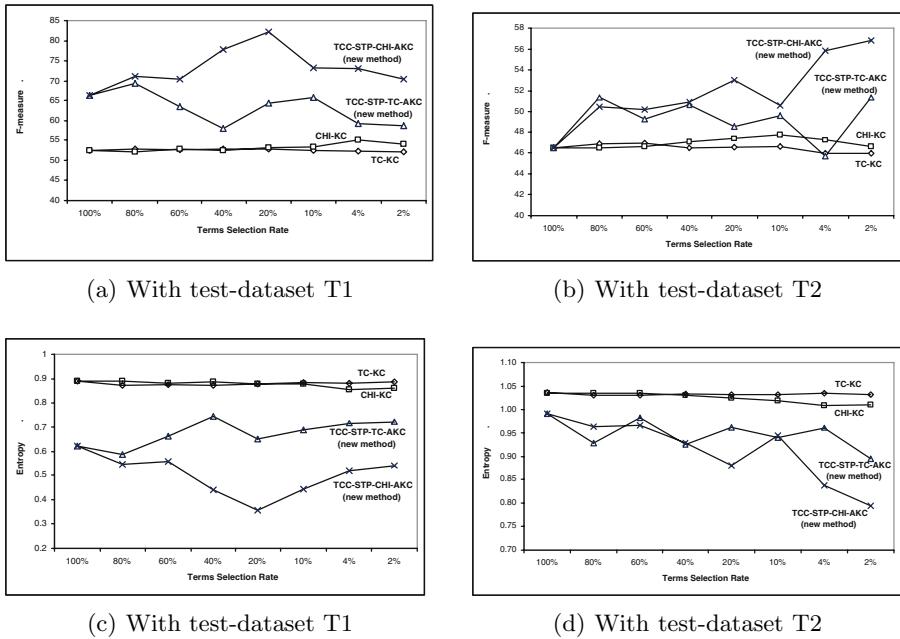
## 5 Experimental Results and Discussion

We performed clustering tasks over two test-datasets T1 and T2 with 4 different configurations and 8 different terms selection rates. The test-dataset T1 has 10 sub-datasets, and each sub-dataset consists of randomly selected one category from each original dataset A, B and C, and is composed of average 50 Web pages. The test-dataset T2 has 10 sub-datasets and each sub-dataset consists of 3 to 7 categories which have ancestor and descendant relationship each other within a original dataset A, B or C, and is composed of 84 Web pages.

The tested four different configurations are named as following: **TC-KC**(Term Contribution(TC) + K-means Clustering(KC)), **CHI-KC**( $\chi^2$  statistic(CHI) + K-means Clustering(KC)), **TCC-STP-TC-AKC**(Term Context Contribution based on Segments-Topic-Paths(TCC-STP) + Term Contribution(TC) + Applied  $k$ -Means Clustering(AKC)), and **TCC-STP-CHI-AKC**(Term Context Contribution based on Segments-Topic-Paths(TCC-STP) +  $\chi^2$  statistic(CHI) + Applied K-means Clustering(AKC)). Our methods TCC-STP and AKC (see 3.2 and 3.3) are applied to TC-KC and CHI-KC configurations.

As it is known that the results of K-means clustering (KC) depends also on the choice of the initial seed centroids, we averaged the results over 10 different seed selections. Compared to this, the initial  $k$  centroids of our Applied K-means Clustering (AKC) were decided as  $k$  Web pages which maximize the distances between them, without any needs to consider the variance among possible clustering results (see 3.3 for more details).

(a) and (b), and (c) and (d) in Figure 2 respectively show the F-measure values and Entropy values of obtained clusters over test-dataset T1 and T2. The average clustering performance based on two traditional term-weighting



**Fig. 2.** Average F-measures and Average Entropies of clusters, which are achieved with different top-ranked query expansion terms

methods the unsupervised method TC and the supervised method CHI was respectively improved by 21%(from 52% to 63%) and 41%(from 53% to 75%) for the F-measure and by 24%(from 0.88 to 0.67) and 48%(from 0.88 to 0.5) for the Entropy, over the test-dataset T1. Over the test-dataset T2, the average F-measure value improved by 6%(from 46% to 49%) and 8%(from 47% to 51%) for the F-measure and by 8%(from 1.03 to 0.95) and 12%(from 1.03 to 0.91) for the entropy. These improvements indicate that many noisy terms (e.g., high weighted for whole given Web pages but is not for the Web pages assigned to a specific cluster) were removed by our TCC-STP and AKC methods.

With our new methods TCC-STP and AKC methods, the best performance is achieved by the TCC-STP-CHI-AKC configuration, and it might be due to the benefits of using known class label information to extract good quality query expansion terms. Even though TCC-STP-CHI-AKC and CHI-KC utilized the same class-label information for clustering tasks, TCC-STP-CHI-AKC showed better performance with our TCC-STP and AKC methods. We also would like to concentrate on the performance of the TCC-STP-TC-AKC configuration which get the second best performance. TCC-STP-TC-AKC is a new unsupervised method for term selection. In practice, the class labels of given Web pages cannot be known before any query sessions, thus TCC-STP-TC-AKC might be the best solution to extract good quality query expansion terms among those four

configurations. The F-measure values and Entropy values of TCC-STP-TC-AKC outperformed those of CHI-KC which is based on the supervised method CHI.

## 6 Conclusion

In this paper, the experimental results showed that our methods can significantly improve the performance of clustering tasks based on the supervised term-weighting method CHI and the unsupervised method TC.

Further developments of our approach will investigate the possibility of interactive query-expansion. Iteratively, the user can select some of the query expansion terms to refine their query expression and then start a new clustering process over the clustered Web pages under the context of the refined query expression.

## References

1. Yin Liu, W.L., Jiang, C.: User interest detection on web pages for building personalized information agent. In: Proceedings of the 5th international Conference on Web-Age Information Management, WAIM 2004. (2004) 280–290
2. Deng Cai, Shipeng Yu, J.R.W., Ma, W.Y.: Block-based web search. In: Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR-2004. (2004) 456–463
3. Deng Cai, Shipeng Yu, J.R.W., Ma, W.Y.: Extracting content structure for web pages based on visual representation. In: Proceedings of the 5th Asia Pacific Web Conference, APWeb 2003. (2003) 406–417
4. Suhit Gupta, Gail Kaiser, D.N., Grimm, P.: Dom-based content extraction of html documents. In: Proceedings of the 12th International World Wide Web Conference, WWW 2003. (2003) 207–214
5. Ruihua Song, Haifeng Liu, J.R.W., Ma, W.Y.: Learning block importance models for web pages. In: Proceedings of the 13th International World Wide Web Conference, WWW 2004. (2004) 203–211
6. Shipeng Yu, Deng Cai, J.R.W., Ma, W.Y.: Improving pseudo-relevance feedback in web information retrieval using web page segmentation. In: Proceedings of the 12th International World Wide Web Conference, WWW2003. (2003) 11–18
7. Cai, D., Yu, S., Wen, J.R., Ma, W.Y.: Vips: a vision-based page segmentation algorithm. Technical report, Microsoft (2003)
8. Yiming Yang, J.O.P.: A comparative study on feature selection in text categorization. In: Proceedings of the 14th International Conference on Machine Learning, ICML 1997. (1997) 412–420
9. Liu, T., Liu, S., Chen, Z., Ma, W.Y.: An evaluation on feature selection for text clustering. In: Proceedings of the Twentieth International Conference on Machine Learning, ICML 2003. (2003) 488–495
10. Salton, G.: Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer. Addison-wesley, Boston, MA, USA (1989)

# Building Clusters of Related Words: An Unsupervised Approach

Deepak P<sup>1,\*</sup>, Delip Rao<sup>2</sup>, and Deepak Khemani<sup>2</sup>

<sup>1</sup> Services Innovation Research Center, IBM India Research Lab, Bangalore

<sup>2</sup> Dept of Computer Science and Engineering, IIT Madras, Chennai  
deepakswallet@gmail.com, delip@cse.iitm.ernet.in,  
khemani@iitm.ac.in

**Abstract.** The task of finding semantically related words from a text corpus has applications in - to name a few - lexicon induction, word sense disambiguation and information retrieval. The text data in real world, say from the World Wide Web, need not be grammatical. Hence methods relying on parsing or part-of-speech tagging will not perform well in these applications. Further even if the text is grammatically correct, for large corpora, these methods may not scale well. The task of building semantically related sets of words from a corpus of documents and allied problems have been studied extensively in the literature. Most of these techniques rely on the usage of part-of-speech or parse information. In this paper, we explore a less expensive method for finding semantically related words from a corpus without parsing or part-of-speech tagging to address the above problems. This work focuses on building sets of semantically related words from a corpus of documents using traditional data clustering techniques. We examine some key results and possible applications of this work.

## 1 Introduction

This study focuses on building sets of semantically related words from a corpus of documents using traditional data clustering techniques. The task of building semantically related sets of words from a corpus of documents and allied problems have been studied extensively in the literature. Most of these techniques stem from the Computational Linguistics community and many involve parsing of the documents. We represent each word as a vector which reflects the distribution of occurrences of the same word in the different documents. Semantically related sets are derived by means of clustering of these word vectors. Our work presents a significant departure from the earlier literature in dealing with the problem. Firstly, we do not make use of any parsing or part of speech tagging techniques and represent each document as just a bag of words, a common representation in the information retrieval and data mining community. Secondly, we attempt to use the information based on the document frequencies of the words whereas the traditional approaches

---

\* This work was done when the author was at IIT Madras, Chennai.

have treated the entire corpus of documents as a collection of sentences. Once again, document frequency has been put to good use and comes from the data mining community [1]. Thirdly, given that our aim is to build collections of words, we use the k-means clustering algorithm instead of hierarchical agglomerative clustering which has been the popular choice in literature.

Section 2 reviews related work in the area. Section 3 puts down the motivation behind our work. The experimental methodology is detailed in Section 4 with the results of the experiments summarized in Section 5. Section 6 lists the conclusions derived from the experiments. Some possible application areas for semantically related word clusters are discussed in Section 7. Section 8 summarizes the contributions of this work and lists pointers for future work.

## 2 Related Work

One among the earliest works which focuses on a related problem [3] talks about identifying the ranked list of similar nouns, given a noun in the corpus. It introduces the concept of mutual information for a noun-verb pair and extends it to define a similarity measure for every noun-noun pairs. The more similar they are according to the similarity measure, the more semantically related, they are expected to be. [6] addresses the problem of clustering words to find sets of similar words. Words are represented by the relative frequency distributions of contexts in which they appear, and relative entropy is used to measure the dissimilarity of those distributions. A soft hierarchical clustering of data is done to get the relationships between the words. [2] describes a methodology to create a thesaurus from a given corpus. From each sentence, they derive triples of the type  $\langle w_1, r, w_2 \rangle$  which indicates that word  $w_1$  is related to word  $w_2$  by the relationship  $r$ . Such triples are used to derive a similarity measure for word-pairs which quantifies the confidence that a word describes another. Hierarchical Agglomerative Clustering is done to get a tree structure to represent the entire thesaurus. [4] addresses the problem of finding the hypernyms of a particular noun (similar to [2]). The similarity measure of a noun pair is parameterized only by the number of times the nouns co-occur in a conjunction or appositive with the other in contrast to [2]. Hierarchical agglomerative clustering is done to obtain a tree which is further used to find hypernyms. As can be seen, most of the work in this regard comes from the computational linguistics community. Finding ‘semantic relationship’ is almost always considered as a problem which involves parsing and exploitation of co-occurrence information. Further, all the methods referred to above, use hierarchical agglomerative clustering to build a measure of semantic relationship between words. [21] uses LSA (Latent Semantic Analysis) and PoS tag information for finding related words. LSA uses Singular Value Decomposition (SVD), a dimension reduction technique, which brings related words closer in the reduced dimension. While it is possible to use LSA for clustering related words, it has been used in conjunction with PoS tag information.

### 3 Motivation

Clustering is a very popular technique in the data mining community and has been applied to document collections to find clusters of similar documents. It has been shown [8] that standard k-means clustering works better than hierarchical techniques for document clustering. k-means has been very popular in the text mining community and many variants have evolved over time (of which most of them try to incorporate semi-supervision) such as COP-k-means [9], Seeded k-means and Constrained k-means [10]. Given that text clustering using k-means has worked well, it is obvious that it has been able to infer the semantic relationship between the different documents. We outline a very standard methodology for extracting the vectors from the text corpus and try to put forward our motivation in a simple and intuitive manner.

Given a corpus, the text clustering task usually starts off with building the term document matrix which has as many rows as the number of documents and as much columns as the number of words. Each entry in the matrix indicates the number of times the corresponding word has occurred in the corresponding document. Each row corresponds to the TF (term frequency) vector of the particular document. Further techniques to process the matrix involve normalization of each document vector to add up to a constant, whereby we get the normalized TF (nTF) vector. An additional step of Inverse Document Frequency Weighting may be incorporated before normalization, whereby we get the normalized TF-IDF vector. Given the TF, nTF or nTFIDF vectors of the documents, clustering is a straightforward task. Having outlined the document clustering task, an analogous method of term clustering is not very difficult to perceive. In the term document matrix, each column corresponds to a term and the transpose can be used as a Document Frequency (DF) vector, whereas a normalized version of the DF vector could analogously be termed the nDF vector.

Given that the clustering of TF, nTF and nTFIDF vectors do aid discovering semantic relationships among documents, we argue that the clustering of the DF, nDF and nDFITF vectors would aid discovering semantic relationships among the words. More abstractly, we argue that if the clustering of the rows of the term document matrix is useful, clustering of the columns of the term-document matrix can't be useless. Having put forward our motivation, we go forward to verify and quantify the utility of such an approach.

### 4 Experiment Methodology

#### 4.1 Dataset

We used the Time corpus [11], a popular dataset in the Information Retrieval and Data Mining communities which consists of 423 articles published by the Time magazine during the cold-war period (1960s). The corpus consists of documents which have comparable lengths which is a desirable property for our experiments. The entire dictionary of words in the corpus, after stop-word removal and stemming, is of size 20000.

## 4.2 Vectors Used

As already mentioned in section 3, we represent each term by the corresponding column vector from the term document matrix (whose elements are term frequencies in a document). The raw column vectors normalized so that each vector has elements summing up to unity from the term document matrix are hereafter referred to as the normalized Document Frequency (nDF) vectors. We use the set of nDF vectors in our experiments.

## 4.3 Clustering Methodology

The work makes no assumption on the type of clustering methodology to use. However, for the purpose of this experiment we demonstrate results using k-means clustering. The sets of nDF vectors were clustered using the k-means algorithm. k-means takes the number of clusters (to be generated in the output) as a parameter. Given that we do not have any knowledge of the number of clusters that exist, we tried out different values of k. The values chosen were 10, 20, 50 and 100. The actual k-means clustering was done using the WEKA Toolkit for Data Mining [12] developed by the University of Waikato, New Zealand.

## 4.4 Evaluation of Clusters

With a dictionary size of 20000, the average cluster size is of the order of hundreds of words. Evidently, it is difficult to manually verify the goodness of the clusters. One obvious solution would be to compare the clusters with a well-defined ontology such as WordNet [13]. But the Time corpus had a lot of proper nouns such as names of countries and people who were in the news during the cold-war period, thus rendering the comparison with WordNet inappropriate. We introduce a hypothesis which aids us in evaluating the clusters.

**Hypothesis:** The points closest to the cluster center are representative of the cluster.

As this hypothesis is intuitively justifiable to an extent, we choose not to further explain it here. We take the m closest points to the cluster center (for each of the clusters generated) and use them to represent the cluster. We call these words as “Representative Words.” If the words thus selected are semantically related to each other then the representative words can be said to be semantically coherent. As we move away from the cluster center, the semantic coherence of the words is expected to decrease. For the purpose of our experiments, we take five words ( $m=5$ ) closest to the cluster center. To evaluate the semantic coherence of the five words thus selected, we use independent knowledge sources such as Google [14] and Wikipedia [15]. Past work, including, [18], [19] and [20], all compare their results against a lexical resource like WordNet. Instead, we appeal to knowledge sources like Google and Wikipedia to evaluate the effectiveness of our clusters. We queried the knowledge sources manually using various combinations of the five words of each cluster and tried to understand the semantic similarity among them in the context of the cold-war (the timeframe during which the articles in the time corpus were published).

## 5 Results

We present herewith the results of the experiments. We choose to present the results of the experiments with  $k=10$  fully and a sample of semantically coherent clusters from the results for  $k=50$  (due to space limitations). Some clusters have less than 5 words in them. Descriptions are not provided for clusters whose semantic coherence we were unable to mine manually and those for which the semantic coherence is evident. The representative words are ordered in ascending order of distances to the center of their clusters. All descriptions were derived by summarizing information found using Google and Wikipedia and do not represent the authors' opinions about the word cluster. All results in the tables that follow have been gathered with  $m=5$ . To get a feel of the decrease in semantic coherence with increasing  $m$ , we present some representative results for  $m>5$  herewith.

**Table 1.** Representative Results with  $m > 5$  for clusters gathered with  $k = 50$ , words ordered in the order of increasing distance from the centroid of the cluster

<Brunei, borneo, Malayan, malay, Singapore, Malaysia, Indonesia, sukarno, malaya, federation, rahman, abdul,...>
<Syria, middle, Arabs, Syrian, Unity, Jordan, Saudi, Union, Iraq, Aflak, Egypt, Yemen, Baath, Arab, Nasser...>
<elisabethville, leopodville, united, central, congo, Katanga, tshombe, troops, president, police,...>

### 5.1 k = 10

**Table 2.** Results for the application of k-means with  $k = 10$

Cluster #	Representative Words (m=5)	Descriptions from Independent Knowledge Sources
0	Damascus, Arabs, Syrian, Egyptian, Jordan	<b>Syria, Egypt</b> and <b>Jordan</b> are <b>Arab</b> nations. <b>Damascus</b> is the capital of <b>Syria</b>
1	time, minister, years, labor, week	<b>Labor</b> is a political party which had <b>ministers</b> in power during the cold war <b>years</b>
2	European, charles, nuclear, market, french	<b>French</b> are the peoples of the <b>European</b> nation of France
3	lemass, Ireland, irish, Dublin	<b>Ireland</b> , whose peoples are called <b>Irish</b> has its capital at <b>Dublin</b> . Sean <b>Lemass</b> was an Irish political leader
4	Saigon, Vietnam, cong, Buddhist, nhu	<b>Saigon</b> is district one of ho-chi-min city, the capital of <b>Vietnam</b> . Madame <b>Nhu</b> , the first lady, was a member of the Viet <b>Cong</b> , which had anti- <b>Buddhist</b> policies

**Table 2.** (*continued*)

5	small, including, finally, high, united	
6	Brunei, malay, Malayan, borneo, Singapore	<b>Malay</b> is the language spoken by the <b>Malayan</b> people and is the official language of Malaysia, <b>Brunei</b> and <b>Singapore</b> . The Malaysian city of Sabah was called British <b>Borneo</b> when it was a British colony
7	constantly, ability, mistakes, endless, aide	
8	peking, red, mao, soviet, communist	<b>Peking</b> was the former name of the Beijing, the capital of China where the book called the little <b>red</b> book of quotations by <b>Mao</b> Zedong was published in 1962. He was trying to drive a wedge between Moscow of <b>soviet</b> Russia and Peking of China. Both China and Soviet Russia were <b>communist</b> nations.
9	famine, densely, Malthusian, ecological, bachelors	

## 5.2 k = 50

**Table 3.** Representative Results for the application of k-means with k = 50

Cluster #	Representative Words (m=5)	Descriptions from Independent Knowledge Sources
0	white, African, Africa, black, god	<b>Africa</b> is known for the racist turmoil between the <b>whites</b> and the <b>blacks</b> .
30	tents, Libyan, fires, mosques, bricks	<b>Libya</b> is known for its <b>mosques</b> .
35	Pakistan, India, Kashmir, Nehru	<b>Nehru</b> was the prime minister of <b>India</b> , which has a dispute with neighboring <b>Pakistan</b> over the occupancy of <b>Kashmir</b>
44	sinistra, palmoiro, Giovanni, toligatti, leone	Partito Comunista Italiano, the Italian Communist Party was headed by <b>Palmoiro Toligatti</b> . <b>Giovanni</b> was an Italian astronomer. <b>Leone</b> Battista was an Italian painter.

## 6 Conclusions

Firstly, as the results show, the clustering of words does indeed reveal the semantic relationship among the words in the corpus. There is a definite bias due to the corpus

used, which in this case is the bias of the cold war period i.e., the semantic relationship between the words in the cold war period is being revealed through our experiments. The results confirm our hypothesis that clustering of nDF vectors would reveal the semantic relationships.

Secondly, as can be seen from the results, the proper nouns (such as India, Singapore, Malay) and their variations (such as Indian, Malayan etc.) get separated from the common nouns i.e., the set of representative words for a cluster is seldom a mixture of proper nouns (and their variations) and proper English words. There are exceptions such as the cluster #9 for k=10. In general, unless there is a strong semantic relationship, the proper nouns get reasonably well separated from the other English words.

Thirdly, the semantic relationship between proper nouns (and variants) is made explicit by clustering. The problem of identifying the semantic relationships between proper nouns has been well studied in the computational linguistics literature. Here, we have achieved a reasonable accuracy of identifying the semantic relationship between nouns without using either parsing or part-of-speech tagging, which are considerably expensive and often used in the literature. The clusters which have only proper nouns in them, such as <Pakistan, India, Kashmir, Nehru> testify our claim.

Fourthly, the increase in distances of the representative words from the center of the cluster does present a decreasing amount of semantic coherence with the words closer to the center. For example, the inclusion of Singapore (word #5) into the top 4 words in cluster #6 with k = 10, does decrease the coherence of the set. Another example would be inclusion of Nehru (word #4), the name of a person into the top 3 words (Pakistan, India and Kashmir) which are names of places and nations. It is interesting to observe that the ordering of words does convey some clues to the semantic relationships.

## 7 Using Clusters of Semantically Related Words

Clusters of semantically related words could be used for query expansion and query relevance measurement in an information retrieval (IR) system. Words in the same cluster as the query words for a query posed by a user in an IR system can be used as suggestions to expand the query for better retrieval. In a multi-word query, whether or not the query terms appear in the same cluster could be used to measure the relevance of the query to the corpus in an IR system. For example, “Moscow+nuclear” would be a less relevant query (to the Time corpus that we have used in our experiments) compared to “France+nuclear” as the query terms in the former appear in different clusters and the query terms in the latter appear in the same cluster.

The technique could be used in a variety of corpus-based unsupervised learning tasks. The hypothesis that words close to the center are representative of the cluster can be used to identify the topic that a topical corpus (a corpus that deals with a specific topic) deals with, by considering the entire collection of words in the corpus as a single cluster. Topical corpora include the collection of postings in a forum, of chat sessions in a focused chat room, and that of entries in a topical weblog. Identification of the context (or sense) of a term’s usage in the corpus can be done by

means of the semantically related word clusters. For example, “red” is used in the context of the Moscow and Soviet Union rather than in the context of colors as “Moscow” and “Soviet” are in the same cluster as “red” in our experiments. Similarly, “cong” is used in the context of the Vietnam Congress and not the US congress.

Identification of semantically related word clusters would aid in automatic annotation of documents from the space of the entire vocabulary (as opposed to classification tasks which have a fixed small set of labels). Each document could be assigned to one or more clusters (based on statistical similarity measures) and the most representative words from those clusters could be used to label the document. Note that in such an approach, the label of a document need not necessarily come from the set of words present in the document. Such sets of words may well be used as compact representations of documents for data mining tasks.

The technique presented is general enough that we could replace the set of words by a set of features and identify the semantic relationships between features. For instance, a spam filter may use features such as occurrence of phrases (e.g., “over 21”, “mortgage rates”) and other non-trivial features such as background color (a red background color is indicative of a porn mail). We would expect our technique to cluster features specific to a category of mail together, e.g., for instance features specific to porn mail might just fall into the same cluster.

As our results show, most of the representative words for a cluster are determiners for particular classes. For instance, the most representative words of cluster 0 in table 2 would intuitively be good determiners for documents relating to the arab world i.e., documents relating to the arab world would cluster well together if we project them on the space of the 5 most representative words of cluster 0. Given that each set of most representative words would most likely be good determiners for one category or the other, the projection of the documents on the union of all such sets would separate out the documents based on the class they belong to. Thus, taking the union of all most-representative-word sets could be used as an unsupervised feature selection technique for document clustering.

## 8 Contributions and Future Work

### 8.1 Contributions

Firstly, we have demonstrated nDF vector clustering as a feasible tool for the extraction of semantically related sets. Secondly, by means of our hypothesis that words closest to the cluster center are representative of the cluster, we have proposed a means of evaluating cluster quality even for a large number of clusters. Thirdly, this is the first study which tries to extract semantic information using the bag-of-words model for documents without using any linguistic techniques. Fourthly, to the best of our knowledge, this is the first study which verifies the applicability of the low-cost k-means algorithm for term clustering. All earlier studies have used the more expensive hierarchical agglomerative clustering algorithm.

## 8.2 Future Work

It is intuitive that the value of k is hardly predictable in cases such as clustering on text data. We experimented with varying values of k in our experiments. Future work could use techniques such as Bayesian Information Criterion [16] to estimate the number of clusters or use algorithms such as bisecting k-means [17] which don't require k as an input parameter. Further, as mining semantic relationships between nouns is the most interesting component of extracting semantic information, computational linguistics techniques could be used to find nouns in the dictionary and cluster them alone. This would render the technique comparable to the techniques which aim at finding the semantic relationship between nouns (from the computational linguistics community). Further, nDFITF clustering could also be tried out to get further insights about the distribution of words in text corpuses.

## References

1. Harry Wu, Gerard Salton, "A comparison of search term weighting: term relevance vs. inverse document frequency", Proceedings of the 4<sup>th</sup> Annual ACM SIGIR, Californian, 1981, pp.30-39
2. Dekang Lin, "Automatic retrieval and clustering of similar words", Proc of COLING-ACL, 1998, pp.768-774
3. Hindle, "Noun Classification from predicate argument structures", Proc of the 28<sup>th</sup> Annual Meeting of the ACL, 1990, pp. 268-275
4. Caraballo, Charniak, "Automatic construction of a hypernym-labeled noun hierarchy from text", Proc of the 37<sup>th</sup> Annual Meeting of the ACL, 1999, pp.120-126
5. A. Maedche, V. Pekar, S. Staab, "Ontology learning part one - On discovering taxonomic relations from the web." In Web Intelligence, pages 301-322. Springer Verlag, 2002.
6. F. C. Pereira, N. Thishby, L. Lee, "Distributional Clustering of English Words", In Proc of the 30<sup>th</sup> Annual Meeting of the ACL, 1993, pp.183-190
7. M Sanderson, W V Croft, "Deriving Concept hierarchies from text", Proceedings of the 22<sup>nd</sup> SIGIR Conference, 1999, pp.206-213
8. Steinbach. M, Karypis. G, Kumar. V, "A comparison of document clustering techniques", In Proceedings of the KDD Workshop on Text Mining, Boston, 2000
9. K. Wagstaff, C. Cardie, S. Rogers, S Schroedl, "Constrained K-Means Clustering with background knowledge", Proceedings of the 18th International Conference on Machine Learning (ICML-2001), pp. 577-584
10. Sugato Basu, Arindham Banerjee, Raymond Mooney, "Semi-supervised Clustering by seeding", Proc of the 19th Intl Conference on Machine Learning (ICML), 2002, pp. 19-26
11. Sabine Bergler. "Collocation patterns for verbs of reported speech--a corpus analysis oil tile time Magazine corpus". Technical: report, Brandeis University Computer Science., 1990.
12. "Weka: Open Source Software for Data Mining", [www.cs.waikato.ac.nz/ml/weka/](http://www.cs.waikato.ac.nz/ml/weka/)
13. "WordNet: Online Lexical Reference System", <http://wordnet.princeton.edu>
14. Google, <http://www.google.com>
15. Wikipedia, The Free Encyclopedia, <http://en.wikipedia.org>
16. Dan Pelleg, Andrew Moore, "X-means: Extending K-means with Efficient Estimation of the Number of Clusters", Proceedings of the 7<sup>th</sup> International Conference on Machine Learning (ICML), 2000, pp. 727-734

17. Sergio N Saveresi, D L Boley, "A comparative analysis on the bisecting K-means and the PDDP clustering algorithms", Intelligent Data Analysis Journal, 2004
18. E. Riloff, J. Shepherd, "A corpus based approach for building semantic lexicons", In Proceedings of the Second Conference on Empirical Methods in Natural Language Processing (EMNLP), 1997, pp.117-124
19. Brian Roark, Eugene Charniak: "Noun-Phrase Co-Occurrence Statistics for Semi-Automatic Semantic Lexicon Construction." In Proc. Of the 36<sup>th</sup> Annual Meeting of the ACL, 1998, pp.1110-1116
20. Dorow. B, Widows. D, "Discovering Corpus-Specific Word Senses", Proc. of the 10<sup>th</sup> Conference of the European Chapter of the ACL, 2003, pp.79-82
21. Dominic Widdows, Unsuperivsed methods for developing taxonomies by combining syntactic and statistical information, HLT-NAACL 2003

# Recognition of Simultaneous Speech by Estimating Reliability of Separated Signals for Robot Audition

Shun'ichi Yamamoto<sup>1</sup>, Ryu Takeda<sup>1</sup>, Kazuhiro Nakadai<sup>2</sup>, Mikio Nakano<sup>2</sup>,  
Hiroshi Tsujino<sup>2</sup>, Jean-Marc Valin<sup>3</sup>, Kazunori Komatani<sup>1</sup>,  
Tetsuya Ogata<sup>1</sup>, and Hiroshi G. Okuno<sup>1</sup>

<sup>1</sup> Graduate School of Informatics, Kyoto University, Japan

{shunichi, rtakeda, komatani, ogata, okuno}@kuis.kyoto-u.ac.jp

<sup>2</sup> HONDA Research Institute Japan Co., Ltd., Japan

{nakadai, nakano, tsujino}@jp.honda-ri.com

<sup>3</sup> CSIRO ICT Centre, Australia

jean-marc.valin@csiro.au

**Abstract.** “*Listening to several things at once*” is a people’s dream and one goal of AI and robot audition, because people can listen to at most two things at once according to psychophysical observations. Current noise reduction techniques cannot help to achieve this goal because they assume quasi-stationary noises, not interfering speech signals. Since robots are used in various environments, robot audition systems require minimum *a priori* information about their acoustic environments and speakers. We evaluate a missing feature theory approach that interfaces between sound source separation (SSS) and automatic speech recognition. The essential part is the estimate of reliability of each feature of separated sounds. We tested two kinds of robot audition systems that use SSS: independent component analysis (ICA) with two microphones, and geometric source separation (GSS) with eight microphones. For each SSS, automatic missing feature mask generation is developed. The recognition accuracy of two simultaneous speech improved to an average of 67.8 and 88.0% for ICA and GSS, respectively.

## 1 Introduction

“*Listening to several things at once*” is a people’s dream and one goal of AI and robot audition, because psychophysical observations reveal that people can listen to at most two things at once. Robot audition is, an essential intelligent function for robots working with humans. Since robots encounter various sounds and noises, robot audition systems should be able to recognize a mixture of sounds and be noise-robust. Since robots are deployed in various environments, robot audition systems require minimum *a priori* information about their acoustic environments and speakers [1,2].

A robot audition system usually integrates sound source separation (SSS) and automatic speech recognition (ASR) subsystems. To minimize *a priori* information, we use blind source separation, a beamformer for SSS, and missing feature theory (MFT) for ASR. The former separates sound signals from a mixture of sounds without assuming the characteristics of sound sources. The latter recognizes speech signals with a *clean* acoustic model by using missing feature masks (MFMs) that specify whether each spectral feature is reliable or not.

The most critical issue in missing feature mask generation is reliability estimation of spectral features in separated speech signals. The conventional studies on MFT focus only on cases where interfering sounds are quasi-stationary noises. This approach cannot handle two simultaneous speech signals. We assume that separated sounds are distorted mainly by signal leakage from other sound sources. Therefore, in separating sounds, the system first estimates signal leakage and then identifies which spectral components are distorted. Finally, it creates MFMs that specify whether each spectral feature is reliable or not.

We demonstrated the performance of automatically generated MFMs by evaluating two robot audition systems that automatically generate them:

**GSS** Geometric source separation (GSS) with eight microphones and automatic MFM generation for it, and

**ICA** Independent Component Analysis (ICA) with two microphones and automatic MFM generation for it.

The separated speech signals and their associated MFMs are given to Multiband Julius, MFT-based ASR, to recognize. In **GSS**, the signal leakage is estimated by a multi-channel post-filter, which estimates signal leakage from other channels and quasi-stationary noises. In **ICA**, a SIMO (single-input multiple-output) model is used to obtain two channels (left and right) for each sound source. Then SIMO signals are used to estimate signal leakage.

This paper describes two systems, **ICA** and **GSS**, from the viewpoint of MFT. It presents MFT-based ASR, explains benchmarks, and presents their results.

## 1.1 Related Work

Noise-robust ASRs have been studied extensively, for example in the AURORA project [3,4]. One common method, in particular for in-car and telephony applications, is multi-condition training (training on a mixture of clean speech and noises) [5,6]. Since an acoustic model obtained by multi-condition training reflects all expected noises in specific conditions, ASR's use of such an acoustic model is effective only as long as the noise is quasi-stationary. This assumption holds well for background noises in a car and on a telephone. However, multi-condition training may not be effective for robots, since they usually work in a dynamically changing noisy environment.

MFT-based ASR has been studied as a method of noise-robust ASR[7]. A spectrographic mask (also called MFM in this paper) is the set of tags that identify reliable and unreliable components of the spectrogram. MFT-based ASR uses this spectrographic mask to avoid corrupt signals during the decoding process. One of main approaches modifies the classifier, or recognizer, to recognize speech signals using reliable separated components and unreliable original input components [8,9,10,11,12]. Except for Yamamoto *et al.* [11,12], most studies have not focused on recognition of speech signals corrupted by interfering speech signals.

Sound source separation has been studied extensively with microphone array systems that usually use beamformer or null beamformer [13]. These systems usually require the precise 3D location of each microphone and information about speakers. On the other

hand, blind source separation does not assume such *a priori* information. ICA assumes only independence of data [14].

Robot audition requires three essential functions, sound source localization, separation, and recognition of separated sounds. Nakadai *et al.* [15] have reported a robot audition system that can recognize three simultaneous speech signals for real-time and real-world processing using a pair of microphones installed in its ear position. Their system was developed by unifying four components: (1) an active audition system to perceive auditory information better by controlling microphone parameters, (2) a real-time multiple human tracking system by integration of the active audition system, face localization, face recognition and stereo vision, (3) an active direction-pass filter (ADPF) to separate sound sources, and (4) ASR by using multiple direction- and speaker-dependent acoustic models. In other words, their system required a lot of information about acoustic environments and speakers.

## 2 General Recognition Architecture

A general architecture for recognizing several speech signals at once consists of three components:

1. Sound source separation,
2. MFT-based ASR, and
3. Automatic MFM generation.

The last is a bridge between the first and second components. In this section, we focus on the second component, MFT-based ASR, since it is used commonly by ICA and GSS systems.

### 2.1 Acoustic Features of MFT-Based ASR

Since sound source separation is performed at the level of spectral representation, we use spectral features for MFT-based ASR. Although Mel-Frequency Cepstrum Coefficient (MFCC) is a common acoustic feature for ASR, it is not appropriate for MFT-based ASR, because a noise in each frequency band spreads to all coefficients in cepstral domain. We used the Mel Scale Log Spectrum (MSLS) obtained by applying Inverse Discrete Cosine Transformation to MFCCs. The calculation of MSLS is described by Yamamoto *et al.* [11]. The acoustic feature vector is composed of 48 spectral-related acoustic features: 24 spectral features and 24 dynamic features.

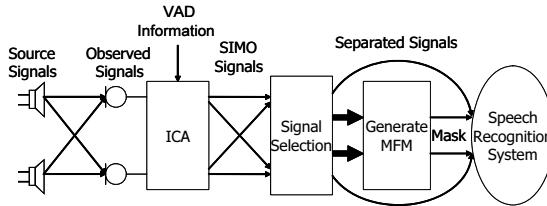
### 2.2 Missing Feature Theory-Based Automatic Speech Recognition

MFT-based ASR outputs a sequence of phonemes from acoustic features of separated speech and the corresponding MFMs. MFT-based ASR is an Hidden Markov Model (HMM)-based recognizer, which is commonly used in conventional ASR systems. The only difference is in their decoding processes. In conventional ASR systems, estimation of a path of maximum likelihood is based on state transition probabilities and output probability in HMM. This process of estimating output probability is modified in MFT-based ASR as follows: let  $M(i)$  be an MFM vector which represents the reliability of the  $i$ -th acoustic feature. The output probability  $b_j(x)$  is given by

$$b_j(x) = \sum_{l=1}^L P(l|S_j) \exp \left\{ \sum_{i=1}^N M(i) \log f(x(i)|l, S_j) \right\}, \quad (1)$$

where  $P(\cdot)$  is a probability operator,  $x(i)$  is an acoustic feature vector,  $N$  is the size of the acoustic feature vector, and  $S_j$  is the  $j$ -th state.

For MFT-based ASR, we used Multiband Julius [16], which is an extension of the Japanese real-time, large vocabulary, speech recognition system Julius[17].



**Fig. 1.** Overview of ICA system

### 3 ICA-Based Separation and MFM Generation

The ICA system consists of three components as shown in Fig. 1: (1) ICA as a blind source separation, (2) MFT-based ASR, and (3) Automatic MFM generation. The last component is a bridge between the first and second ones.

#### 3.1 Frequency-Domain ICA

We used a frequency domain representation instead of a temporal domain one. The search space is smaller because the unmixing matrix is updated for each frequency bin, and thus its convergence is faster and less dependent on initial values.

The signals were assumed to be observed by linearly mixing sound sources, expressed as follows:

$$\mathbf{x}(t) = \sum_{n=0}^{N-1} \mathbf{a}(n) \mathbf{s}(t-n), \quad (2)$$

where  $\mathbf{x}(t)$  is the observed signal vector, and  $\mathbf{s}(t)$  is the source signal vector.  $\mathbf{a}(n)$  is the mixing filter matrix with length  $N$ . In our experiment, the number of microphones,  $J$ , was two and the number of multiple sound sources,  $L$ , was two.

A frequency-domain ICA works as follows. First, the short-time analysis of observed signal is conducted by frame-by-frame discrete Fourier transform (DFT) to obtain the observed vector  $\mathbf{X}(\omega, t)$  in each frequency bin  $\omega$  and at each frame  $t$ . The unmixing process can be formulated for a frequency bin  $\omega$

$$\mathbf{Y}(\omega, t) = \mathbf{W}(\omega) \mathbf{X}(\omega, t), \quad (3)$$

where  $\mathbf{Y}(\omega, t)$  is the estimated source signal vector, and  $\mathbf{W}$  represents a (2 by 2) unmixing matrix in frequency bin  $\omega$ .

For estimating the unmixing matrix  $\mathbf{W}(\omega)$  in (3), an algorithm based on the minimization of the Kullback-Leibler divergence is often used. Therefore, we use the following iterative equation with non-holonomic constraints:

$$\mathbf{W}^{j+1}(\omega) = \mathbf{W}^j(\omega) - \alpha \{\text{off-diag}\langle \phi(\mathbf{Y})\mathbf{Y}^h \rangle\} \mathbf{W}^j(\omega), \quad (4)$$

where  $\alpha$  is a step size parameter that has effects on the speed of convergence,  $j$  expresses the value of the  $j$ -th step in the iterations, and  $\langle \cdot \rangle$  denotes the time-averaging operator. The operation, off-diag( $\mathbf{X}$ ), replaces the diag-element of matrix  $X$  with zero. In this paper, the nonlinear function,  $\phi(\mathbf{y})$ , is defined as  $\phi(y_i) = \tanh(|y_i|)e^{j\theta(y_i)}$ .

### 3.2 Two ICA's Problems of Permutation and Scaling

Frequency-domain ICA suffers from two ambiguities: *scaling ambiguity*, i.e., the power of separated signals differs at each frequency bin, and *permutation ambiguity*, i.e., signal components are swapped among different channels. These ambiguities occur because ICA estimates both unmixing matrix  $\mathbf{W}$  and source signal vector  $\mathbf{Y}$  at the same time. We solved these ambiguities in order to recover the spectral representation as completely as possible using Murata's method[18]. This solution gives SIMO signals, which are used to generate MFM. The permutation ambiguity can be solved by taking into consideration correlation of envelopes of power spectrum among frequency bins.

### 3.3 Improvement by Voice Activity Detection (VAD)

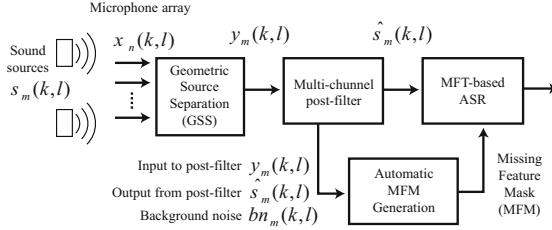
Since the convolution model does not reflect actual acoustic environments, no method based on this model can completely decompose each signal component. The spectral distortion of separated signals is mainly caused by signal leakage in the desired speech signal. If such leakage is very strong, it is difficult to determine the end of a speech sample. An incorrect estimation of a period of speech would degrade the recognition accuracy of ASR severely.

We used VAD that determines the period of utterance in order to improve the performance of separation and recognition. The number of active speakers is used as VAD information, since ADPF provides such information stably [19]. The region of silent periods is filled with silent spectrum obtained in advance. If such a region is filled with 0 signal, it may not be treated as silence by ASR with an acoustic model that is trained with clean speech signals.

### 3.4 MFM Generation for an ICA System

The error spectrum,  $\mathbf{e}$ , is estimated with SIMO signals. An MFM,  $\mathbf{M}$ , for the estimated observed spectrum,  $\mathbf{x}$ , is defined as follows:

$$\mathbf{M} = \begin{cases} 1 & |\mathbf{F}(\mathbf{x}) - \mathbf{F}(\mathbf{x} - \mathbf{e})| < \theta \\ 0 & \text{otherwise} \end{cases}, \quad (5)$$



**Fig. 2.** Overview of GSS system

where  $\mathbf{F}(\mathbf{x})$  is  $x$ 's feature and  $\theta$  is the threshold. The masks for time differential feature at time  $k$  are defined as follows:

$$\mathbf{M}(k) = \begin{cases} 1 & |\Delta\mathbf{F}_k(\mathbf{x}) - \Delta\mathbf{F}_{k-1}(\mathbf{x} - \mathbf{e})| < \hat{\theta} \\ 0 & \text{otherwise} \end{cases}, \quad (6)$$

where  $\hat{\theta}$  is the threshold.

## 4 GSS-Based Separation and MFM Generation

The GSS system consists of three components as shown in Fig. 2: (1) GSS for sound source separation, (2) multi-channel post-filter for improving separation, and (3) automatic MFM generation based on the post-filter. Since it has been reported in the literature [11,12,20], we describe it briefly.

### 4.1 Sound Source Separation and Improvement

We modified the GSS approach proposed by Parra *et al.* [21] so as to provide faster adaptation using a stochastic gradient and shorter time frame estimations [22]. The initial separation using GSS is followed by the multi-channel post-filter that is based on a generalization of beamformer post-filtering [23,22] for multiple sources. This post-filter uses adaptive spectral estimation of background noise and interfering sources to enhance the signal produced during the initial separation.

The main idea resides in the fact that, for each source of interest, the noise estimate is decomposed into stationary and transient components assumed to be due to leakage between the output channels in the initial separation stage.

This method of GSS operates in the frequency domain. Let  $s_m(k, l)$  be real (unknown) sound source  $m$  at time frame  $l$  and for discrete frequency  $k$ . The vector corresponding to the sources  $s_m(k, l)$  is  $\mathbf{s}(k, l)$ , and matrix  $\mathbf{A}(k)$  is the transfer function leading from the sources to the microphones. The signal observed at microphones is thus

$$\mathbf{x}(k, l) = \mathbf{A}(k)\mathbf{s}(k, l) + \mathbf{n}(k, l), \quad (7)$$

where  $\mathbf{n}(k, l)$  is the non-coherent background noise. The matrix  $\mathbf{A}(k)$  can be estimated using the result of a sound localization algorithm. Assuming that all transfer functions have unity gain, the elements of  $\mathbf{A}(k)$  can be expressed as:

$$a_{ij}(k) = e^{-j2\pi k \delta_{ij}}, \quad (8)$$

The separation result is then defined as  $\mathbf{y}(k, l) = W(k, l)\mathbf{x}(k, l)$ , where  $W(k, l)$  is the separation matrix. This matrix is estimated using the GSS algorithm described by Valin *et al.* [22].

The output of the GSS algorithm is then enhanced by a frequency-domain post-filter based on the optimal estimator originally proposed by Ephraim *et al.* [24]. An input of multi-channel post-filter is the output of GSS  $\mathbf{y}(k, l) = (y_1(k, l), \dots, y_M(k, l))$ . An output of the multi-channel post-filter is  $\hat{s}(k, l)$ , which is defined as

$$\hat{s}_m(k, l) = G_m(k, l)y_m(k, l), \quad (9)$$

where  $G_m(k, l)$  is a spectral gain. The estimation of  $G_m(k, l)$  is based on minimum mean-square error estimation of spectral amplitude. To estimate  $G_m(k, l)$ , noise variance is estimated.

The noise variance estimation  $\lambda_m(k, \ell)$  is expressed as

$$\lambda_m(k, \ell) = \lambda_m^{stat.}(k, \ell) + \lambda_m^{leak}(k, \ell), \quad (10)$$

where  $\lambda_m^{stat.}(k, \ell)$  is the estimate of the stationary component of the noise for source  $m$  at frame  $\ell$  for frequency  $k$ , and  $\lambda_m^{leak}(k, \ell)$  is the estimate of source leakage.

We computed the stationary noise estimate,  $\lambda_m^{stat.}(k, \ell)$ , using the minima controlled recursive average technique [25]. To estimate  $\lambda_m^{leak}$ , we assumed that the interference from other sources is reduced by factor  $\eta$  (typically  $-10 \text{ dB} \leq \eta \leq -5 \text{ dB}$ ) by the LSS. The leakage estimate is thus expressed as

$$\lambda_m^{leak}(k, \ell) = \eta \sum_{i=0, i \neq m}^{M-1} Z_i(k, \ell), \quad (11)$$

where  $Z_m(k, \ell)$  is the smoothed spectrum of the  $m^{\text{th}}$  source,  $Y_m(k, \ell)$ , and is recursively defined (with  $\alpha_s = 0.7$ ) [22]:

$$Z_m(k, \ell) = \alpha_s Z_m(k, \ell - 1) + (1 - \alpha_s) Y_m(k, \ell). \quad (12)$$

## 4.2 MFM Generation for GSS System

We computed the MFM using input  $y_m(k, l)$ , output  $\hat{s}_m(k, l)$ , and the estimated background noise,  $bn_m(k, l)$ , of the multi-channel post-filter. The variables filtered by the Mel filter bank are  $Y_m(k, l)$ ,  $\hat{S}_m(k, l)$ , and  $BN_m(k, l)$ , respectively. For each Mel-frequency band, the feature is considered reliable if the ratio of the output energy over the input energy is greater than a threshold,  $T_{MFM}$ . This assumes that the more noise present in a certain frequency band, the lower the post-filter gain will be for that band.

The MFM  $M_m(k, i)$ , ( $i = 1, \dots, N$ ) for the spectral feature is defined as

$$M_m(k, i) = \begin{cases} 1, & \frac{\hat{S}_m(k, i) + BN_m(k, i)}{Y_m(k, i)} > T_{MFM} \\ 0, & \text{otherwise} \end{cases}. \quad (13)$$

The MFM  $M_m(k, i)$ , ( $i = N + 1, \dots, 2N$ ) is defined as

$$M_m(k, i) = \prod_{i=k-2, i \neq k}^{k+2} M_m(i, i - N). \quad (14)$$

## 5 Experiments and Evaluation

To evaluate efficiency of automatic MFM generation based on leak estimation, we performed experiments on recognition of two simultaneous speech signals.

### 5.1 Recording Conditions

We used Robovie-R2 for the experiments, with eight microphones attached to the body. The positions of the microphones are shown in Fig. 3. For the ICA system, a pair of upper front microphones were used. Simultaneous speech signals were recorded in a room, as shown in Fig. 4. Its reverberation time was about 0.35 seconds (RT20). Japanese words were played simultaneously through loudspeakers at the same distance from the robot. Locations varied over five distances (at 50, 100, 150, 200, and 250 cm from the robot) and three directions. One loudspeaker was fixed in front of the robot, and the other was placed at 30, 60, or 90° left of the robot. The volume of the loudspeakers was set at the same level for all locations.

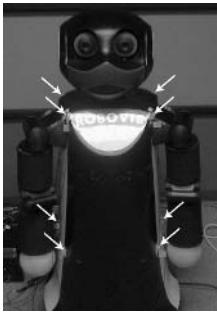
### 5.2 Speech Recognition

Multiband Julius was used as the MFT-based ASR. In the experiments, we used a tri-phone acoustic model trained and a grammar-based language model to recognize an isolated word. A triphone is an HMM which has 3 states and 4 mixtures in each state, and trained on 216 clean, phonemically balanced words distributed by ATR. The size of the vocabulary was 200 words.

### 5.3 Results

Fig. 5 and Fig. 6 show word recognition rates for the ICA and GSS systems, respectively. The horizontal axis indicates speakers' positions, and the vertical one indicates word correct rates. For example, “30 deg., 50 cm” on the horizontal axis means that one speaker is located 50 cm in front of the robot, and the other one is located 50 cm away at 30° left of center.

The ICA-based MFM generation (the ICA system) improved word correct rates by an average of 5.6%, and the GSS-based MFM generation (the GSS system) improved word correct rates by an average of 4.8%. The word correct rates of two simultaneous speech improved to an average of 67.8 and 88.0% for the ICA and GSS systems, respectively.

**Fig. 3.** Robovie-R2**Fig. 4.** Robovie-R2 in the experiment room

## 5.4 Discussion

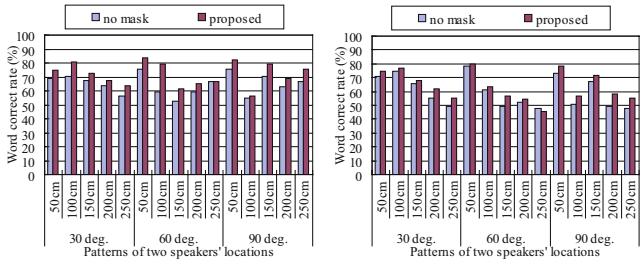
We have two observations.

The ICA system worked better in the near field than in the far field, because room transfer functions such as reverberation degraded the separation performance of ICA. The effect created by the intervals between the two speakers did not degrade the word correct rate so much for the ICA system. The optimiazed unmixing matrix obtained by ICA is the reason for the system's robustness with intervals.

The GSS system worked better in the far field than in the near field, because a large difference in the time delay of arrival (TDOA) increases resolution of GSS. Narrow intervals between the speakers degraded the separation performance of GSS and the multi-channel post-filter, because the difference in TDOA decreased GSS calculates the TDOA from locations of sound sources using the geometric constraints of microphones and does not take into consideration transfer fuctions of the body of the robot. The unmixing matrix obtained by ICA estimates such body transfer functions.

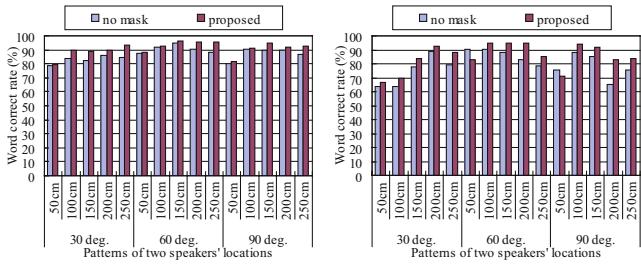
## 6 Conclusion

We presented two kinds of missing-feature approaches to separate and recognize two simultaneous speech signals. The ICA system uses ICA for sound source separation with two microphones. The GSS system uses GSS, a kind of beam-former, for sound source



a) Center speaker

b) Left speaker

**Fig. 5.** Recognition results of ICA system

a) Center speaker

b) Left speaker

**Fig. 6.** Recognition results of GSS system

separation with eight microphones. Both separated sounds are recognized by Missing-Feature Theory based ASR. These two systems were evaluated based on recognition rates of simultaneous speech uttered by two speakers. We demonstrated that robot audition systems consisting of blind source separation and MFT-based ASR with automatic MFM generation recognized two simultaneous speech signals 5.6% and 4.8% better than conventional systems.

## Acknowledgment

Our research is partially supported by Grant-in-Aid for Scientific Research and 21 COE Program. We thank Jean Rouat and François Michaud, Université of Sherbrooke for their help. We thank the members of Honda Research Institute Japan Co., Ltd. and the members of Okuno laboratory, Kyoto University.

## References

1. Okuno, H.G., Nakatani, T., Kawabata, T.: Interfacing sound stream segregation to speech recognition systems — preliminary results of listening to several things at the same time. In: Proceedings of 13th National Conference on Artificial Intelligence (AAAI-96), AAAI (1996) 1082–1089
2. Okuno, H.G., Nakatani, T., Kawabata, T.: Understanding three simultaneous speakers. In: Proc. of IJCAI-1997. (1997) 30–35
3. AURORA: (<http://www.elda.fr/proj/aurora1.html>)  
“<http://www.elda.fr/proj/aurora2.html>”
4. Pearce, D.: Developing the ETSI AURORA advanced distributed speech recognition front-end & what next. In: Proc. of Eurospeech-2001, ESCA (2001)
5. Lippmann, R.P., Martin, E.A., Paul, D.B.: Multi-style training for robust isolated-word speech recognition. In: Proc. of ICASSP-87, IEEE (1987) 705–708
6. Blanchet, M., Boudy, J., Lockwood, P.: Environment adaptation for speech recognition in noise. In: Proc. of EUSIPCO-92. Volume VI. (1992) 391–394
7. Raj, B., Stern, R.M.: Missing-feature approaches in speech recognition. Signal Processing Magazine **22**(5) (2005) 101–116
8. Barker, J., Cooke, M., Green, P.: Robust ASR based on clean speech models: An evaluation of missing data techniques for connected digit recognition in noise. In: Proc. of Eurospeech-2001, ESCA (2001) 213–216
9. Cooke, M.P., Green, P., Josifovski, L., Vizinho, A.: Robust automatic speech recognition with missing and unreliable acoustic data. Speech Communication **34**(3) (2000) 267–285
10. Renevey, P., Vetter, R., Kraus, J.: Robust speech recognition using missing feature theory and vector quantization. In: Proc. of 7th European Conference on Speech Communication Technology (Eurospeech-2001). Volume 2., ESCA (2001) 1107–1110
11. Yamamoto, S., Valin, J.M., Nakadai, K., Ogata, T., Okuno, H.G.: Enhanced robot speech recognition based on microphone array source separation and missing feature theory. In: Proceedings of IEEE International Conference on Robotics and Automation (ICRA 2005), IEEE (2005) 1489–1494
12. Yamamoto, S., Nakadai, K., Valin, J.M., Rouat, J., Michaud, F., Komatani, K., Ogata, T., Okuno, H.G.: Making a robot recognize three simultaneous sentences in real-time. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2005), IEEE (2005) 897–902

13. Haykin, S.: Adaptive Filter Theory, 4th. ed. Prentice Hall, Inc. (2002)
14. Hyvärinen, A., Karhunen, J., Oja, E.: Independent Component Analysis. John Wiley & Sons, Inc. (2001)
15. Nakadai, K., Okuno, H.G., Kitano, H.: Robot recognizes three simultaneous speech by active audition. In: Proceedings of IEEE International Conference on Robotics and Automation (ICRA-2003), IEEE (2003) 398–403
16. Multiband Julius: ([http://www.furui.cs.titech.ac.jp/mband\\_julius/](http://www.furui.cs.titech.ac.jp/mband_julius/))
17. Kawahara, T., Lee, A.: Free software toolkit for Japanese large vocabulary continuous speech recognition. In: International Conference on Spoken Language Processing (ICSLP). Volume 4. (2000) 476–479
18. Murata, N.: An approach to blind source separation based on temporal structure of speech signals. In: Neurocomputing. (2001) 1–24
19. Nakadai, K., Hidai, K., Mizoguchi, H., Okuno, H.G., Kitano, H.: Real-time auditory and visual multiple-object tracking for robots. In: Proc. of IJCAI-2001. (2001) 1424–1432
20. Yamamoto, S., Nakadai, K., Nakano, M., Tsujino, H., Valin, J.M., Takeda, R., Komatani, K., Ogata, T., Okuno, H.G.: Genetic algorithm-based improvement of robot hearing capabilities in separating and recognizing simultaneous speech signals. In: Proceedings of 19th International Conference on Industrial, Engineering, and Other Applications of Applied Intelligent Systems (IEA/AIE'06). Volume LNAI 4031., Springer-Verlag (2006) 207–217
21. Parra, L.C., Alvino, C.V.: Geometric source separation: Margin convolutive source separation with geometric beamforming. IEEE Transactions on Speech and Audio Processing **10**(6) (2002) 352–362
22. Valin, J.M., Rouat, J., Michaud, F.: Enhanced robot audition based on microphone array source separation with post-filter. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2004), IEEE (2004) 2123–2128
23. Cohen, I., Berdugo, B.: Microphone array post-filtering for non-stationary noise suppression. In: ICASSP-2002. (2002) 901–904
24. Ephraim, Y., Malah, D.: Speech enhancement using minimum mean-square error log-spectral amplitude estimator. IEEE Transactions on Acoustics, Speech and Signal Processing **ASSP-33**(2) (1985) 443–445
25. Cohen, I., Berdugo, B.: Speech enhancement for non-stationary noise environments. Signal Processing **81**(2) (2001) 2403–2418

# Chinese Abbreviation-Definition Identification: A SVM Approach Using Context Information\*

Xu Sun, Houfeng Wang, and Yu Zhang

Department of Computer Science and Technology  
School of Electronic Engineering and Computer Science

Peking University, Beijing, 100871, China  
sunxu@pku.edu.cn, wanghf@pku.edu.cn, sdrzy@pku.edu.cn

**Abstract.** As a special form of unknown words, Chinese abbreviations represent significant problems for Chinese text processing. The goal of this study is to automatically find the definition for a Chinese abbreviation in the context where both the abbreviation and its definition occur, enforcing the constraint of one sense per discourse for an abbreviation. First, the candidate abbreviation-definition pairs are collected, and then a SVM approach using context information is employed to classify candidate abbreviation-definition pairs so that the pairs can be identified. The performance of the approach is evaluated on a manually annotated test corpus, and is also compared with two other machine learning approaches: Maximum Entropy and Decision Tree. Experimental results show that our approach reaches a good performance.

## 1 Introduction

The presence of unknown words makes the natural language text not always an ideal material for text processing applications. By unknown words, we mean words and symbols that are not listed in our system dictionary, including proper nouns, abbreviations, misspelled words, etc. For modern Chinese language, a large number of abbreviated words and their definitions are used interchangeably, in order to save space and give readers an intensive impression. Moreover, new abbreviations, such as ‘非典’ for ‘非典型性肺炎’ (Severe Atypical Respiratory Syndrome), are constantly being created, which justifies the need for software tools that can automatically identify abbreviation definitions in documents.

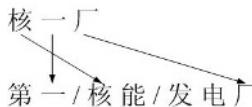
The goal of this study is to automatically find the abbreviation-definition<sup>1</sup> pairs, which is a special subtask for lexical cohesion analysis. It is also a key component for coreference resolution of named entities. The paper views abbreviation-definition identification as a classification task and a machine learning method using context information is applied to classify potential abbreviation-definition candidates.

\* Supported by National Social Science Foundation of China (No. 05BYY043) and National Natural Science Foundation of China (No. 60473138).

<sup>1</sup> In this paper, the word *abbreviation* will always stand for *Chinese abbreviation* if there is no specific indication.

We define a Chinese abbreviation as a short representation of a multi-character word or more frequently a sequence of compounding words. E.g., the abbreviation ‘高校’ is from ‘高等/学校’ (Institutions of Higher Education), which contains two words. Chang (Chang et al., 2004) excluded two abbreviation formation regularities:

- (1) **Mapping substring to null string** E.g., ‘欧洲/经济/与/货币/联盟’ (European Economic and Monetary Union) to ‘欧盟’.
- (2) **Changing character order** E.g., ‘第一/核能/发电厂’ (First Nuclear Power Plant) to ‘核一厂’, as illustrated in Fig.1.



**Fig. 1.** An example of disordered abbreviation formation

However, these two regularities cover more than 20 percents of factual abbreviations. Thus in our study we made the following assumptions about abbreviations:

- (1) An abbreviation includes fewer characters than its original definition, and an original definition can be a word or a phrase containing more than one word.
- (2) The characters used in an abbreviation are a subset of those appearing in the original definition, but the character order can be different.
- (3) We further define an abbreviation as a word that neither its whole sequence nor its substring is registered in our system vocabulary (otherwise it will be segmented during the word segmentation process). Though actually there are abbreviations containing ‘real’ words, our experiment performed on an abbreviation set containing 5121 entries shows that this kind of abbreviation is few (about 4%).

The previous related work of Chinese abbreviation expansion is relatively few. Chang et al. presented a HMM based error-recovery approach for abbreviation identification and root word recovery (Chang et al., 2004). Compared with the generative model introduced by Chang et al., the SVM approach employed in our system directly models the posterior probability of abbreviating and can get more flexibility to combine extra information, e.g., the similarity between the context of an abbreviation and that of its expansion candidate can be treated as an additional feature and then be directly combined into the SVM model.

This paper is organized as follows: Next section describes the system architecture. Section 3 and section 4 present the candidate search process and the feature set respectively. The remainder of this paper is experimental results and our conclusion.

## 2 System Description

Our abbreviation-definition identification system is made up of candidate search and abbreviation-definition disambiguation. The candidate search component is designed to collect both the abbreviation candidates and then the potential definition candidates, thus two sub-systems are employed: abbreviation candidate collector and

definition candidate collector. The next task is to determine the most appropriate definition candidate for each given abbreviation, or it may find that none of the definition candidates is appropriate for the abbreviation candidate, that is, the system decides that the abbreviation candidate is not a real abbreviation. A SVM based machine learning method is used in the disambiguation process. The detailed descriptions of candidate search and abbreviation-definition disambiguation are respectively presented in section 3 and section 4.

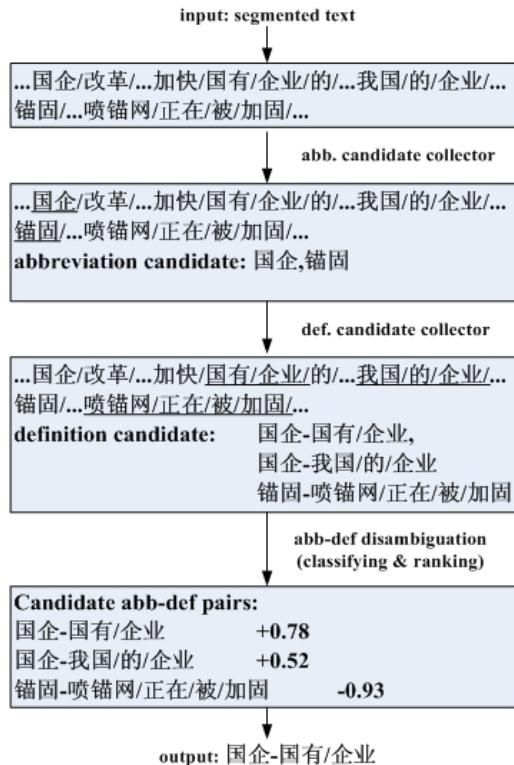


Fig. 2. System overview

The abbreviation-definition disambiguation component is the core of the system, in that it determines both the final outcome of abbreviation-identification task and definition-identification task (abbreviation-identification determines whether or not an abbreviation candidate is a real abbreviation, while definition-identification finds the corresponding definition for a given abbreviation). In this way, the tasks of abbreviation-identification and definition-identification are not isolated in the system architecture.

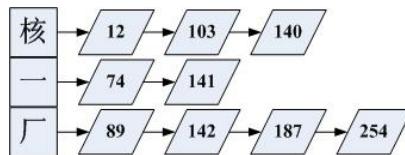
The system overview is illustrated in fig. 2. As we can see, two abbreviation candidates ‘国企’ and ‘锚固’ are collected from the source text (where only ‘国企’ is a real abbreviation and ‘锚固’ is not). Thereafter, corresponding definition candidates are collected from context, and three abbreviation-definition candidate pairs are formed: ‘国企—国有/企业’, ‘国企—我国/的/企业’ and ‘锚固—喷锚网/正在/被/加固’.

The three candidate pairs are classified and ranked by the abbreviation-definition disambiguation component, and only the abbreviation-definition pair ‘国企—国有/企业’ is outputted as ‘abbreviating’. Although the candidate pair ‘国企—我国/的/企业’ is classified as POSITIVE (abbreviating), it is discarded in that it shares the same abbreviation of another candidate pair ‘国企—国有/企业’ and only the one with the highest score will be considered as the final abbreviation expansion result in our system. On the other hand, the candidate pair ‘锚固—喷锚网/正在/被/加固’ is classified as NEGATIVE (non-abbreviating), and thus is abandoned.

### 3 Candidate Search

Both of abbreviation candidate collector and definition candidate collector are based on segmented Chinese text. Based on assumption (3) in Section 1, unknown character sequences and their substrings will be considered as potential abbreviation candidates.

When an OOV (out of vocabulary) word appears, it can also be a proper noun, a misspelled word, or a number rather than an abbreviation, thus in order to refine the abbreviation candidates, we implemented heuristics to exclude noise. Based on the observation that the number of single-character abbreviations (e.g., ‘法’ for ‘法国’) is much less than multi-character abbreviations and that it is possible to be enumerated, we use a single-character abbreviation list which is automatically collected from training corpus as a portion of our abbreviation candidate search heuristics. Besides, most of human names and numbers can be successfully identified and be excluded (a Chinese family name list is used to identify Chinese person names, and a transliterated name character list is used to identify transliterations of foreign names). However, inevitably, noise can still remain, and the final decision of abbreviation identification will be made by abbreviation-definition disambiguation, which is beyond the scope of candidate search and thus it will be presented in next section.



**Fig. 3.** An illustration of character based inverted file index

After abbreviation candidates are collected, the next task is to collect definition candidates for each abbreviation from the entire local document rather than only from local sentence or paragraph. Based on assumption (1) and (2) in section 1, we implement an inverted file index based algorithm to fulfill the collection task. Fig. 3 illustrates an inverted file index example for abbreviation ‘核一厂’, where each abbreviation character is linked with its offsets in the text. An abbreviation-definition matching algorithm shown in Fig. 4 is used to collect definition candidates, where DIST is a threshold indicating the tolerable distance (number of characters) between adjacent characters (which are from given abbreviation) in the definition candidates, and it allows a trade-off between efficiency and quality. In our system, DIST is set as 6.

---

INPUT: an abbreviation A, its inverted-index IDX, and local document D

---

WHILE IDX is not completely searched

---

TEMP:=GetAnOffsetSequence(IDX)

---

TEMP:=Sort(TEMP)

---

IF MaxAdjacentDistance(TEMP)<=DIST

---

OFFSET\_MIN:= GetMin(TEMP)

---

OFFSET\_MAX:=GetMax(TEMP)

---

DEF\_CAND:=GetString(D, OFFSET\_MIN,OFFSET\_MAX)

---

IF Length(DEF\_CAND)>Length(A)

---

DefCandCollection.Add(DEF\_CAND)

---

OUTPUT: DefCandCollection

---

**Fig. 4.** The abbreviation-definition matching algorithm

## 4 Abbreviation-Definition Disambiguation

We use SVM to disambiguate abbreviation-definition pairs. The key to the classification is to select appropriate features. We try to employ those features that effectively capture the distinction of the AC-DC pairs (AC indicates an abbreviation candidate, and DC indicates a corresponding definition candidate). Several features, including ‘context similarity’, are inspired by previous studies (Zahariev, 2004; Akira et al., 2001; Toole, 2000), while most of others such as ‘position relation’ and ‘distance’ are derived from our investigation of abbreviation formation regularities. The following features are considered in our system, including local features (without context information) and global features (using context information):

### 4.1 Local Features

**(1) Difference in Length.** This feature specifies the difference between the number of characters within AC and the number of words within DC. The value of this feature depends on the number of words in DC which maps to null string in AC.

**(2) Final character.** For the abbreviations of organization names, the final character contains rich information. E.g., the ‘院’ (institute/academy) of both abbreviation ‘中科院’ and its definition ‘中国/科学/院’ (Chinese Academy of Sciences) is usually a symbolic character for an organization name.

**(3) Character selection.** In many cases, the first character of a word in the definition tend to be selected into its corresponding abbreviation. E.g., ‘北京/大’ (Peking University) to ‘北大’. This feature records the position information of characters.

**(4) Word order.** This feature specifies whether or not the word order has been changed during abbreviating. In most cases, the word order will be kept the same.

**(5) Frequency selection.** This feature compares the universal frequency between selected characters and non-selected characters (obviously there are both selected characters and neglected ones within the DC, because AC is shorter than DC.).

## 4.2 Global Features

- (1)  **$w_{+1}$  and  $w_{-1}$  of AC.** A large part of abbreviations are from named entities, and their contextual words have its own trait. This feature is used to record the left/right word of AC and to investigate the above mentioned contextual trait.
- (2) **Length of AC's  $w_{+1}$  and  $w_{-1}$ .** This feature specifies the length of AC's contextual words.
- (3)  **$w_{+1}$  and  $w_{-1}$  of DC.** From an identical motivation form global feature (1), this feature is used to record the left/right word of DC.
- (4) **Length of DC's  $w_{+1}$  and  $w_{-1}$ .** This feature specifies the length of DC's contextual words.
- (5) **Frequency of AC.** This feature represents the frequency of AC in local document. To deal with the sparseness, only 4 values are defined: '1', '2-3', '4-5' and 'MORE THAN 5'.
- (6) **Frequency of DC.** This feature represents the frequency of DC in local document.
- (7) **Position relation.** This feature identifies the position relation between AC and its DC. Generally speaking, if an abbreviation and its definition cooccur in a document, it would be more probable that the definition precedes the abbreviation.
- (8) **Context similarity.** This feature estimates the similarity between AC context and DC context. Words appearing before and after an AC/DC are collected as context information. We use a window of size 5, and the window size need to be adjusted when sentence boundary is met. To measure the vector similarity, we use the cosine metric (Akira et al., 2001).
- (9) **Distance.** This feature is used to represent the distance (number of words) between AC and its DC. An AC usually has more than one corresponding DC, and in general, the DC with minimum distance is more probable to become the definition.

## 5 Evaluation

Our experiment data comes from the People's Daily corpus. The selected data contains 361479 sentences from 9117 documents, which is divided into two sets: one from 6078 documents, for training; and the other from 3039 documents, for testing. The annotated corpus is required both for training and testing, and we annotated the abbreviation-definition pairs in three steps: first, annotate abbreviations manually. Second, generate abbreviation-definition candidate pairs using the abbreviation-definition matching algorithm, which is described in section 3. Third, pick out the correct abbreviation-definition pairs manually.

For some abbreviations, there might be more than one correct definition. E.g., '人大' can respectively be abbreviated from both '人民/代表/大会' (National People's Congress) and '中国/人民/大学' (Renmin University of China). However, this multi-sense case seldom occurs in a single document so that we enforce the constraint of one sense per discourse for each abbreviation.

### 5.1 Experiment

To evaluate the system, we use precision, recall, and F-measure. They are respectively defined as follows (Akira et al., 2001):

$$P = \frac{w}{w+x+y} \quad (1)$$

$$R = \frac{w}{w+x+z} \quad (2)$$

$$F = \frac{2 * P * R}{P + R} \quad (3)$$

where w is the number of abbreviations expanded correctly; x is the number of abbreviations expanded incorrectly; y is the number of abbreviation candidates that were not abbreviations but were incorrectly expanded by the system; and z is number of abbreviations not detected as abbreviations by the system<sup>2</sup>.

In the SVM approach, data consisting of two categories is classified by dividing space with a hyperplane. For extended versions of the method, in general, the inner region of the margin in the training data can include a small number of examples, and the linearity of the hyperplane is changed to non-linearity by using kernel functions. (Vapnik, 1995; Cristianini et al., 2000)

During the tuning of the SVM model<sup>3</sup>, we select a linear function as the kernel function according to the experimental statistics, a part of which is shown in Table 1. It is interesting to note that the linear kernel function outperforms the radial basis kernel as well as the polynomial kernels, with the final F-measure of 84.3%. One possible reason might be the ‘over fit’ problem of training for the radial basis and polynomial kernels. On the other hand, the linear kernel function is extremely efficient both in training and classifying, which can also be noticed in Table 1.

In order to deal with data sparseness problem, we discard the features occurring only once in the training data.

**Table 1.** Experimental results upon a variety of SVM kernel functions<sup>4</sup>

Kernel Functions	P (%)	R (%)	F (%)	T-secs	C-secs
Linear Function	87.2	81.5	84.3	2.7	0.1
Radial Basis Function	88.8	76.6	82.3	26.2	8.0
Polynomial Functions	$(x \bullet y + 1)^2$	85.0	82.1	83.5	20.9
	$(x \bullet y + 1)^3$	77.7	87.8	82.5	19.3
					6.4

<sup>2</sup> Some abbreviations do not have corresponding definitions in local document, they would not be counted into z.

<sup>3</sup> We used the software SVM <sup>light</sup> (T. Joachims, 1999) developed by T. Joachims as the support vector machine.

<sup>4</sup> Where the *T-secs* is an indicator for *CPU seconds of training*, and *C-secs* stands for *CPU seconds of classifying*. Both the training and classifying are performed on a CPU of 1.6G HZ.

## 5.2 Comparison to Other Methods

To assess the performance of our SVM model, we implemented two other machine learning methods, which is described as follows:

**(1) Decision Tree.** The decision tree paradigm constructs classifiers by dividing the data set into smaller and more uniform groups, based on a measure of disparity which usually is entropy. The best choice of variable and threshold is the one that minimizes the disparity measures in the resulting groups. We used the C4.5 decision tree toolkit (<http://www.rulequest.com>) for our simulations.

**(2) Maximum Entropy.** The main idea of maximum entropy model (MEM) is to select probability distribution with highest entropy value based on the constraints. Highest entropy distribution gives the model minimum prejudice and maximum impartiality subject to the constraints, because any other selection would bring extra reasonable assumptions. We use the Generalized Iterative Scaling (GIS) algorithm to train feature weights and an additional Exponential Smoothing algorithm (Joshua Goodman, 2004) to penalize large weights.

The experimental results are shown in Table 2. For our abbreviation expansion task, the performance order of these methods is as follows:

**F-measure: SVM > MEM > DT**

**Precision: DT > SVM > MEM**

**Recall: MEM > SVM > DT**

Since they use exactly the same feature set, the comparison was strict. The number of abbreviation candidates that were incorrectly expanded is much less in Decision Tree method, thus it achieved the highest Precision of 88.8%. On the other hand, Maximum Entropy method achieved the highest Recall of 83.9%, with a dramatic increase of 168 on correct abbreviation expansions. With the F-measure of 84.3%, Support Vector Machine method presented a good compromise between Precision and Recall.

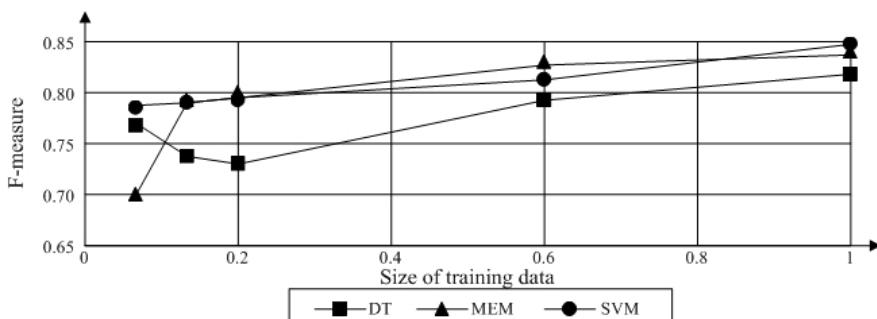
**Table 2.** Experimental results upon a variety of methods

Method	w	P (%)	R (%)	F (%)
Decision Tree	1467	88.8	75.6	81.7
Maximum Entropy	1627	83.0	83.9	83.5
Support Vector Machine	1580	87.2	81.5	84.3

Compared with the DT method, the improvement of MEM and SVM framework comes from their ability to more robustly combine features that do not form partitions of the event space, but instead overlap in arbitrarily complex ways. Compared with the MEM approach, the performance of the SVM approach is slightly superior within our abbreviation-definition disambiguation task, and the slight superiority might relate to our specific feature set. In some other fields of NLP (natural language processing) such

as WSD (word sense disambiguation), similar comparison result between the SVM approach and the MEM approach can be observed as well (Wu et al., 2004).

We also test how the size of training data affected our experimental results. Fig. 5 shows that the difference of the F-measure rates between the MEM and the SVM was larger when the size of the training data was decreased especially from 2/15 data-size to 1/15 data-size. This indicates that when the size of the training data is smaller, the SVM approach has a more stable performance on our abbreviation-definition disambiguation feature set. On the other hand, the curved line of DT goes down rapidly when the size of the training data is increased from 1/15 data-size to 1/5 data-size, which is abnormal. It indicates that the performance of decision tree method on our feature set is volatile in a relatively small size of training data.



**Fig. 5.** Experimental results when changing the size of the training data

Chang presented a HMM based error-recovery model (Chang et al., 2004) for abbreviation identification and root word recovery, which is introduced in section 1. In Chang's study, the performance of guessing definitions from abbreviations is about 51%. However, it needs to be taken into account that Chang uses a different corpus.

## 6 Conclusion and Future Work

In this paper, we proposed a SVM based framework for automatic abbreviation-definition identification in Chinese text. It reaches an encouraging performance according to the experimental result upon the People's Daily Corpus. Moreover, using exactly the same feature set, additional experiments show that SVM slightly outperforms MEM and DT methods, and it is more effective and stable when the size of training data is small.

Feature design is a crucial step of supervised abbreviation-definition identification. Experimental result indicates that our feature set utilizing context information has discriminative power. For the portability concern, only general characteristics of context information are employed in our system.

Our future work will consist in abbreviation expansion using supervised learning method for those abbreviations whose corresponding definitions are not in context.

## Acknowledgments

We would like to thank Galen Andrew, Wuguang Shi and Yun Xing for helpful comments on earlier versions of this paper.

## References

1. Vladimir N. Vapnik. *The Nature of Statistical Learning Theory*. Springer, 1995.
2. T. Joachims. Making large-scale support vector machine learning practical. In Scholkopf et al., 1999, pages 169-184.
3. Nello Cristianini and John Shawe-Taylor. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, 2000.
4. Jin-Shin Chang and Yu-Tso Lai. A Preliminary Study on Probabilistic Models for Chinese Abbreviations. ACL, 2004.
5. Zahariev, M. A (Acronyms). Ph.D. thesis, Simon Fraser University, 2004.
6. Akira Terada, Takenobu Tokunaga and Hozumi Tanaka. Automatic expansion of abbreviations by using context and character information. In Proc. of the Sixth Natural Language Processing Pacific Rim Symposium, 2001.
7. Joshua Goodman. Exponential Priors for Maximum Entropy Models. In Proc. of HLT-NAACL, 2004, pages 305-312.
8. Toole Janine. A hybrid approach to the identification and expansion of abbreviations. In Proc. of RIAO, 2000, pages 725-736.
9. Dekai Wu, Weifeng Su, and Marine Carpuat. A Kernel PCA method for superior word sense disambiguation. In Proc. of the 42nd Annual Meeting of the Association for Computational Linguistics, 2004.

# Clause Boundary Recognition Using Support Vector Machines

Hyun-Ju Lee, Seong-Bae Park\*, Sang-Jo Lee, and Se-Young Park

Department of Computer Engineering

Kyungpook National University

Daegu 702-701, Korea

{hjlee, sbpark, sjlee, separk}@sejong.knu.ac.kr

**Abstract.** This paper proposes a method for Korean clause boundary recognition. Clause boundary identification can be regarded as a three-class classification task, and it can be converted into a two-phase binary classification task. Then it is natural to apply SVMs to clause boundary recognition, since SVMs are basically binary classifiers. Specifically we first recognize the ending points of clauses, and then identify the starting points by considering the typological characteristics of Korean. In addition, since there is not a standard Korean corpus containing clause boundary information, we prepare a Korean clause identification dataset. In the evaluation, support vector machines yield the improvement of performance over memory-based learning or decision trees.

## 1 Introduction

Many natural language processing (NLP) problems can be considered as classification tasks. SVMs are well-known classifiers for their good generalization performance in high dimensional feature spaces, and have been applied to many pattern recognition problems. In the field of natural language processing, SVMs are applied to text categorization, and are reported to achieve high accuracy without falling into overfitting even with a large number of words taken as features [6]. Besides, SVMs are applied to named entity recognition [17], phrase chunking [8], and dependency structure analysis [9], and they all showed high performance.

Syntactic parsing is a process to find a structure of an input sentence, and is one of the critical processes of NLP. However, the ambiguities involved in the natural languages keep the syntactic parsing from becoming a practical and accurate tool for NLP applications. Because of the complexity of syntactic analysis, partial parsing has been studied as an alternative to full-sentence parsing. Because a clause is a grammatical unit which is just lower than a sentence, the clause identification is the deepest level of analysis among many possible levels of partial parsing.

---

\* Corresponding author.

In this paper, we propose a novel method for Korean clause identification. Clause identification is a process to recognize clauses in complicated sentences. This problem can be regarded as a three-class classification task. That is, a word in the sentence can be classified into one of three classes - a beginning of a clause (S), an ending of a clause (E), neither a beginning nor an ending (X). In addition, it can be converted into two binary-class tasks without loss of generality. Since SVMs are basically binary classifiers, it is natural to apply SVMs to our task.

Clause identification is, thus, performed in two phases. The ending points of each clause are first recognized, and then the starting points are identified in the second phase. In identifying Korean clauses, the language typological characteristics of Korean should be fully considered. Korean sentence is a partially free word-order language with forming basically Subject-Object-Verb order. A verb must come in the final position in the sentence, while the words before the verb are free or even omitted. Consequently it is relatively easy to recognize ending points of a clause, while starting points are much more difficult to find. Therefore the features for recognizing ending points must be different from those for starting. In addition, in the order of classification, the ending identification must be preceded to the starting.

## 2 Previous Work

There have been many studies for English clause identification. This problem was initially studied by a rule-based method. Ejerhed [4] showed that a parser can benefit from automatically identified clause boundaries in discourse. Leffa [10] formulated the linguistic rules for segmenting clauses in an English/Portuguese machine translation system. The identified clauses are classified into either a noun or an adverb. These rules for segmenting clauses are based on the text of English grammar and a set of 1,000 sentences taken at random from newspaper texts. However these rule-based systems require a large set of hand-crafted rules, implying a great deal of human labor. In order to avoid these efforts, Orsan [12] adopted a hybrid method for clause splitting in English text. First, he used a memory-based learning for predicting clause boundaries. The results of a machine learning algorithm are processed by a shallow rule-based module in order to improve the accuracy of the method.

The shared task of CoNLL-2001 was a clause identification employing machine learning methods. Six systems have participated in the task. They used a wide variety of machine learning techniques. Among the six participating systems, the system of Carreras and Marquez [1] outperformed all others. It used AdaBoost with decision trees as base learners.

For Korean, it is also a critical task to identify the clause unit. Yun [16] used rules for searching splitting points to divide a complex sentence into simple sentences. These rules were induced from only the POS tag information. For the purpose of handling zero anaphora, Kim et al. [7] splitted the complex sentences into elementary event units. She used the verbal ending classification

table related to complex sentence composition. Like the previous study, a set of hand-crafted rules and the dictionary information were used for Korean clause identification.

However, rule-based approaches have problems in coverage and consistency [9]. Since a great deal of time and human efforts are required, the limitation of their coverage must be augmented by some additional efforts in order to prepare the large set of hand-crafted rules. Since rules are usually related to one another, a confliction among them must be resolved.

On the other hand, machine learning methods can overcome the weakness of a rule-based system. However, there is no system that applies machine learning methods to Korean clause identification task. One of the reasons for this is that there is no Korean corpus containing clause boundary information. In this paper, we prepare Korean Clause identification dataset by extracting clause boundary information from a Korean syntactic tree tagged corpus.

### 3 Two-Phase Clause Boundary Identification

#### 3.1 Clause Identification

Generally the sentence length of agglutinative languages such as Korean is long. A verbal connective ending in agglutinative languages connects a predicate with other predicates. Therefore most long sentences are complex sentences, where a complex sentence consists of more than two clauses. The longer a sentence is, the more serious the complexity of analysis is. Complex sentence analysis has been a critical problem in natural language parsing. If we divide complex sentence into clauses, a full parsing problem can be simplified to be the dependency analysis between clausal units.

A clause is generally defined as a word sequence which contains a subject and a predicate. However Korean differs definitively from languages such as English and German in that a subject is dispensable in forming a sentence. The subject and other components except predicates are optional. The number of constituents in a sentence depends on the type of predicate. By the type of predicate, subjects can be appeared more than twice or often omitted. Thus, it is difficult to apply the previous definition on clause. Since the nucleus of the Korean sentence is a predicate, we define a Korean clause as “a maximum word sequence containing a verb”. Here is an example sentence<sup>1</sup>.

This sentence contains four clauses. The clauses are enclosed between brackets. That is, the brackets represent the clause boundary, and the number of brackets indicates the level of a clause. This sentence can be divided two clauses. The clause (1) is connected to the clause (2) subordinately. The clause (2) embeds a genitive clause “세계에서 힘하기로 유명한”, and then again, this clause embeds a noun clause “힘하기로”.

---

<sup>1</sup> The symbols and abbreviations used for simplification in this paper are as follows.

#: spacing, -: morpheme boundary, POST: postposition, NOM: nominative, END: ending, SUFF: suffix, COP: copular.

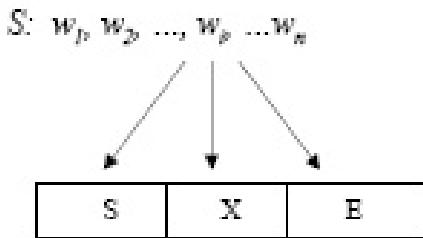
((남아메리카-에서#1천#2백#킬로미터#이상# 떨어지-어#있-으며) (1)  
 ((South America-*POST* # one thousand # two hundred # kilometer #  
 more than # far away-*END* # is-*END*)

(그#사이-의#바다가#(세계에서#(험하-기-로)#유명하-ㄴ)#드레이크해  
 협-아-다.)) (2)

(the # between-*POST* # sea-*NOM* # (world-*POST* # (rough-*SUFF-POST*)  
 # famous-*END*) # Drake Passage-*COP-END*)

*“Drake Passage which is far from South America more than 1200km is famous for roughness all over the world.”*

Since the objective of this study is to recognize the clause boundary in Korean sentence, the type and the level of clause are not considered. Clause identification is, therefore, to find both the starting point and ending point of a clause. This problem can be regarded as a classification task. In other words, the problem is to classify a word  $w_i$  in a sentence as one of three classes - a beginning of a clause (S), an ending of a clause (E), neither a beginning nor an ending (X). This idea is given in Figure 1.

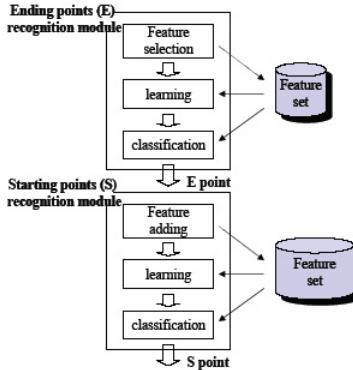


**Fig. 1.** The clause identification task regarded as a classification

### 3.2 Two-Phase Model

In Section 2, we defined a Korean clause as “a maximum word sequence containing a verb.” This definition argues that the nucleus of the clause is the predicate. Therefore, the number of clauses in a sentence is identical to that of predicates.

Syntactically Korean is a verb-final language with the verb appearing at the end of each clause. Therefore, the clause ending points are the verbs. That is, a verb and verbal endings must come in the final position of the sentence. It is relatively easy to find them out, while it is far more difficult to recognize the starting points of clauses. This is because the words before the verb are free-order or even they can be omitted. Due to these syntactical characteristics of Korean, any word before the verb can be a starting point. Thus, the ending



**Fig. 2.** The structure of Korean clause boundary recognition model

points are much clearer than starting points. The number and type of obligatory constituents depends on a vocabulary of predicate which is the ending of clause.

In this paper, the clause boundary identification system consists of two phases as shown in Figure 2. Above all, we make up of a feature set to recognize the ending points. Then, SVM is trained with this feature set. SVM determines whether a word in sentence is an ending or not. Secondly we add features to the previous feature set. These additional features are required to identify starting points. In the same manner, SVM is trained with the augmented feature set, and it classifies whether a word is a starting or not.

Since SVMs are basically binary classifiers, it is more natural to organize two binary classifiers than a three-class classifier. As far as concerned with the order of SVM application, we do not find starting points in the first place but ending points. In cases that the previous output is inputted to the following phase, the errors in the previous phase are propagated to the next phase. That is, the more previous errors are, the more errors are produced in the following phase. The previous errors, therefore, must be minimized. In Subject-Object-Verb order languages like Korean, the ending point is much clearer than starting. Therefore the possibility involving errors in the predicted ending is less than that in the predicted starting.

### 3.3 Features

Table 1 and Table 2 show the type and context of features for recognizing Korean clause. To determine whether the word  $w_i$  is an ending point or not, the lexicons, POS tags, chunk types, and ending features of  $w_i$  itself and surrounding words are used. For surrounding words, a word of left and right context is used for the lexicons, POS tags and chunk types, while a word of left context is used for ending feature. We only used the word that appears more than five times in the corpus for training. As a result, 4,171 words are selected as a vocabulary feature from total 16,838 words.

**Table 1.** Features for recognizing ending points

Context	Features			
	Word	POS	Chunk	End
$w_{i-1}$	$W_{i-1}$	$POS_{i-1}$	$C_{i-1}$	$E_{i-1}$
$w_i$	$W_i$	$POS_i$	$C_i$	
$w_{i+1}$	$W_{i+1}$	$POS_{i+1}$	$C_{i+1}$	

**Table 2.** Features for recognizing starting points

Context	Features				
	Word	POS	Chunk	End	Start
$w_{i-3}$	$W_{i-3}$	$POS_{i-3}$	$C_{i-3}$	$E_{i-3}$	$S_{i-3}$
$w_{i-2}$	$W_{i-2}$	$POS_{i-2}$	$C_{i-2}$	$E_{i-2}$	$S_{i-2}$
$w_{i-1}$	$W_{i-1}$	$POS_{i-1}$	$C_{i-1}$	$E_{i-1}$	$S_{i-1}$
$w_i$	$W_i$	$POS_i$	$C_i$	$E_i$	
$w_{i+1}$	$W_{i+1}$	$POS_{i+1}$	$C_{i+1}$	$E_{i+1}$	
$w_{i+2}$	$W_{i+2}$	$POS_{i+2}$	$C_{i+2}$	$E_{i+2}$	
$w_{i+3}$	$W_{i+3}$	$POS_{i+3}$	$C_{i+3}$	$E_{i+3}$	

The morphological information is sufficient for recognition of the clause ending, because only identifying the predicate of the sentence is required. But in case of the clause starting, the narrow range of morphological information is not enough. Thus, in order to recognize the clause starting points, we use more features and the broader range of context than those used to recognize the clause ending points. The features for recognizing the starting points are shown in Table 2. For surrounding words, three words of left and right context are used for the lexicons, POS tags, chunk types and ending features, while three words of left context are used for ending feature.

## 4 Experiments

### 4.1 Data

For the evaluation of the proposed method, Korean clause identification dataset is prepared. This dataset is derived from the parsed corpus, which is a product of STEP 2000 project supported by Korean government. The corpus consists of 12,092 sentences with 111,658 phrases and 321,328 words. One sentence in the corpus contains four clauses on average. Figure 3 shows an example sentence in the STEP 2000 parsed corpus. This parsed corpus is formatted with a binary tree structure. We extracted automatically the information of clause boundary from this corpus. Figure 4 shows an above example sentence in the dataset. The format of the dataset follows that of CoNLL-2001 dataset [15].

Each instance in the training and test data consists of five columns. The first column contains the lexicon, the second presents a part-of-speech tag which is

(S
(VP
(VP
(VP
(VP (NP /nq)+/jca
(VP
(NP (NP 1/nnc 2/nnc+/nbu ) /ncn) /pvg ))+
(AUXP /ecx+/px)+(AUXP /ecx /px ))+/ecs
(VP
(NP
(NP (MODP /mmd ) /ncn )+/jcm /ncn )+/jcs
(VP
(NP
(ADJP (NP /ncn )+/jca
(ADJP
(NP (ADJP /paa )+/etn )+/jca
/ncps+/xsm ))+/etm
/nq )+/jp )))+/ef+./sf )

**Fig. 3.** An example sentence of the STEP 2000 parsed corpus

	nq	B-NP	S	X	South America
	jca	I-NP	X	X	<i>Postposition: POST</i>
1	nnc	B-NP	X	X	one thousand
2	nnc	I-NP	X	X	two hundreds
	nbu	I-NP	X	X	kilometer
	ncn	I-NP	X	X	more than
	pvg	B-VP	X	X	far away
...	...	...	...	...	...
	nq	B-NP	X	X	Drake Passage
	jp	B-VP	X	X	<i>Copular: COP</i>
	ef	I-VP	X	E	<i>Ending: END</i>
.	sf	O	X	X	

**Fig. 4.** An example of Korean clause identification dataset

based on the KAIST tag set [18]. The third column contains the chunk tag. It indicates one of nine chunk types. This chunk tag is based on STEP 2000 Korean chunk data [13]. The fourth presents the starting points of clauses, and the fifth contains ending features: a beginning of a clause (S), an ending of a clause (E), neither a beginning nor an ending (X).

## 4.2 Evaluation

We use ten-fold cross validation to estimate the performance of our system. In each fold, the corpus is divided into two parts: training (90%) and testing (10%).

Also the evaluation measure is defined as follows.

$$\text{Accuracy} = \frac{\text{Correctly recognized words}}{\text{Total words}} \times 100(\%)$$

$$\text{Recall} = \frac{\text{Correctly recognized ending (starting) points}}{\text{Clause ending (starting) points}} \times 100(\%)$$

$$\text{Precision} = \frac{\text{Correctly recognized ending (starting) points}}{\text{Recognized ending (starting) points}} \times 100(\%)$$

$$F_{\beta} - \text{score} = \frac{(\beta^2 + 1) \cdot \text{Recall} \cdot \text{Precision}}{\beta^2 \cdot \text{Recall} + \text{Precision}}$$

To evaluate the performance of SVM in perspective, we compare SVM with some well-known machine learning algorithms: (1) decision tree, (2) memory-based learning. We use SVM<sup>light</sup> [5] for support vector machines and C4.5 release 8 [14] for decision tree induction while TiMBL [3] is used for memory-based learning. The experimental results are shown in Table 3. We set  $\beta$  of F-score to 1 in order to impose a identical weight on recall and precision.

The experimental results shown in Table 3 assert that the clause start recognition is more difficult than the clause end recognition. This is because the morphological information is enough to detect the clause end but is not for the clause start. Support vector machines present better performance than the memory-based learning and the decision tree. Since it is sufficient to use morphological features for the ending points, there is not a great difference of performance among three systems. For starting points, the F-score of support vector machine is 82.30, so the improvement is 3.45 over the memory-based learning and 10.98 over the decision tree. Since it is very difficult to find starting points in the Korean clause, it is significant that performance of the starting points recognition is higher than other methods.

In order to recognize the clause starting point, we used more types of features and broader range of context than those used to recognize the clause ending point. In the case that there is a starting point in the left context of  $w_i$ , we add features of word, POS tag, and chunk for the corresponding ending point per a starting point. Table 4 shows how much the performance is improved by adding these features. By augmenting features of ending, the F-score for the starting

**Table 3.** The experimental results of various machine learning algorithms

Task	Measure	SVM	MBL	Decision Tree
End	Accuracy	<b>99.55 ± 0.04%</b>	99.51 ± 0.04%	99.55 ± 0.05%
	Precision	97.71 ± 0.25%	<b>98.00 ± 0.29%</b>	97.98 ± 0.25%
	Recall	<b>99.20 ± 0.26%</b>	98.50 ± 0.30%	98.76 ± 0.30%
	F-score	<b>98.45 ± 0.16</b>	98.25 ± 0.15	98.37 ± 0.17
Start	Accuracy	<b>96.54 ± 0.09%</b>	95.61 ± 0.13%	93.75 ± 0.16%
	Precision	<b>88.55 ± 0.55%</b>	79.39 ± 0.83%	68.61 ± 0.84%
	Recall	<b>76.87 ± 0.61%</b>	78.33 ± 0.44%	74.26 ± 0.62%
	F-score	<b>82.30 ± 0.46</b>	78.85 ± 0.49	71.32 ± 0.61

**Table 4.** The improvement of the performance for the starting point recognition with augmenting features of ending

	Recall	Precision	Accuracy	F-score
Before	76.87%	88.55%	96.54%	82.30
After	77.29%	89.10%	96.64%	82.77
Difference	+0.42%	+0.55%	+0.10%	+0.47

point recognition was improved by 0.47. This improvement is statistically significant, because the values are estimated by ten-fold cross validation.

## 5 Conclusions

We have proposed a novel method for a Korean clause identification based on the support vector machines. The clause boundary identification is in general regarded as a three-class classification task, but the proposed clause boundary identification system was composed of two phases, and the three-class task is transformed into two binary-class tasks. Since SVMs are basically binary classifiers, it is natural to adopt the proposed method. Thus, it is not required to expand SVMs to solve the three-class classification problem.

In the order of classification, we first recognize the ending points of clauses, and then identify the starting points. This procedure is set by considering the typological characteristics of Korean, which are partially free word order with forming basically Subject-Object-Verb order and the words before the verb are often omitted. Thus, in order to recognize the clause starting points we used more types of features and broader range of context than those to recognize the clause ending points.

The proposed method was evaluated on Korean clause identification dataset. The experimental results showed that our clause boundary recognition system achieved 82.30 of F-score for clause starting recognition and 98.45 for ending. This fact implies that a clause starting detection is more difficult than a clause ending detection. We think that this could be because only the narrow range of morphological information is not enough for the recognition of the clause starting while it is sufficient for the clause ending.

In addition, support vector machines presented higher performance than other machine learning methods such as a memory-based learning and a decision tree. Through our experiments, we showed that SVMs showed the high performance for the clause boundary recognition as in other natural language processing tasks.

## Acknowledgements

This work was supported by the Korea Research Foundation Grant funded by the Korean Government (KRF-2005-202-D00465).

## References

1. X. Carreras and L. Marquez, "Boosting Trees for Clause Splitting," In *Proceedings of CoNLL-2001*, pp. 73–75, 2001.
2. X. Carreras, L. Marquez, V. Punyakanok, and Dan Roth, "Learning and Inference for Clause Identification," In *Proceedings of ECML '02*, pp. 35–47, 2002.
3. W. Daelemans, J. Zavrel, K. Sloot and A. Bosch, *TiMBL: Tilburgh Memory Based Learner, Version 4.1, Reference Guide*, ILK 01-04, Tilburgh University, 2001.
4. E. Ejerhed, "Finding Clauses in Unrestricted Text by Finitary and Stochastic Methods," In *Proceedings of ANLP '88*, pp. 219–227, 1998.
5. T. Joachims, *Making Large-scale SVM Learning Practical*, Technical Report LS8-Report 24, Universität Dortmund, 1998.
6. T. Joachims, "Text Categorization with Support Vector Machines: Learning with Many Relevant Features," In *Proceedings of ECML '98*, pp. 137–142, 1998.
7. K.-J. Kim, Y.-H. Song, and J.-H. Lee, "Implementation of the System Dividing Simple Sentences from Embedded Sentence in Korean," In *Proceedings of the Fifth Conference on Hangul and Korean Language Information Processing*, pp. 25–34, 1993.
8. T. Kudoh and Y. Matsumoto, "Use of Support Vector Learning for Chunk Identification," In *Proceedings of CoNLL-2000*, pp. 142–144, 2000.
9. T. Kudoh and Y. Matsumoto, "Japanese Dependency Structure Analysis Based on Support Vector Machines," In *Proceedings of EMNLP/VLC-2000*, pp. 18–25, 2000.
10. V. Leffa, "Clause Processing in Complex Sentences," In *Proceedings of LREC '98*, pp. 937–943, 1998.
11. A. Molina and F. Pla, "Clause Detection using HMM," In *Proceedings of CoNLL-2001*, pp. 70–72, 2001.
12. C. Orsan, "A Hybrid Method for Clause Splitting in Unrestricted English Text," In *Proceedings of ACIDCA 2000*, pp. 129–134, 2000.
13. S.-B. Park and B.-T. Zhang, "Text Chunking by Combining Hand-Crafted Rules and Memory-Based Learning," In *Proceedings of the ACL 2003*, pp. 497–504, 2003.
14. R. Quinlan, *C4.5: Programs for Machine Learning*, Morgan Kaufmann Publishers. 1993.
15. Erik F. Tjong Kim Sang and Herve Dejean, "Introduction to the CoNLL-2001 Shard Task: Clause Identification," In *Proceedings of CoNLL-2001*, pp. 53–57, 2001.
16. Yun Seung, *A Study on Methods of Dividing Complex Sentences in the Korean Language*, MS thesis of Dept. of Korean Linguistic Informatics, Yonsei University, 2001.
17. K. Takeuchi and N. Collier, "Use of Support Vector Machines in Extended Named Entity Recognition," In *Proceedings of CoNLL-2002*, pp. 119–125, 2002.
18. J.-T. Yoon and K.-S. Choi, *Study on KAIST Corpus*, CS-TR-99-139, KAIST CS, 1999.

# Large Quantity of Text Classification Based on the Improved Feature-Line Method

XianFei Zhang<sup>1</sup>, BiCheng Li<sup>2</sup>, WenBin Mu<sup>1</sup>, and Yin Liu<sup>3</sup>

<sup>1,2,3</sup>Zhengzhou Information Science and Technology Institute  
No. 835, P.O. BOX 1001, Zhengzhou, Henan, 450002, China

<sup>1</sup> zhangxianfei2003@126.com

<sup>2</sup> 1bclm@263.net

<sup>3</sup> xiaotuqinmo@yahoo.com.cn

**Abstract.** Feature-Line Method deems that a line between two points in the same class of space represents the space feature better than a single point. However, it brings faults in the classification results in terms of distance only. Here coefficient was put forward to eliminate the influence of the off-group point to classification, which was also combined with the central distance of class, then formed the improved algorithm, which is used in two different capacity document repositories. The results of experiment show that the improved algorithm support large document repositories very well, and it can be used in large-scale text classification and text retrieval.

## 1 Introduction

With the high-speed development of information technology and the usage of information management technology, more and more data have been accumulated. The automatic classification method of large quantity of data has become a very important task in Data Mining, in which Web (especially Text) Classification method seems more important than others. Text Classification<sup>[1]</sup> is an important field in Information Research, and enjoys a wide-ranging usage in a lot of aspects such as Text Retrieval<sup>[2]</sup>, Searching Engine and Information Filter<sup>[3]</sup>.

Feature Line Method was advanced and widely used in Face Identification field in recent years. It deems that a line in one class of space can better represent feature of whole space than a single point. Because both paper and face which is taken for vector in space of high-dimension, and they have essential connection with each other, so it can be used in text classification and text retrieval<sup>[2]</sup>.

The contribution of this paper concludes:

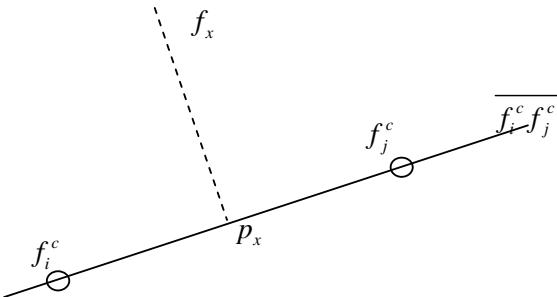
- the use of the feature line method in text classification, extend domain of the algorithm;
- put forward concept of “dimension trap” and “class central distance”, give the weight coefficient and the final improved algorithm;
- form the concrete step of the improved algorithm using fast calculation;
- experiment different document repository and bedded document repository using different classification algorithms;

This paper is arranged as follows. The second part makes a brief introduction about the idea of the feature line method. The third part concentrates on the shortcomings of the feature line method, and puts forward the improved algorithm combining with the concept of Weighted Coefficient and Class Central Distance. The forth part is concrete steps of the improved algorithm using fast calculation. The relevant test data, capability analysis and conclusion will be given in the last.

## 2 Nearest Feature Line Algorithm

Nearest Feature Line (NFL) is a new classification method in recent years. At first, it was put forward by stan z.Li<sup>[4]</sup> in Face Identification and applied in lots of fields. Its idea is structuring feature line (line of two sample points in class), researching and classifying according to the shortest distance from sample point to feature line.

It supposes that feature points  $f_i^c$  and  $f_j^c$  are in feature space  $f^c$ , m is dimension of  $f^c$ . Definite distance of two feature points is  $\Delta f = \|f_i^c - f_j^c\|$ . Line  $\overline{f_i^c f_j^c}$  is called the feature line.



**Fig. 1.** The projection of feature point in feature line

If  $f_x^c \in \overline{f_i^c f_j^c}$ , then  $f_x^c = (1-\lambda)f_i^c + \lambda f_j^c$ ,  $\lambda \in R$ . Marking  $T^c = \{f_i^c | 0 \leq i \leq N_c\}$ ,  $T^c$  is the set of  $N_c$  feature points in  $f^c$ , so these  $N_c$  feature points can form  $K_c$  feature lines,  $K_c = N_c(N_c - 1)/2$ . Marking  $S = \{\overline{f_i^c f_j^c} | 0 \leq i, j \leq N_c, i \neq j\}$ ,  $S$  is one of sub-space of the whole space. If there are  $Q$  classes in whole space, the set of feature line can be denoted  $N_{total}$  ( $N_{total} = \sum_{c=1}^Q K_c$ ). The needed classified sample is  $f_x$ . Marking

$Dist(f_x, \overline{f_i^c f_j^c})$  is the distance from  $f_x$  to  $\overline{f_i^c f_j^c}$ ,  $p_x$  is the projection of  $f_x$ ,  $Dsit(f_x, \overline{f_i^c f_j^c}) = \|f_x - p_x\|$ , as  $p_x \in \overline{f_i^c f_j^c}$ , we can get  $p_x = (1-\lambda)f_i^c + \lambda f_j^c = f_i^c + \lambda(f_j^c - f_i^c)$ . As  $\overline{f_x p_x}$  is perpendicular with  $\overline{f_i^c f_j^c}$ , so

$$(f_x - p_x) \bullet (f_i^c - f_j^c) = 0 \quad (1)$$

The sign “ $\bullet$ ” represents inner product, so we can get

$$\lambda = [(f_x - f_i^c) \bullet (f_j^c - f_i^c)] / [(f_j^c - f_i^c) \bullet (f_j^c - f_i^c)] \quad (2)$$

NFL algorithm first calculates distance from  $f_x$  to the feature line  $\overline{f_i^c f_j^c}$  of every class, and gets  $N_{total}$  results, then, arrange these  $N_{total}$  results according to their values. According to the arrangement, it divides  $f_x$  into the corresponding class.

Generally speaking, the amount of calculation of NFL algorithm is very heavy<sup>[5]</sup>, especially in case of high-dimension feature space and many sample classes, the problem of computational complexity seems more seriously.

### 3 Algorithm Analysis and Improvement

#### 3.1 Analysis

NFL algorithm may appear such situation that distance from the needed classified sample to some feature beside the class is farther than distance from it to samples inside the class. In figure 2, if not considering influence of class, feature line with the shortest distance to unknown sample  $f_x$  is  $\overline{f_i^c f_j^c}$ , but in fact, it should be  $\overline{f_i^c f_j^c}$  in its class. If just judging by distance, it would lead to wrong result.

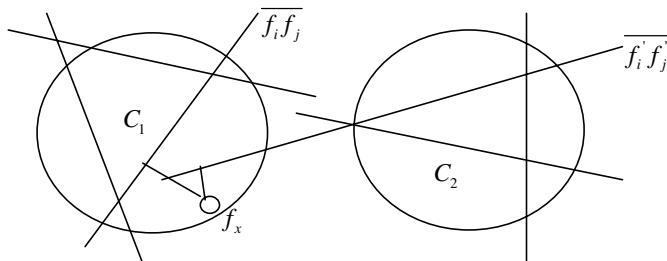


Fig. 2. Situation of wrong judgment

As average value of class can reflect its distribution, if the distance from unknown sample to some class average value is very far, it should not belong to this class.

However, there may be a situation that distance of feature line from unknown sample to some feature line in its class is very short. If we use NFL to classify at that moment, the unknown sample would be classified that class. This result is obviously unreasonable. So it's necessary to restrict the range of feature line's value.

### 3.2 Correlatives Concept

First, it calculates distance from the needed classified sample  $X$  to the average value  $X_a$  of class F. Generally speaking, distance involves all attributes. It deems that influence of those features to distance is almost equal. But it would give rise to faults if depending on the equal influence on classifying.

**Definition 1 “Dimension Trap”:** When judging an unknown sample's class by its attributes, it can lead to fault; this phenomenon is called “Dimension Trap”.

So it's necessary to add a value for each attribute. Different attribute have different affection on classifying.

**Definition 2 Class Central Distance:** the distance of samples in class to class center is called class central distance. It can be described as the following formula:

$$\|X - X_a\| = \sqrt{\sum_{j=1}^n (\omega_j (x_j - X_a))^2} \quad (3)$$

$\omega = (\omega_1, \omega_2, \dots, \omega_n)$  is value vector.  $X = (x_1, x_2, \dots, x_n)$ ,  $X_a = \sum_{i=1}^{|F|} X_i$ ,  $X_i$  is the known sample.

Because values of off-group<sup>[6]</sup> points can affect maximum of  $\|X - X_a\|$ , in order to eliminate this effect, it's necessary to quote the concept of weighted coefficient.

**Definition 3 weighted coefficient:** In order to eliminate the influence of the off-group value, we define the following coefficient as weighted coefficient.

$$C = 1 / (1 + \sqrt{\sum_{i=1}^n y_i^2}) \quad (4)$$

Here,  $y_i$  is the variance of  $W_i$  in training samples,  $W_i$  is row vector.  $i = 0, 1, 2, \dots, n$ .

### 3.3 Bases of Improving

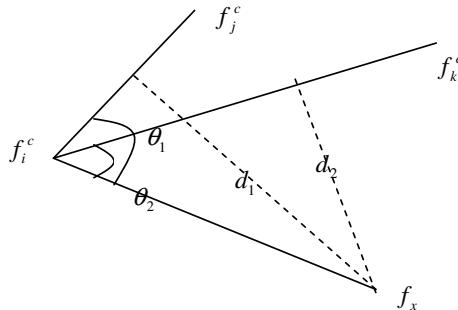
Because the distance which we adopt is decided by the absolute value of difference of its weight, so the value of this distance can be influenced by off-group degree of its weights. In order to avoid the influence of off-group value to maximum of distance, the weighted coefficient which was introduced should be correlative to variances of each weight, and direct ratio with reciprocal of variance.

If  $\|X - X_a\| \leq C \max\|X_i - X_a\|$ ,  $X$  may belong to this class; and if  $\|X - X_a\| > C \max\|X_i - X_a\|$ , then  $X$  should not belong to it.

## 4 The Improved Algorithm

### 4.1 Fast Calculation

In calculating NFL of class  $f^c$ , it need calculate the projection of the feature point first, and then calculate the distance from feature point to its projection. There are twice “•”, once product of number and vector and once divide of number in one projection calculating, which can be seen  $4 * K_c$  times “•” operation and  $K_c$  times divide operation. So, we can adapt fast calculation method in calculating NFL. If  $f_i^c, f_j^c$  and  $f_k^c$  are feature points in class  $f^c$ ,  $f_x$  is the unknown sample, so  $\overline{f_i^c f_j^c}$  and  $\overline{f_i^c f_k^c}$  are two feature lines in  $f^c$ .  $d_1 = \|\overline{f_i^c f_x}\| \sin \theta_1$  is the distance of  $f_x$  to line  $\overline{f_i^c f_j^c}$ ,  $d_2 = \|\overline{f_i^c f_x}\| \sin \theta_2$  is the distance of  $f_x$  to line  $\overline{f_i^c f_k^c}$ .



**Fig. 3.** Calculation of the minimum distance

If  $d_1 \geq d_2$ , we can get the following formula:  $\|\overline{f_i^c f_x}\| \sin \theta_1 \geq \|\overline{f_i^c f_x}\| \sin \theta_2$  and  $\|\overline{f_i^c f_x}\| \cdot |\cos \theta_1| \leq \|\overline{f_i^c f_x}\| \cdot |\cos \theta_2|$ , there are the following formula:

$$|\cos \theta_1| = |(f_j^c - f_i^c) \bullet (f_x - f_i^c)| / (\|\overline{f_i^c f_x}\| \cdot \|\overline{f_i^c f_j^c}\|) \quad (5)$$

$$|\cos \theta_2| = |(f_k^c - f_i^c) \bullet (f_x - f_i^c)| / (\|\overline{f_i^c f_x}\| \cdot \|\overline{f_i^c f_k^c}\|) \quad (6)$$

So, we can get:

$$d_1 \geq d_2 \Leftrightarrow |(f_j^c - f_i^c) \bullet (f_x - f_i^c)| / \|f_i^c f_j^c\| \leq |(f_k^c - f_i^c) \bullet (f_x - f_i^c)| / \|f_i^c f_k^c\| \quad (7)$$

Since  $\|f_i^c f_j^c\|$  can be calculated and deposited before identification, so we can reduce the calculation of training sets. When dimension of feature space is large, the whole calculation using the fast calculation is a quarter of the former.

## 4.2 Concrete Steps of the Improved Algorithm

Here is the concrete step of text classification using the improved feature line algorithm.

- (1) Calculating the central point of known samples by  $X_a = \sum_{i=1}^{|F|} X_i$ ;
- (2) Calculating the weighted coefficient using the formula.  $C = 1 / (1 + \sqrt{\sum_{i=1}^n y_i^2})$ ;
- (3) Calculating the distance of the needed classified sample  $X$  and average value of class  $F$   $\|X - X_a\|$  using the formula  $\|X - X_a\| = \sqrt{\sum_{j=1}^n (\omega_j (x_j - X_a))^2}$ ;
- (4) On condition that the inequality of  $\|X - X_a\| \leq C \max \|X_i - X_a\|$ , calculating the feature line distance between unknown sample point and NFL of class F;
- (5) Arranging all the satisfied feature line distances  $d_1, d_2, \dots, d_n$  using fast calculation;
- (6) Choosing the class of the k sample points which has the shortest distance as the needed identified sample's class.

The improved algorithm identifies class twice, so it reduces the amount of calculation and improves classification accuracy.

## 5 Experiments

### 5.1 Choosing of Document Repository

Our experiment method is fixing certain class, and then using different document repositories to test capability of classifier. Here two document repositories we adopted are DR -1 (document repository-1) and DR-2. DR-1 is 500 pieces of texts which come from <http://www.xinhuanet.com>. It includes five kinds, which are politics, military, economy, culture and PE. Each class has average documents. DR-2 is a double deck document repository which includes 14190 pieces of document. The first layer contains 12 classes and the second layer contains 60 classes. The concrete class of DR-2 is as follows:

**Table 1.** DR-2 and the instance of its bedded

finance and economics 819	fortune	19	automobile 590	knowledge	118
	finance	267		news	258
	corporation	164		run	176
	character	64		policy	38
	consume	91		carve out	39
	security	214		management	412
terrain 150	city	71	ability person 608	hunt up	39
	custom	47		record	39
	cate	32		salary	40
computer 2943	virus	631	PE 2805	text	39
	technology	574		baskeball	962
	software	426		ping pong	112
	network	517		chess	50
	game	102		swimming	94
	commerce	693		track/field	84
house property 935	building	76	sanitation 1406	tennis	131
	house	433		badminton	555
	fitment	172		football	817
	room	254		health	625
education 848	print	48	art 546	male/female	335
	employment	146		mentality	63
	examination	173		medication	383
	study abroad	67		antique	51
	training	21		aesthetics	84
	campus	226		literature	153
science 1040	recruit	127	entertain- ment 1500	stage	185
	ancient	183		music	73
	life	459		film	499
	chronometer	169		music	500
	natural	229		skill	501

All documents we adopted are given by word frequency matrix mode. In pretreatment, participle tool is system of ICTCLAS<sup>[7]</sup> which comes from China Science Academy, word warehouse in which adopts standard of Beijing University. Participle label adopts the first stair label. Figure and punctuation were taken out in result of participle. The reduce-dimension of training documents is the function of  $\chi^2$ , coefficient of which is  $\rho = 0.9$ .

## 5.2 Process of Experiment

Our experiments were adopted VC++ 6.0 and has been compiled in Windows XP. Hardware configure is P4 2.4 CPU and 512M memory. Classical algorithms, such as KNN, Bayes<sup>[8]</sup> and SVM, were adopted to compare with our improved algorithm in different document repositories. Bayes algorithm adopts multinomial model. SVM adopts linearity nucleus function. All experiments adopt cross-verify method to evaluate the classification results. That is dividing documents into five parts (one of them is

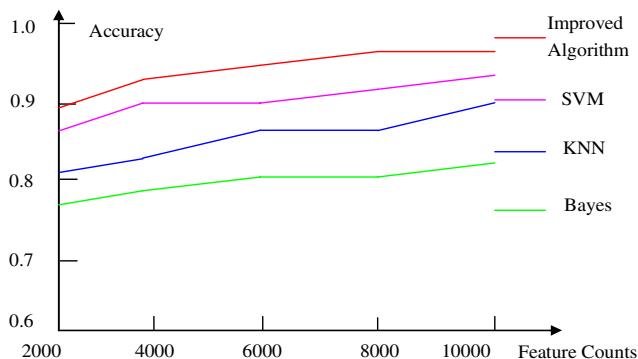
used to be test sets, and the rest are training sets), and circulating the operation five times. Average result of five times classifying is the final result.

### 5.3 Experiment Result and Algorithm Evaluating

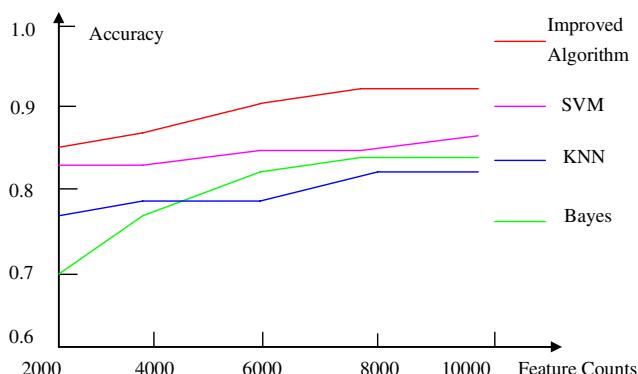
For convenience of comparing, we adjust four classifier make the same recall, and then just compare accuracy of classifier. The final result of experiment is as follows.

**Table 2.** Different classifier's accuracy of the four algorithms

	Improved Algorithm	KNN	Bayes	SVM
DR-1	0.7632	0.6884	0.7127	0.7593
DR-2:12	0.9457	0.8547	0.8983	0.9182
DR-2:60	0.9015	0.7865	0.8616	0.8472



**Fig. 4.** Accuracy curve of four classifiers in DR-2:12



**Fig. 5.** Accuracy curve of four classifiers in DR-2:60

Result of experiments is as follows: runtime of four classifiers in DR-2 is a little longer than in DR-1, but it can be solved by advancing the hardware configures of computer. In accuracy of classifier aspect, on condition of the same recall, accuracy of the improved algorithm has been greatly advanced than other algorithms. In concrete comparison we can find that the improved algorithm almost has no advantages, however, in large-scale document repository, it has a good work. In DR2:12, the effect of improved algorithm is best, here its classified accuracy is higher 3% than SVM algorithm and nearly 10% than the other algorithms. In DR2:60, advantage of the improved algorithm is not evidence as in DR2:12. That is because distributing of documents in DR2:60 is very unbalanced.

## 6 Conclusion and Future Work

When document repository is not large enough, classical classification algorithms have quite well classified result, but when scale of document repository is sizeable, the effect of classification is not so good. While our improved feature line algorithm has a very satisfied result in large-scale document repository, so it can well be used in large document repository Text Classification, Text Retrieval and Text Mining. Future work should be concentrated on other better classification methods which can be used in large-scale Text Processing.

## References

1. Lewis,D.D.Naive Bayes at forty: The independence assumption in information retrieval. Machine Learning:ECML=98,Tenth European Conference on Machine Learning. (1998) 4-15.
2. Frakes W B, Baeza-Yates R. Information retrieval:data structures and algorithms. Prentice Hall,Englewood Cliffs, NJ, USA, 1992.
3. Belkin N, Croft WB. Information filtering and information retrieval: two sides of the same coin. Communications of the ACM, 1992, 33(12): 29-38.
4. S.Z.Li and J.Lu, Face Recognition Using the Nearest Feature Line Method, IEEE Trans Neural Networks , Vol.10, No 2, (1999) 439-443..
5. Zhao Li Qi Wei Li Ziqing. Camera Lens Researching Based on the Improved NFL algorithm. Software Transaction. Vol.13, (2002) 0586-05.
6. Olivia Parr Rud. Data Mining Cookbook: Modeling Data for Marketing, Risk, and Customer Relationship Management ISBN 7-111-12221-6.
7. Huaping Zhang. Chinese Lexical Analysis Using Hierarchical Hidden Markov Model. Second SIGHAN workshop affiliated with 41th ACL. Sapporo Japan, July, (2003) 63-70
8. Andrew McCallum, Kamal Nigam. A Comparison of Event Models for Naïve Bayes Text Classification. AAAI/ICML-98 Workshop on Learning for Text Categorization[C]. Menlo Park, CA.

# Automatic Multi-level Summarizations Generation Based on Basic Semantic Unit for Sports Video

Chen Jianyun<sup>1</sup>, Zhao Xinyu<sup>1</sup>, Duan Miyi<sup>1</sup>, Wu Tingting<sup>1</sup>, and Lao Songyang<sup>2</sup>

<sup>1</sup> Beijing Institute of Graphics, 100029, PRC

<sup>2</sup> National University of Defense and Technology, Changsha 410073, PRC

**Abstract.** Sports video has been widely studied due to its tremendous commercial potentials. Despite encouraging results from various specific sports games, it is almost impossible to extend a summarization system for a new sports game due to the lack of sports video modeling. In this paper, we automatically generate multi-level summarizations for sports video based on Basic Semantic Unit (BSU), which is specially presented for sports video and can be extended to a new sports game. The results of the preliminary experiments indicate that our work provides a generic, novel and effective solution for automatic multi-level sports video summarizations.

## 1 Introduction

Sports video is widely distributed over networks and people are drowning in lots of video information. Therefore, automatic summarization generation of sports video to help people browse, retrieve and consume sports video has become more and more important.

Many approaches towards video summarization have been reported in literature, e.g. [1-5]. Most methods are developed for particular sports game, specific features, or specific environments. For example, some require MPEG audio bit stream processing [1], some are restricted to color analysis [2], to object tracking [3], or to event detection [4]. And some are focused on replay generation [5]. But few attentions have been paid to the framework of video summarization on the basis of the characteristics of sports video. In this paper, we present the concept of Basic Semantic Unit especially for sports video and a novel generic multi-level summarization framework based on BSU.

The rest of this paper is organized as follows. In Section 2, we explain the idea of Basic Semantic Unit and model the general sports video summarization framework. Furthermore we classify BSU of sports video and give the relations between all kinds of BSU and multi-level summarizations. Then in section 3, models and methods of every-level summarization are discussed in detail. And we present the summarization demo by soccer video and other sports video in section 4. At last Section 5 conclude this paper.

## 2 Summarizing Sports Video Based on Basic Semantic Unit

Comparing with news or film videos, sports video has well-defined content structure and domain rules [6]. Although a typical game lasts several hours, only part of it is of

importance in terms of semantic understanding or perceptive appreciating. These important parts occur semi-periodically or periodically, such as serve in tennis, pitching in baseball and so on.

## 2.1 Definition and Classification of Basic Semantic Unit for Sports Video

We abstract those recurrent important semantic parts during the game as “Basic Semantic Unit”, abbreviated to “BSU”, and decompose sports video into a regular set of BSU with different types and grains, which are the basis of summarizing sports video. And then the BSU of sports video can be classified into Audio BSU, Object BSU, View BSU, Scene BSU and Event BSU, etc.

### Audio BSU

Usually, sports video is composed of game, advertisement and studio. The soundtrack of game is noisy speech; the soundtrack of advertisement is the mixture of speech and music, while the soundtrack of studio is pure speech. These three kinds of soundtracks correspond to three kinds of Audio BSU: ***Game\_AudioBSU***, ***Ad\_AudioBSU*** and ***Studio\_AudioBSU***.

### View BSU

There are a fixed number of cameras in playing field. We define view as a camera with a specific angle and position in sports video. There are three main types of view BSU in sports video, namely ***Loose\_ViewBSU***, ***Medium\_ViewBSU*** and ***Tighe\_ViewBSU***, abbreviated to Loose View, Medium View and Tight View, as shown in figure 1. These three kinds of views differ in their shooting scale and the difference is usually reflected as the ratio of the field area.

### Object BSU

Only a limited number of object types are observed in sports video which include ball, player, referee, coach, goal, sideline, corner arc (and other field markings), and on screen captions. Automatic object segmentation and tracking is a much more difficult. For this reason in this paper we only select four types of object BSU: ***Caption***, ***Goal***, ***Referee*** and ***Corner\_arc***, as shown in figure2.

### Scene BSU

Sports game has two kinds of semantic states: in play and out of play. According to the game rules [6], the ball is out of play in the following circumstances: one is that it has wholly crossed a boundary line whether or not on the ground or in the air. The



**Fig. 1.** *Loose\_ViewBSU*, *Medium\_ViewBSU* and *Tight\_ViewBSU* in sports video

other is that the game has been stopped by the referee. And the game is in play at all other times from the start of the match to the finish. That is to say that it is in play when the ball within boundaries of the field and the game is not stopped by the referee. Therefore, we define two kinds of Scene BSU: *In\_Play\_SceneBSU* and *Out\_of\_Play\_SceneBSU*, abbreviated to In\_Play and Out\_of\_Play. And it is easily found that three kinds of View BSU constitute In\_Play and Out\_of\_Play with temporal relations. Figure 3 gives the instants of Scene BSU in soccer, volleyball and tennis video.

### Event BSU

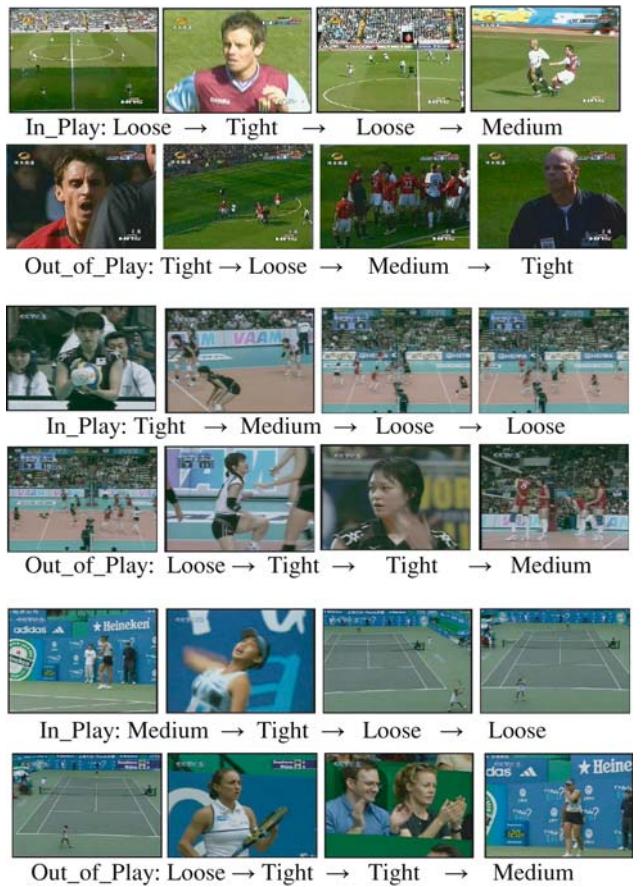
Event BSU is the essential content from the sports video on the user's demand. Figure 4 contain two simplified diagrams of scoring event which is indexed by some low-level cues such as appearance of goal area, updating of scoreboard, slow-motion replays and so on. So we detect semantic event using Bayesian network which integrates the low-level video content cues and then form the summarization of event.

## 2.2 The Framework of Multi-level Summarizations Based on BSU for Sports Video

We develop a framework for modeling automatic multi-level summarizations generation for sports video based on BSU, as



**Fig. 2.** Object BSU in sports video



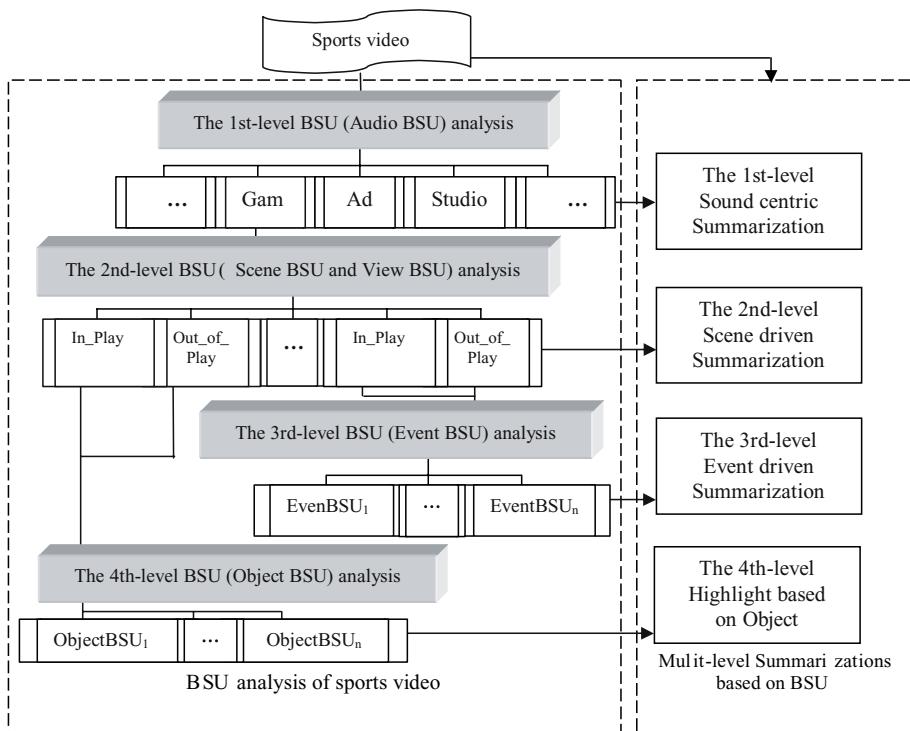
**Fig. 3.** Three kinds of View BSU constitute In\_Play and Out\_of\_Play with temporal relations in soccer, volleyball and tennis video

shown in figure 5. The presentation of BSU converts the problem of video summarization for sports video into a pattern classification problem.

To generate the first-level sound-centric summarization which will attempt to parse the structure of the soundtrack in sports video, we classify and segment Audio BSU.



**Fig. 4.** Typical sequences of scoring event in soccer video



**Fig. 5.** Multi-level Summarizations based on all kinds of BSU for Sports video

To generate the second-level scene-driven summarization which will retain the affects of recording and editing by the director, we analyze the temporal relations among View BSU and Scene BSU.

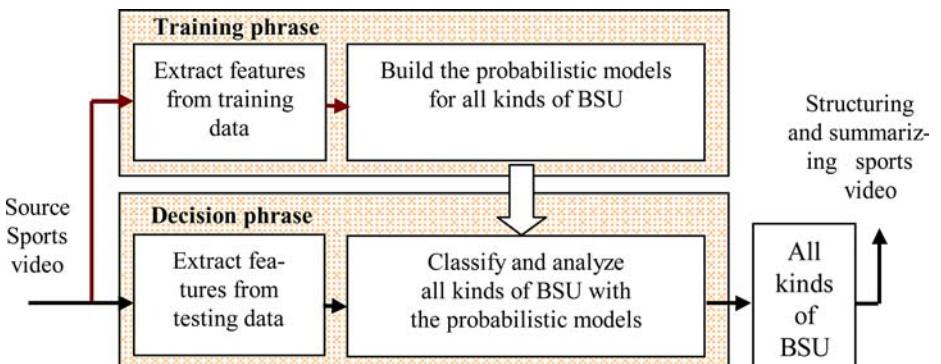
To generate the third-level event-driven summarization which will contain all the events important in sports video, for examples a skim of a soccer game would contain all the goals, we detect Event BSU fusing multi-features.

To generate the fourth-level highlight based on object of sports video which will abstract semantics in sports programs by detecting objects, such as caption, and so on, we segment and track Object BSU.

Since BSU represent the basic semantic units of sports video, the representative frames of all kinds of BSU will form the storyboards while the concatenation of BSU will form the skims. Next, we discuss the implementation of multi-level summarizations based on BSU in details.

### 3 Analysis of BSU for Sports Video Summarizations

The analysis and processing of all kinds of BSU is the foundation of automatic multi-level summarizations generation for sports video. We prefer the stochastic model rather than fine-tuning method. With this in mind, the probabilistic framework for mapping low-level feature to high-level semantic units is proposed, as shown in figure 6. In training phrase, features are extracted from training data and the probabilistic models for BSU are built. Then, in decision phrase, testing data are analyzed and classified with the above probabilistic models.



**Fig. 6.** Using the probabilistic models for mapping low-level features to high-level BSU

#### 3.1 The 1st-Level Sound-Centric Summarization — The Classification and Segmentation of Audio BSU

On the basis of figure 6, we use the Gaussian Mixture Model [7] for automatic audio classification and segmentation for sports video which is processed in two steps. In training phrase, clip-based audio features, mainly MFCC, are extracted from training data and three GMMs for three kinds of Audio BSU are built using Maximum Likelihood Estimate. Then, in decision phrase, the same audio features are got from testing data and the audio sequence is classified with the highest post-probability.

The result is evaluated as *classification accuracy*, defined as the number of correctly classified clips over total number of clips. And the *overall average classification*

accuracy over the entire dataset reaches 88%. It indicates that the algorithm performance is satisfactory.

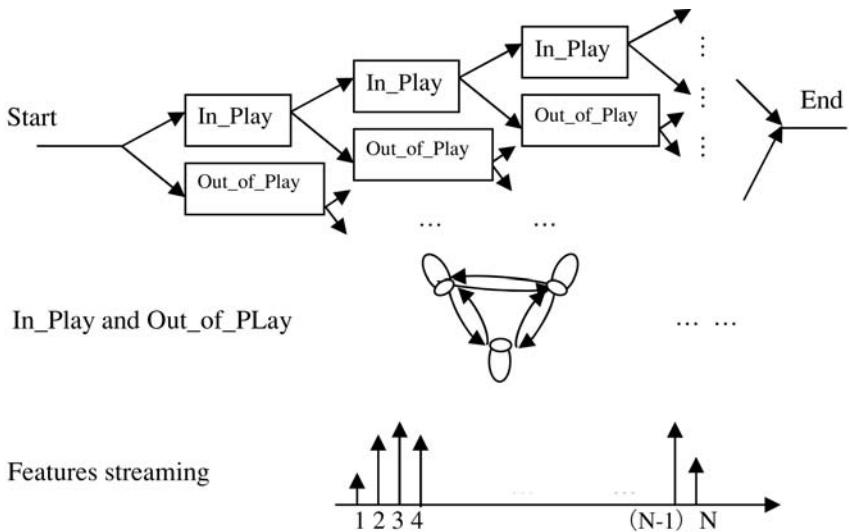
So, the representative frames of Audio BSU will form the 1st-level storyboards while the concatenation of Audio BSU will form 1st-level skims.

### 3.2 The 2nd-Level Scene-Driven Summarization — The Temporal Relations Analysis Among Scene BSU and View BSU [8]

Sports game has distinct In\_Play and Out\_of\_Play states, and besides, Scene BSU consists of different sub-structures with the switching of View BSU as shown in figure 7. This is so analogous to speech recognition that we use Hidden Markov Model to analyze the statistical temporal relations of View BSU and Scene BSU. In training phrase, two HMMs for Scene BSU are built using Baum-Welch algorithm. And the transition likelihoods between In\_Play and Out\_of\_Play are also calculated. Then, in decision phrase, the feature vector likelihood under each HMM is firstly evaluated and the optimal path is found with dynamic programming.

The result is evaluated as *classification accuracy* too and the *overall average classification accuracy* over the entire dataset is 85%. It is promising that high-level recording and editing rules for sports video can be discussed and computed with generic statistical models.

So, the representative frames of Scene BSU will form the 2nd-level storyboards while the concatenation of Scene BSU will form 2nd-level skims.

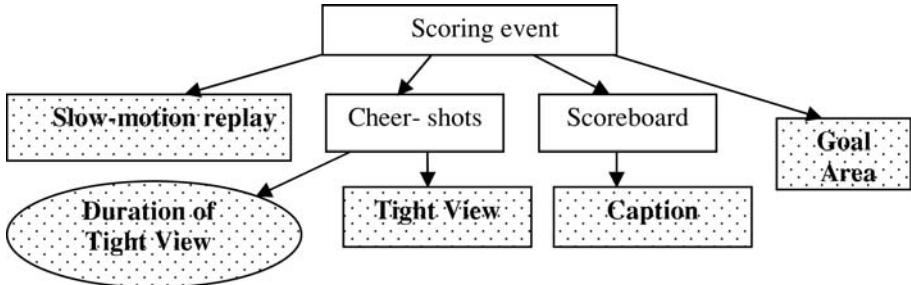


**Fig. 7.** The temporal relations among Scene BSU and View BSU

### 3.3 The 3rd-Level Event-Driven Summarization — The Detection of Event BSU Fusing Multi-features[9]

As we mention before, scoring event is indexed by some low-level cues such as appearance of goal area, updating of scoreboard, slow-motion replays and tight view.

On the basis of figure 4, we use Bayesian network to model scoring event detection in soccer video by integrating multi-features, as shown in figure 8. We follow the standard convention that shaded nodes are observed while clear ones are hidden. We also use the non-standard convention that square nodes are discrete and binary while round one, “*Duration of Tight View*”, is Gaussian continuous distribution.



**Fig. 8.** Bayesian network for scoring event detection in soccer video

In training phrase, we determine the probability distributions of nodes with Maximum Likelihood Estimate. Then, in decision phrase, the first step is getting the evidences which are the input of the next step. And the second step is computing the probability of the node “*Scoring event*”. If the probability of the node being true is larger than being false, we think there is a scoring event in the testing data. We make use of the classical inference method — the junction tree algorithm.

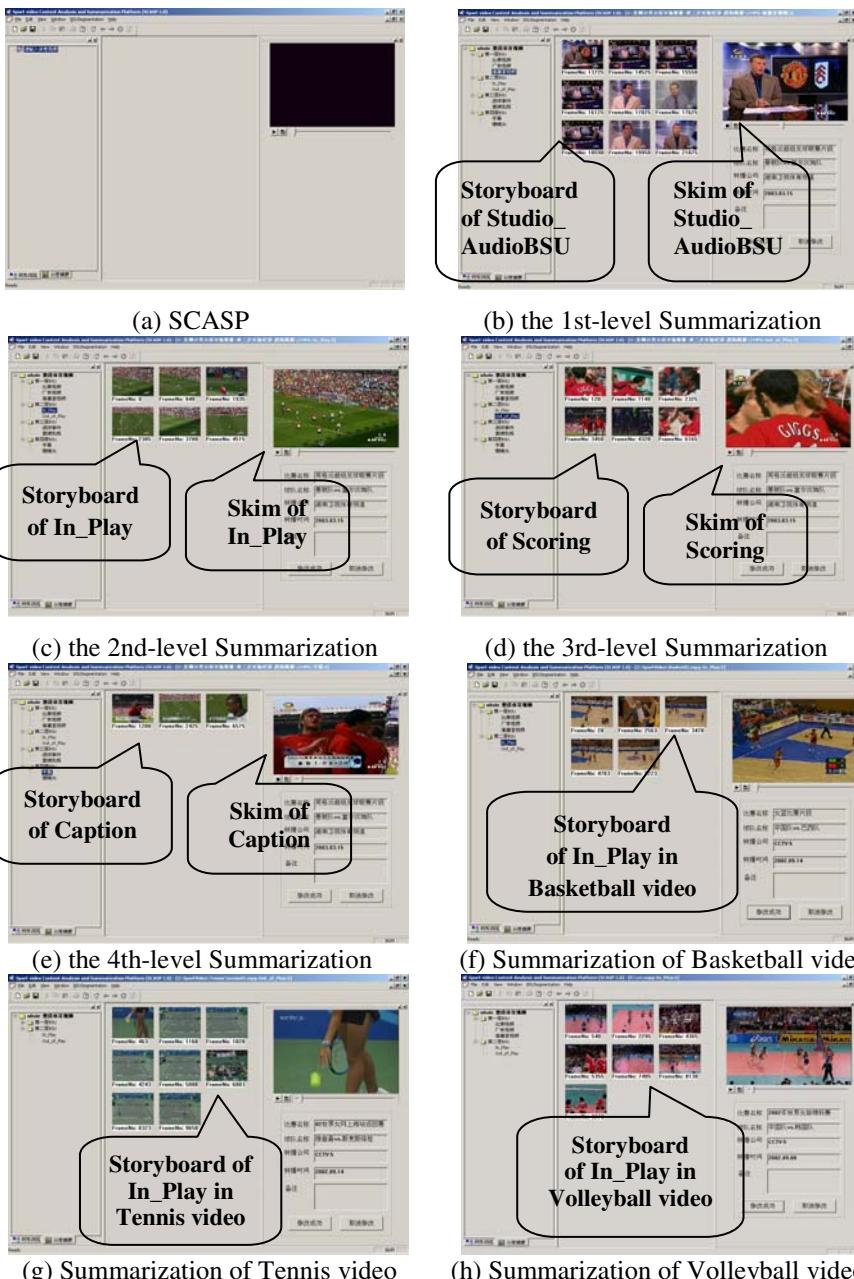
The result is evaluated as *Precision* and *Recall*. And the *overall average Precision and Recall* over the entire dataset is 73.3% and 87.8%. There are two main advantages for abstracting events using Bayesian network: one is that it can model the rules and fuse multi-features, the other is that the method can be extended to other Event BSU in sports video.

So, the representative frames of Event BSU will form the 3rd-level storyboards while the concatenation of Event BSU will form 3rd-level skims.

Specially, the details of the algorithms for low-level evidences which are just the analysis of Object BSU, for example the detection of caption and goal area, can be found in our previous paper [19]. Accordingly, the representative frames of Object BSU will form the 4th-level storyboards while the concatenation of Object BSU will form 4th-level skims.

## 4 The Design and Realization of SCASP

On the framework of multi-level summarizations based on BSU for sports video in figure 5 and the probabilistic models in figure 6, we design and realize the Sports video Semantic Content Analysis and Summarization Platform—SCASP, as shown in figure 9. Subfigure (b) (c) (d) (e) show the four-level summarizations for soccer video respectively corresponding to Audio BSU, Scene BSU, Event BSU and Object BSU. And the ideas of multi-level summarizations based on BSU are also applied to



**Fig. 9.** Sports video Semantic Content Analysis and Summarization Platform -- SCASP

other sports video. Subfigure (f) (g) (h) give the summarizations of basketball, tennis and volleyball video. The results of the preliminary experiments indicate that our approaches are generic, robust and effective.

## 5 Conclusions and Future Works

In this paper, we have described a novel framework for automatic multi-level sports video summarizations. On the basis of the characteristics of sports video, the most recurrent important semantic parts of sports video are modeled by “BSU”. And we classify BSU into Audio BSU, View BSU, Scene BSU, Event BSU and Object BSU, which are respectively in accord with multi-level summarizations. Moreover, we discuss the classification and analysis of all kinds of BSU using the probabilistic models in details. We have demonstrated the framework on soccer, basketball, tennis and volleyball video through SCASP. And the experimental results have shown that the framework provides a general approach to summarizing sports video.

Besides refining the technique we describe in this article, the future works include introducing new semantic elements for a BSU layer, incorporating more advanced statistical tools in the analysis of BSU, and formalizing our general BSU-based framework for sports video summarization.

## Acknowledgement

This research was supported by the National Science Foundation of China - 60572137.

## References

1. David A. Sadlier, Sean Marlow, Noel O' Connor and Noel Murphy. MPEG Audio Bit-stream Processing Towards the Automatic Generation of Sports Programme Summaries. In Proc/ of IEEE International Conference on Multimedia and Expo (ICME'02), Laussane, Switzerland, (2002) 26-29
2. F. Coldefy, P. Bouthemy. Unsupervised Soccer Video Abstraction Based on Pitch, Dominant Color and Camera Motion Analysis. In Proc. of the 12th ACM International Conference on Multimedia, New York, NY, USA, (2004) 268-271
3. Ahmet Ekin, A. Murat Tekalp. Automatic Soccer Video Analysis and Summarization. In Proc. of SPIE: Storage and Retrieval for Media Databases 2003, SPIE Volume 5021, Santa Clara, CA, USA, (2003) 339-350
4. Jinjun Wang, Changsheng Xu, Engsiong Chng, et al. Automatic Replay Generation for Soccer Video Broadcasting. In Proc. of the 12th ACM International Conference on Multimedia, New York, NY, USA, (2004) 32-39
5. Hao Pan, P. van Beek, M. I. Sezan. Detection of Slow-motion Replay Segments in Sports Video for Highlights Generation. In Proc. of IEEE International Conference on Acoustic, Speech and Signal Processing (ICASSP'01), Salt Lake City, UT, USA, (2001)
6. The Official Website of the Olympic Movement. [http://www.olympic.org/uk/games/index\\_uk.asp](http://www.olympic.org/uk/games/index_uk.asp)
7. Zhao Li. Speech Signal Processing, Publishing House of Mechanical Industry, Beijing, PRC, (2003).
8. L. X. Xie, Peng Xu, ShiF-Fu Chang. Structure Analysis of Soccer Video with Domain Knowledge and Hidden Markov Models. Pattern Recognition Letters, Vol. 25, Issue 7, (2004) 767-775
9. Chen Jianyun, Wu Lingda, et al. Semantic Event Detection in Soccer Video by Integrating Multi-features Using Bayesian Network. In Proc. of the International Symposium on Intelligent Multimedia, Video and Speech Processing (ISIMP2004), HongKong, (2004) 262-265

# Query-Topic Focused Web Pages Summarization

Seung Yeol Yoo and Achim Hoffmann

School of Computer Science and Engineering  
University of New South Wales Sydney 2052, NSW, Australia  
`{syy, achim}@cse.unsw.edu.au`

**Abstract.** We present a novel Web Pages Summarizer **ContextSummarizer** that groups the given Web pages into ‘sense-clusters’ respecting a user’s topical interests. ContextSummarizer constructs then an extractive summary for each sense-cluster. A user’s topical interest is described by the user who selects and refines some of the word senses disambiguated within the content contexts of the given Web pages. The semantic similarity measures between the contents of Web pages/segments/sentences and the user-selected word senses were used to choose the most topically relevant sentences as the extractive summaries referring to a user’s topical interest. ContextSummarizer addresses the semantic-alignment problem between the content of a Web page, the user’s topical interest, and the extractive summary of the Web page. Our case studies and experimental results showed that our query-topic focused extractive summaries returns more topically relevant sentences for an extractive summary than those produced by existing summarization systems.

## 1 Introduction

The continuous and rapid growth of Web information requires efficient assistance to Web users for information filtering and interpretation. Such efficient assistance is expected to be helpful in elucidating the occurred topics in retrieved Web pages. Further help can be provided by organizing the Web pages into topically associated collections (as done already by e.g. Web directories such as Yahoo!).

However, the topic-based elucidation and organization of multiple Web pages is difficult to be achieved without automated summarization techniques due to the labor-intensive and diverse nature of a user’s understanding of the contents of multiple Web pages. In this context, fully automatic summarization, having similar accuracy like the summary by a human who can extract the gist of Web information according to their information needs, is not yet possible. For instance, even though general purpose summaries of multiple Web pages can be extracted by a fully automatic method (e.g., [1]), such summaries are likely to contain irrelevant information with respect to user’s specific topic interests.

In this paper, we propose a novel Human Aided Machine Summarization (HAMS) system, called **ContextSummarizer**. ContextSummarizer goes beyond previous HAMS systems such as [2] where a user should highlight passages within a Web page. The user-highlighted passages were utilized as the source for the generation of topic-focused summaries. With such systems, a user has to

have not only a local view over each target Web page but a global view over a whole set of Web pages to properly choose the most topically interesting content segments. This tends to cognitively overload the user. In order to reduce a user's cognitive load our system iteratively suggests a set of classifiers (i.e., context-words and their senses) discovered from Web pages, and then asks the user to select only topically relevant classifiers to their interests (From now on, we interchangeably use a user's topic interest(s) or topic attention(s) over a collection of Web pages with '**query-topic(s)**').

The following is a central problem of generic summarization systems (e.g. Copernic Summarizer [3] and Pertinence Summarizer [4]): Without any topic focus for summarization, we cannot evaluate or compare the quality of a summary. To demonstrate the strength of ContextSummarizer, we also compared the query-topic focused 'extractive summaries' generated by our ContextSummarizer with those generated by Columbia Newsblaster [5]. The Columbia Newsblaster groups the given Web pages into a few predefined topic categories, and then generates an extractive summary on the Web pages clustered into each topic category. As an extrinsic evaluation result, we showed the terms extracted from our extractive summaries improve Web pages clustering.

This paper is organized as follows: In Section 2, we discuss related work. Section 3 briefly describes the system architecture composed with ContextExplicator, ContextSegmentor and ContextSummarizer. In Section 4, we introduce our summarization algorithm. Section 5 presents our initial evaluation results based on some case studies and an extrinsic evaluation. Finally, the conclusions are presented in Section 6.

## 2 Related Work

Recently a lot of work has been done on Web Page(s) summarization (e.g., [1,2,6,7,8,9,10,11,12]). In those works, Web pages are generally considered as having quite different characteristics in both structure and content, compared to pure-text documents. The work in [6,7] followed an abstractive summarization approach rather than an extractive summarization approach. Due to the difficulty of explicitly representing a query-topic and aligning the semantics between given Web pages and the query-topic, most extractive summarization systems focused on constructing generic summaries rather than topic-focused summaries.

Copernic Summarizer [3] uses statistical and linguistic algorithms for single-document summarization. This system pinpoints "key concepts" to select conceptually (i.e., topically) relevant sentences as a generic summary. However, the "key concept" is limited to work as a term (i.e., a term does not distinguish possibly different senses appearing in different sentences and documents) rather than a concept (i.e., a sense of the term, which can be distinguished from the other senses of the term) for summarization purposes. Newsblaster [5] has shown a good starting point by grouping articles together into predefined topic categories and summarizes the clustered Web pages in each topic category. (Newsblaster is based on two different summarizers: One looks for an informative summary

that is common information conveyed across all the given articles. The other produces a coherent (topic-specific) summary of the articles grouped into the same category.)

## 2.1 Our Approach for Web Pages Summarization

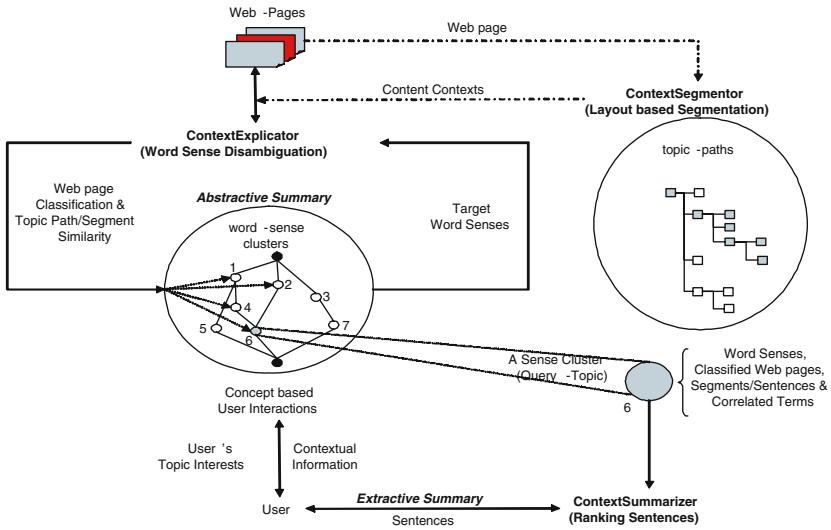
Future Web pages summarization systems might support both abstractive summaries and extractive summaries. An abstractive summary would provide structured meta-data (i.e., a collection of topic-index that describes itself and points out relevant Web pages) that make it possible to navigate the mass of information, within the given Web pages, at a higher abstraction level. An extractive summary would show the most salient sentences within topically relevant Web pages, which could be used as the coherent information source for many applications. ContextSummarizer aims to generate an abstractive summary and extractive summaries, which are seamlessly connected with each other.

ContextSummarizer can be differentiated in two ways from Newsblaster: Firstly, for topic-focused summarization, ContextSummarizer can generate extractive summaries for various query-topics (they are selected or refined by a user according to their topical interests), but Newsblaster is limited to generate extractive summaries for a few predefined topics. Secondly, ContextSummarizer provides users with a way that can simultaneously adjust the importance weights between topic-specificity and topic-informativeness of a summary, but Newsblaster generates two different summaries that separately consider only one aspect of ‘topic-specificity’ or ‘topic-informativeness’.

Another advanced feature of our ContextSummarizer is the conceptual representation of query-topics. A concept is utilized as the basic element for the semantic communication between the contents of Web pages and a users’ topic interests about those contents.

## 3 System Architecture

ContextSummarizer cooperates with two modules (See Figure 1) : **ContextSegmentor** [13] and **ContextExplicator** [14]. Firstly, ContextSegmentor performs the layout-based segmentation algorithm on each Web page to discover ‘topic-paths’ of the respective Web page. Secondly, ContextExplicator performs the word-sense disambiguation algorithm on each topic-path of a Web page to calculate its similarity with each user-selected word sense. Thirdly, according to a specific threshold value of the similarity measure, Web pages are classified into several word senses. The co-occurred word senses within a set of Web pages imply a sense cluster that is contextualized by several features. The associations among such sense clusters are represented by a conceptual lattice. In this paper, we call the conceptual lattice the “abstractive summary” of the given Web pages. Finally, **ContextSummarizer** executes a Web pages Summarization Algorithm on a user-selected sense-cluster (e.g., node 6 in Figure 1) to mine the most important sentences (see section 4 for more details) that are called “extractive



**Fig. 1.** System Architecture: ContextSummarizer cooperates with ContextSegmentor [13] and ContextExpiator [14] to discover query-topic focused “abstractive summary” and “extractive summary” from the given Web pages

summary” in this paper. (See [14] for more details about the construction of an abstractive summary).

## 4 Summarization Algorithm

The importance of a semantic alignment problem between a user’s query-topic and an extractive summary was discussed in Section 2. To address this semantic alignment problem, the specificity value referring to a user-selected query-topic and the informativeness value reflecting the whole contents of relevant Web pages are considered by ContextSummarizer.

The notation used to explain our summarization algorithm follows:

$q_t$ : a query term;

$W$ : the originally given Web pages  $\{w_i|i > 0\}$ ;

$WS$ : all known word-senses  $\{\sum_t WS(q_t)|WS(q_t)$  returns the known word-senses of  $q_t\}$  for all query terms  $\{q_t|t > 0\}$ ;

$sc_c$ : a sense cluster contextualized with a set of word-senses  $WS_c$  ( $\subset WS$ ) and a set of Web pages  $W_c$  ( $\subset W$ ) including contents which are sense-matched with all word-senses of  $WS_c$ ;

$qt_k$ : a sense cluster  $sc_c$  (where  $k=c$ ) that is selected by a user as a query-topic. The Web pages  $W_k$  are used as the source for the generation of an extractive summary;

$CE_{k**}^*$ : a content-element related to a query-topic  $qt_k$ . The subsumption relationship among content-elements, having different granularities, can

be illustrated as a sentence  $CE_{klmno}^{st} < \text{a segment } CE_{klmn}^{seg} < \text{a topic-path } CE_{klm}^{tp} < \text{a Web page } CE_{kl}^w$ ;

$CT(ws_{kq})$ : the context-words of a word sense  $ws_{kq} \in WS_k$ ;

$CTentW(CE_{k**}^*)$ : representative content-words occurred within a content boundary of  $CE_{k**}^*$ ;

## 4.1 The Specificity Values of Context-Elements

$$S(CE_{k**}^*, qt_k) = \sum_q R(CT(ws_{kq}), CE_{k**}^*) \quad (1)$$

Equation 1 measures the specificity value, referring to the user-selected query-topic  $qt_k$ , of a context-element  $CE_{k**}^*$ . As the semantic of the query-topic  $qt_k$  can be indicated by the relevant word-senses  $WS_k$ , it calculates the sum of similarities between each word-sense  $ws_{kq} \in WS_k$  and the content-element  $CE_{k**}^*$ .  $R(CT(ws_{kq}), CE_{k**}^*)$  is calculated by the sum of  $TF * ICEF$  (it is applied from the TF\*IDF in [15]) for all context-word  $ct \in CT(ws_{kq})$ ; the number  $TF$  of occurrences of a context-word  $ct \in CT(ws_{kq})$  within a content-element  $CE_{k**}^*$  is multiplied by the inverted content-element frequency  $ICEF$ , at the same granularity with the content-element  $CE_{k**}^*$ , of the context-word.

All context-elements belonging to the same granularity with  $CE_{k**}^*$  were ranked according to their specificity values referring to the user-selected query-topic  $qt_k$ , with one constraint: when a content-element has non-zero specificity values for more word-senses, the content-element was higher-ranked than other content-elements having non-zero specificity value for less word-senses. The  $T$  top-ranked content-elements became the targeted content-elements to discover their informativeness values.  $T$  depends on the user-chosen compression rate.

## 4.2 The Informativeness Values of Context-Elements

The  $T$  content-elements, selected according to the ranks of specificity values, might not always be matched by the  $T$  top-ranked informative (i.e., representative) content-elements of the Web pages  $W_k$ . Two possible reasons are:

1. During an iterative specialization procedure of a word-sense  $ws_{kq}$  relevant to a query-topic  $qt_k$ , ContextExplicator attempted to recommend more representative context-words that can distinguish relevant content-elements to the word-sense. However, our observation says that users preferred locally distinctive context-words to globally distinctive context-words.
2. ContextExplicator allows a user to construct a new conceptual-model by conjunctively combining predefined word-senses. The problem is that the context-words defining a word-sense were usually recommended according to their occurrence-frequencies within different source data (e.g., different Web pages, topic-paths, segments or sentences) at different time. Thus, it is necessary to reconsider their informativeness of the contents of new a data source, when users want to reuse predefined word-senses and their context-words as features.

Hence, ContextSummarizer needs not only to consider the specificity values (i.e., *distinction ranks*; they reflect the relevance value of each content-element to a query-topic  $qt_k$ ) but also the informativeness values (i.e., *coverage ranks*; they reflect the coverage value of each content-element to all content-elements at the same granularity) of content-elements. ContextSummarizer measures the informativeness of a content-element with the following equation:

$$I(CE_{kp}^*) = \frac{WF(CTentW_{kp}, CE_{kp}^*)}{\sum_s WF(CTentW_{ks}, CE_{ks}^*)} \quad (2)$$

where  $CE_{kp}^*$  is a content-element, and  $\{CE_{ks}^*|s\}$  are all content-elements (including  $CE_{kp}^*$ ) having the same granularity as  $CE_{kp}^*$ .  $I(CE_{kp}^*)$  has value-range of [0, 1]. We define the representative content-words  $CTentW_{k**}$  (here  $k** \equiv ks$  or  $kp$ ) for the content-element  $CE_{k**}^*$  as a set of unique content-words (i.e., noun or noun phrases) having equal or more occurrence-frequency ( $\neq 0$ ) than any context-word  $cw_g \in CT(ws_{kq})$  within  $CE_{k**}^*$ . Function  $WF()$  returns the occurrence frequency of the content-words  $CTentW_{k**}$  within the content of  $CE_{k**}^*$ . The informativeness (i.e., representativeness) of a content-element  $CE_{kp}^*$  is indicated by the normalized frequency within  $CTentW_{kp}$  of the content-element  $CE_{kp}^*$  over the sum frequency of  $CTentW_{ks}$  within all content-elements  $\{CE_{ks}^*|s\}$ .

### 4.3 Combining the Specificity and Informativeness

ContextSummarizer uses the following equation to combine the specificity value and the informativeness value of a content-element  $CE_{kp}^*$ :

$$SI(CE_{kp}^*) = \alpha * \frac{S(CE_{kp}^*, qt_k)}{M_s} + \beta * I(CE_{kp}^*) \quad (3)$$

where  $M_s = Max(S(CE_{k**}^*, qt_k))$  used to convert the value-range of  $S(CE_{kp}^*, qt_k)$  into [0,1]. If  $M_s = 0$  then it sets  $M_s = 1$ .  $0 \leq \alpha \leq 1$  and  $\beta = 1 - \alpha$ .  $\alpha$  and  $\beta$  is respectively the relative weight of specificity and informativeness.  $\alpha$  and  $\beta$  are adjustable according to a user's needs. The default value of  $\alpha$  is 0.5. When  $\alpha > \beta$ , our system returns a more query-topic focused summary rather than a generic summary. The content-elements are ranked by their  $SI(CE_{kp}^*)$  values.

### 4.4 Sentences for a Web Pages Summary

To extract sentences to compose a summary, ContextSummarizer iteratively ranks the content-elements at different granularity from topic-path to sentence. Finally, ContextSummarizer applied the following four constraints (user-adjustable at summarization run time) to select sentences for a summary.

1. Compression Rate: the total number of selected sentences for a summary is controlled by the requested percentages of top-ranked sentences in relation to the total number of sentences of the target Web pages.

2. Length: it removed sentences that do not contain a sufficient number of content-words.
3. Redundancy: if a sentence does not contain a sufficient number of unique content-words against each higher ranked sentence, the sentence is removed.
4. Abstraction Level: The abstraction level of an extractive summary is indirectly adjusted according to the user-selected sense-cluster. For example, the summary for a sense-cluster (i.e., a topic) would be topically more specific compared to those of its ancestor sense-clusters (i.e., super-topics), and would be topically more general description compared to those of its descendant sense-clusters (i.e., sub-topics).

The finally selected and ranked sentences are grouped and presented to users according to the original orders in the target Web pages.

## 5 Evaluation

In [16], existing summarization evaluation techniques are subdivided into two categories: an intrinsic approach and an extrinsic approach. The intrinsic approach independently evaluates a summary from any application purposes. One way of the intrinsic approach is to compare a summary generated by a summarization system with a human-produced “gold” summary. However, this approach has the problem of achieving agreement on what constitutes a “gold” summary [17]. This problem is exacerbated by the fact that different summaries might be judged “good” summaries depending on the current user’s topic interests. Thus, in this evaluation, we substituted showing a few comparable summarization results for an intrinsic approach. (We entrust the quality comparison task over those summarization results to each reader of this paper). The extrinsic approach compares the performances of an application that utilizes the system-generated summaries for its specific task. In this evaluation, we showed how the summaries extracted by ContextSummarizer can be utilized to improve clustering performances.

### 5.1 Case Studies

A major deficiency of general summarization systems, antithetical to topic-focused summarization systems, is that their summaries are significantly different at the same compression rate even over the same document(s). For example, in the “2. A major deficiency of general summarization systems’ of [18], Copernic Summarizer and Pertinence Summarizer showed noteworthy differences in their summaries over the same Web page “TWP1”. However, without any topic-focus respecting a user’s specific information need, it is not possible to judge which system generated better quality summaries.

Columbia Newsblaster [5] makes multi-document summaries on topic clusters. The topic clusters are based on six predefined categories (i.e., main topics) having hierarchical sub-topic levels. For example, in the “3.1 Multi-WebPages

Summarization by Columbia Newsblaster [18]”, Columbia Newsblaster generated a summary that is talking about “festival”(subtopic) under the category of “entertainment”(topic). The Web pages clustered into the same topic cluster are assumed to provide topically coherent contents to generate a multi-document summary. This may only be achievable using predefined topics. Unfortunately, no predefined topic categories can cover all potential topical interests of a user. For example, when a user requests a summary about “ticketing of the festival”, Columbia Newsblaster cannot provide an answer.

ContextSummarizer resolved those problems, mentioned above, by constructing topic and subtopic categories at summarization run time by considering not only content contexts of the target Web pages but also a query-topic. For example, let us assume that a user is looking for news about “ticketing of the festival”, and another user wants to get news about “music events during the festival”. The summaries, generated by ContextSummarizer that respect the users’ different interests, are displayed in the “a) and b) of 3.2 Multi-WebPages Summarization by ContextSummarizer [18]”.

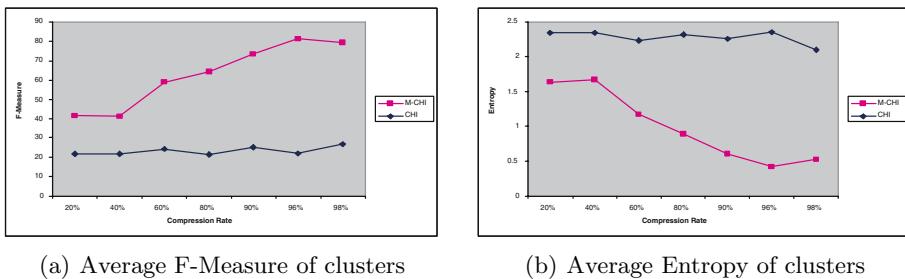
## 5.2 Comparisons of Clustering Performances

In this subsection, we consider that the content-words occurred within a better summary can be used as more accurate features to properly subgroup the given test Web pages into topically relevant clusters.

**Experimental Setup:** The user-selections for the context-words of a query-topic might be diverse for different users. In our experiments, we eliminated such user-dependent deviations to allow more comparable evaluation of an extractive summary, by assuming that users’ common topic interests (i.e., query-topics) are the main topics occurred within the given test Web pages. For the test Web pages, we collected 150 documents (i.e., articles from 02/04/2006 to 02/07/2006) about 19 news topics in Columbia Newsblasters.

**Evaluation Measure:** Entropy and F-measure were used to evaluate the clustering performance. Those measures of a cluster respectively indicates the uniformity and the weighted harmonic mean of precision and recall of assigned Web pages to the cluster. The Entropy of all obtained clusters is defined by the weighted sum of the entropy of each obtained cluster, as following:  $Entropy = \sum_{k=1}^{C'} \frac{|W_k|}{N} \sum_{j=1}^C p_{jk} \times \log(p_{jk})$ , where  $p_{jk} = \frac{1}{|W_k|} |\{w_i | label(w_i) = c_j\}|$  is the entropy of a obtained cluster.  $N$  is the total number of Web pages in the tested original clusters.  $C'$ ,  $C$  denote the number of obtained clusters and the number of original clusters respectively.  $W_k$  is the set of Web pages in an obtained cluster.  $label(w_i)$  returns the cluster-label of the Web page  $w_i$ . The cluster-label of a generated cluster is decided by the original cluster-label of the most shared Web pages in the generated cluster. The F-measure of all obtained clusters is defined by the average sum of the F-measure of each obtained cluster. The traditional F-measure (i.e., F1 measure; recall and precision are evenly weighted) of a obtained cluster is defined as  $F(W) = \frac{2 \times Precision(W) \times Recall(W)}{Precision(W) + Recall(W)}$ .

**Evaluation Results:** We compared the quality of clusters that are constructed by utilizing a set of content-words extracted with a supervised feature selection method,  $\chi^2$  statistic [19]<sup>1</sup> from two different content sources: One content source is a set of ranked content-words  $S_1$  occurred within the given test Web pages. Another content source is a set of ranked sentences  $S_2$  for specific topic(s), by our ContextSummarizer. A compression rate is applied to two different sources  $S_1$  and  $S_2$  for the selection of the final content-words that are used as the features for the clustering of the given test Web pages. For example, to apply a compression rate of 5% to content source  $S_1$ , it selects the 5% top-ranked content-words  $C_1$  from  $S_1$ . To apply a compression rate of 5% to content source  $S_2$ , it selects top-ranked sentences  $S_2$  that contains the same number of content-words with  $C_1$ , and then calculates the weights of the content-words  $C_2$  occurred within  $S_2$ .  $C_1$  and  $C_2$  are respectively used to construct two different groups of clusters and the results for each group of clusters are compared.



**Fig. 2.** CHI and M-CHI methods utilized the content-words, extracted from the test Web pages and the summaries generated by ContextSummarizer, to construct clusters

Figure 2 shows the average F-measure and Entropy of generated clusters, at each compression rate 20, 40, 60, 80, 90, 96 and 98%. Through all compression rates, the clustering performances based on the M-CHI method, using the content-words  $C_2$  extracted from the sentences summarized by ContextSummarizer, were much better than those based on the CHI method, using the content-words  $C_1$  extracted from the test Web pages. The trend of F-measure and Entropy measure of M-CHI was getting better when the compression rate became higher (up to 96%). It can be understood as the evidence of two facts: 1) ContextSummarizer returned “good” summarized sentences that are including topically coherent information to properly cluster relevant Web pages into the tested 19 topics, and 2) A reasonable summarization performance of ContextSummarizer was kept at high compression rates. It means that much fewer (i.e., human readable number of) sentences can be presented as an extractive summary to the user instead of very large Web pages.

<sup>1</sup>  $\chi^2$  statistic is a supervised feature selection method which measures the statistical significance of association between a term and a category using known class labels.

## 6 Conclusion

ContextSummarizer adapts its summarization results according to: a) a requested query-topic (i.e., contextual features describing the user's attention to partial contents of the given Web pages). The user can easily move their topical interests in one of the sense-clusters composing a concept-model, b) the semantic-aligned contents (i.e., Web pages, topic-paths, segments and sentences) to the query-topic, c) the redundancy removal from the topically matched and top-ranked sentences. It provides a way to compact the summary sentences extracted from large scaled Web pages, and d) the requested compression rate of an extractive summary.

Some case studies and experimental results showed that a query-topic focused extractive summary can be composed of more topically consistent sentences rather than a generic summary.

## References

1. Shen, D., Chen, Z., Yang, Q., Zeng, H.J., Zhang, B., Lu, Y., Ma, W.Y.: Web-page classification through summarization. In: Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2004, ACM (2004) 242–249
2. Amitay, E., Paris, C.: Automatically summarising web sites - is there a way around it? In: Proceedings of 9th International Conference on Information and Knowledge Management, CIKM 2000, ACM (2000) 173–179
3. (CopernicSummarizer: <http://www.copernic.com//en/products/summarizer/>)
4. (PertinenceSummarizer: [http://www.pertinence.net/index\\_en.html](http://www.pertinence.net/index_en.html))
5. (Columbia Newsblaster: <http://www1.cs.columbia.edu/nlp/newsblaster/>)
6. Alam, H., Hartono, R., Kumar, A., Rahman, F., Tarnikova, Y., Wilcox, C.: Web page summarization for handheld devices: A natural language approach. In: Proceedings of the Seventh International Conference on Document Analysis and Recognition Volume II, ACM (2003) 1153–1157
7. Berger, A.L., Mittal, V.O.: *ocelot*: a system for summarizing web pages. In: Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2000, ACM (2000) 144–151
8. Buyukkokten, O., Garcia-Molina, H., Paepcke, A.: Seeing the whole in parts: text summarization for web browsing on handheld devices. In: Proceedings of the Tenth international conference on World Wide Web, WWW 2001, ACM (2001) 652–662
9. Delort, J.Y., Bouchon-Meunier, B., Rifqi, M.: Links for a better web: Enhanced web document summarization using hyperlinks. In: Proceedings of the fourteenth ACM conference on Hypertext and hypermedia, ACM (2003) 208–215
10. Delort, J.Y., Bouchon-Meunier, B., Rifqi, M.: Web document summarization by context. In: Proceedings of the 12th international conference on World Wide Web - Alternate Track Papers & Posters, WWW 2003, ACM (2003)
11. Jatowt, A.: Web page summarization using dynamic content. In: Proceedings of the 13th international conference on World Wide Web - Alternate Track Papers & Posters, WWW 2004, ACM (2004) 344–345
12. Jatowt, A., Ishizuka, M.: Temporal web page summarization. In: Proceedings of 5th International Conference on Web Information Systems Engineering, WISE 2004, Springer (2004) 303–312

13. Yoo, S.Y., Hoffmann, A.: Pseudo-relevance feedback in web information retrieval using segments' subjective importance values. In: 2005 IEEE/WIC/ACM International Conference on Web Intelligence (WI 2005), IEEE Computer Society (2005) 422–425
14. Yoo, S.Y., Hoffmann, A.: Knowledge-level management of web information. In: 6th Asia-Pacific Web Conference, APWeb 2005, Springer (2005) 597–608
15. Salton, G., Buckley, C.: Term-weighting approaches in automatic retrieval. Information Processing and Management **24** (1988) 513–523
16. Mani, I., Maybury, M.T.: Advances in Automatic Text Summarization. MIT Press (1999)
17. Stergos D. Afantinos, V.K., Stamatopoulos, P.: Summarization from medical documents: a survey. Artificial Intelligence in Medicine **33** (2005) 157–177
18. ContextSummarizer: <http://www.cse.unsw.edu.au/~syy/ContextSummarizer-casestudy.html>. (2005)
19. Luigi Galavotti, F.S., Simi, M.: Experiments on the feature selection and negative evidence in automated text categorization. In: Research and Advanced Technology for Digital Libraries, 4th European Conference (ECDL 2000). (2000) 59–68

# Invariant Color Model-Based Shadow Removal in Traffic Image and a New Metric for Evaluating the Performance of Shadow Removal Methods

Young Sung Soh, Hwanju Lee, and Yakun Wang

Dept. of Information Engineering, Myongji Univ., Namdong, Yongin, Kyunggido, Korea  
{soh, niceju77, wykluck}@mj.ac.kr

**Abstract.** To track objects in a traffic image sequence, objects must be extracted first. Background differencing is frequently used to extract objects. When objects are extracted, it is quite possible that shadows are included. With shadow it is not easy to do precise tracking. Thus shadows need to be removed. To do this, we proposed invariant color-based shadow removal method. Many shadow removal methods were proposed. To compare the quality of methods, several metrics were suggested. However, they suffer from inconsistency where qualitative and quantitative results do not coincide. In this paper, we proposed a new metric having such consistency.

## 1 Introduction

Background differencing is frequently used in tracking systems to extract objects. Shadows may be included in the objects and need to be removed for precise tracking. Thus we need a shadow removal method which can differentiate between real object and the shadow. Also we need an objective and consistent metric to evaluate the performance of various shadow removal methods. Many methods were proposed for shadow removal. Normalized RGB(NRGB)[1] uses the property that shadow preserves chromaticity in color images, but loses brightness. NRGB does not work well for road surface which usually does not have chromaticity component. Gradient Direction Correlation(GDC)[2] compares extracted object area with corresponding background, selects dark pixels as shadow candidate pixels, computes the correlation between gradient directions of each candidate and corresponding background pixel, and classifies as a shadow pixel if correlation is above some threshold. GDC assumes strong texture component in the image. Color Normalized Cross Correlation (CNCC)[3] uses HSI color model to estimate the color similarity between the object and corresponding background. CNCC does not perform well on gray tone road surface. Color Information and History Data(CIHD)[4] method classifies pixels in extracted foreground blob into vehicle pixel and shadow pixel based on RGB values, computes the ratio between the number of shadow pixels and the blob size, and labels the blob as either regular vehicle blob or dark vehicle blob. CIHD removes shadow pixels for regular blobs. For dark vehicle blobs, CIHD makes use of removal history data to cut off the shadow from blob. CIHD does not differentiate between self and cast shadows, and tends to remove self shadow. To overcome aforementioned difficulties, we proposed invariant color-based shadow removal method.

There are two representative methods in evaluating the performance of shadow removal methods. Cohen et al.[7] introduced two metrics : Detection Rate and False Alarm Rate. However they did not verify whether nonshadow area belongs to foreground or background. To improve on this, Prati et al.[8] defined Shadow Detection Rate and Shadow Discrimination Rate, but these do no not support the consistency between qualitative and quantitative evaluations. In this paper, we proposed a new objective and consistent metric.

In chapters 2 and 3, we show relevant works and proposed method respectively. Chapter 4 describes the experimental results and chapter 5 summarizes the conclusion and future research.

## 2 Relevant Works

### 2.1 Shadow Removal

Many methods were proposed for shadow removal. In this paper, we review four representative works. NRGB[1] uses the property that when background is covered by shadow, brightness changes much, but chromaticity is well preserved. They use single Gaussian for background modeling, compute normalized RGB for both foreground and corresponding background, and compare them if they are similar. NRGB yields poor results for gray tone background such as road surface. GDC[2] first selects dark pixels from foreground using a predefined threshold. Then it computes the gradient directions of each dark pixel and corresponding background pixel. If they agree, then it is a shadow pixel. Since GDC assumes strong texture component on the background, it fails when the assumption does not hold as in the case of road surface. CNCC[3] uses HSI color model to estimate the color similarity between foreground and background. To assess color similarity, they compute normalized cross correlation between colors of foreground and corresponding background. If they are similar, then it is a shadow. Since CNCC assumes strong chromaticity component on background, it is not suitable for gray tone background we will be working on. CIHD[4] uses mixture of Gaussians(MOG) with slight modification for background modeling. They check each pixel in foreground if it has enough chromaticity component and if it is not dark.. If these conditions are satisfied, the pixel is classified as a vehicle pixel, otherwise as a shadow pixel. They then compute the ratio between the number of vehicle pixels and foreground size. If it exceeds some threshold, it is a regular vehicle blob, otherwise it is a dark vehicle blob. For regular vehicle blob, pixels marked as shadow are removed. For dark vehicle blob, they use previous history of shadow removal to discard the shadow from the blob CIHD tends to remove self shadow because it cannot differentiate between self and cast shadows.

### 2.2 Evaluation Metric for Shadow Removal Methods

There have been efforts on evaluating various shadow removal methods. Cohen et al.[7] proposed two metrics DR(Detection Rate) and FAR(False Alarm Rate) shown in equation 1.

$$DR = \frac{TP}{TP + FN}, \quad FAR = \frac{FP}{TP + FP} \quad (\text{Eq. 1})$$

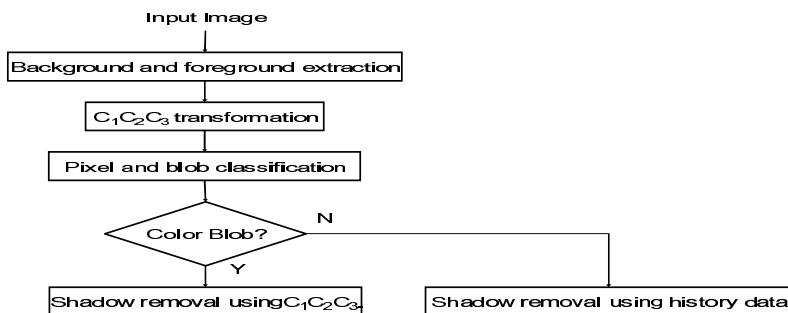
Here, TP(True Positive) is the number of shadow pixels classified as shadow pixels, FP(False Positive) the number of background or vehicle pixels classified as shadow pixels, and FN(False Negative) the number of shadow pixels classified as background or vehicle pixels. These metrics do not differentiate between background and vehicle pixels when they are misclassified as shadow pixels. Thus their metrics can measure how well shadows are detected, but cannot measure how well shadows are discriminated from background or vehicle pixels. To improve on this, Prati et al[8] proposed shadow detection rate( $\eta$ ) and shadow discrimination rate( $\xi$ ) shown in equation 2.

$$\eta = \frac{TP_S}{TP_S + FN_S}, \quad \xi = \frac{\overline{TP}_F}{TP_F + FN_F} \quad (\text{Eq. 2})$$

Here, subscripts S and F mean shadow and foreground respectively.  $\overline{TP}_F$  is the difference between the number of foreground ground truth pixels and the number of foreground pixels misclassified as shadow pixels.. These metrics suffer from following problem. The larger the  $\eta$  and the  $\xi$ , the better. If some method A has high  $\eta$  and low  $\xi$ , and method B has low  $\eta$  and high  $\xi$ , we cannot tell which method is better. Moreover, quantitative evaluation using these metrics does not usually correspond to qualitative evaluation done by human.

### 3 The Proposed Method

Fig. 1 shows the block diagram of the proposed shadow removal method. Detailed explanation of each block follows.



**Fig. 1.** Block diagram of shadow removal method

#### 3.1.1 Background and Foreground Extraction

We use modified MOG proposed by Choi[4]. They basically use MOG proposed by Stauffer et al.[5]. However [5] suffers from the problem that it is very sensitive to

minor variations. That is, as some distribution keeps getting values close to its mean, it becomes narrower and narrower, thus a value which is a little bit off the mean is considered to lie in the tail of the distribution, resulting in false classification between background and foreground. So they modify MOG update process such that its standard deviation does not fall under some predefined threshold.

### 3.1.2 $C_1 C_2 C_3$ Transformation

Input image is in RGB format. We obtain one more image by transforming RGB to  $C_1 C_2 C_3$  using equation 3.

$$C_1 = \arctan\left(\frac{R}{\max(G, B)}\right), \quad C_2 = \arctan\left(\frac{G}{\max(R, B)}\right), \quad C_3 = \arctan\left(\frac{B}{\max(R, G)}\right) \quad (\text{Eq. 3})$$

We use  $C_1 C_2 C_3$  since it is invariant to shadow and is proven to be the best invariant color model comparing to RGB, normalized RGB, HIS, YUV, and  $l_1 l_2 l_3$  in the study conducted by Salvador[6].

### 3.1.3 Pixel and Blob Classification

First we classify each pixel in a blob into one of BP(Bright Pixel), CP(Color Pixel), and NCP(Non Color Pixel). BP is the number of pixels whose average RGB values exceed predefined threshold which shadows can never have. Then we classify remaining pixels into CP and NCP by computing the difference among  $C_1$ ,  $C_2$ , and  $C_3$ . The smaller the difference, the closer to gray tone class which is NCP. Using this classification result, we determine the type of blob between CB(Color Blob) and NCB(Non Color Blob). To do this we compute CPR(Color Pixel Ratio) for each blob by  $CP/(CP+NCP)$ , and if CPR is above some threshold, then it is CB, otherwise it is NCB.

### 3.1.4 Shadow Removal Using History Data

For those blobs belonging to NCB, we use shadow removal method proposed by Choi.[4]. They classify pixels in each blob into shadow and vehicle pixels, and compute the ratio of vehicle pixels over all pixels in that blob. If it is above some threshold, it is a regular vehicle blob, otherwise it is a dark vehicle blob. For regular blob, they just remove shadow pixels, shrink the box enclosing the shadow-removed foreground and save the horizontal and vertical shrink ratio of enclosing box. For dark vehicle blob, they apply horizontal and vertical shrink ratio of the spatially and temporally closest regular blob to reinforce the shadow cutting.

### 3.1.5 Shadow Removal Using $C_1 C_2 C_3$

For those blobs belonging to CB, we use  $C_1 C_2 C_3$  to remove shadow. For each blob, we keep pixels with label BP and CP, and remove pixels with label NCP.

It is quite probable that pixels belonging to vehicle window are classified as NCP and are removed since they usually appear dark. Since the portion of the window in vehicle is not ignorable, we need a mechanism to detect window pixels and reclassify them as vehicle pixels, i.e., CP. Assuming vehicle moves from top to bottom direction in the image and using the observation that front window is in between vehicle roof

and hood, we scan CB(Color Blob) vertically to find such patterns. If multiple patterns are found in a single column, we choose the longest.

### 3.2 New Metric for Shadow Removal Method Evaluation

GT, TP, TN, FP, and FN mean Ground Truth, True Positive, True Negative, False Positive, and False Negative respectively as in section 2.2. Rewriting and expanding equation 2 for  $\eta$  and  $\xi$  proposed by Prati et al.[8], we get equation 4.

$$\eta = \frac{TP_S}{TP_S+FN_S} = \frac{TP_S}{GT_S}, \quad \xi = \frac{\overline{TP}_F}{TP_F+FN_F} = \frac{GT_F-FN_F}{GT_F} = \frac{TP_F}{GT_F}$$

$$GT = GT_F+GT_S, \quad GT_S = TP_S+FN_S, \quad GT_F = TP_F+FN_F \quad (\text{Eq. 4})$$

Here F corresponds to vehicle, S to shadow, GT to number of foreground pixels,  $GT_F$  to number of vehicle pixels in foreground, and  $GT_S$  to number of shadow pixels in foreground. The fundamental problem with  $\eta$  and  $\xi$  is that they are treated to have equal weights. Depending on the illumination direction, shadow size varies. For small shadow, a slight misclassification on shadow may affect  $\eta$  considerably and for large shadow, a slight misclassification on vehicle may affect  $\xi$  considerably. Moreover quantitative evaluation results from  $\eta$  and  $\xi$  do not coincide with qualitative evaluation results. To overcome this difficulty, we propose a new metric  $\gamma$  as in equation 5.

$$\gamma = \frac{GT_S}{GT} * \eta + \frac{GT_F}{GT} * \xi = \frac{TP_S+TP_F}{GT} \quad (\text{Eq. 5})$$

Proper weights are associated with  $\eta$  and  $\xi$ . Since  $\eta$  is related to shadow, we reflect  $\eta$  in  $\gamma$  by the portion of shadow in foreground. By the same logic, since  $\xi$  is related to vehicle, we reflect  $\xi$  in  $\gamma$  by the portion of vehicle in foreground. This metric provides consistency between quantitative and qualitative evaluations

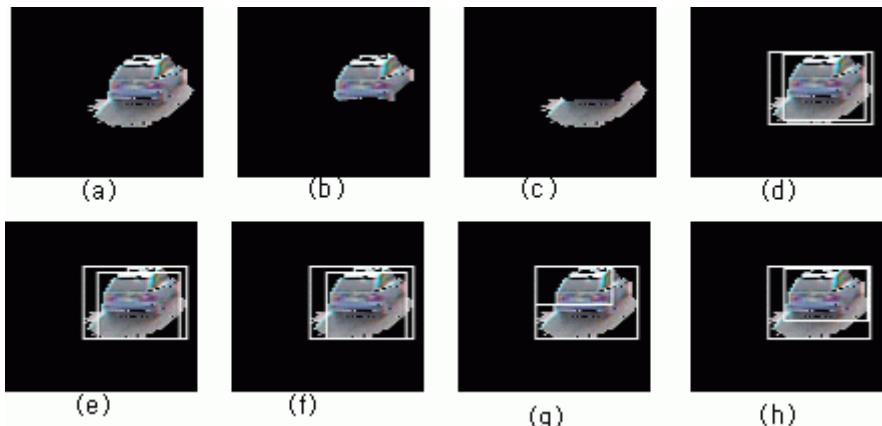
## 4 Experimental Results

The experiment was conducted using the image sequences captured on road near some stations of Seoul metropolitan subway. Images were in RGB and of size 320x240.

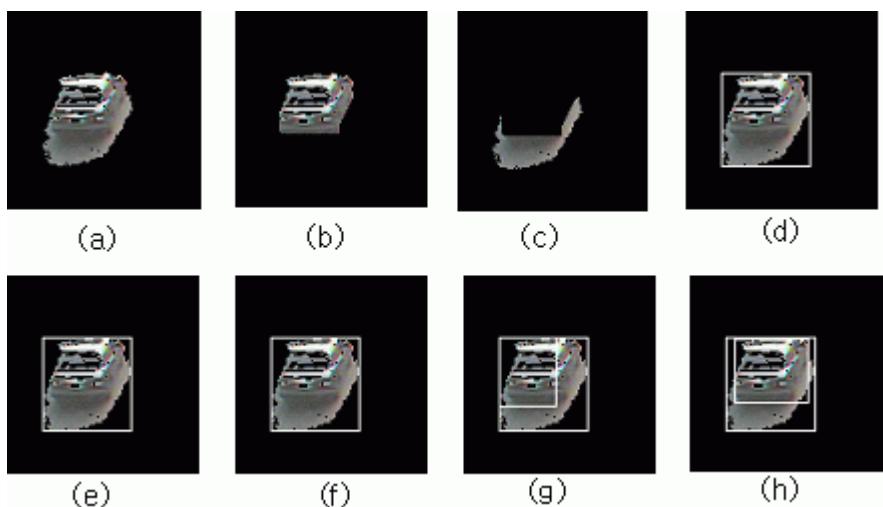
### 4.1 Shadow Removal

Figs. 2 and 3 show the performance of various shadow removal methods. Each figure has 8 images from (a) to (h) where (a) represents extracted foreground from the image, (b) the vehicle part of foreground, and (c) the shadow part of foreground respectively. Images (b) and (c) were obtained manually to use as a ground truth. Images (d), (e), (f), (g), and (h) were obtained by removing the shadow using NRGB,

GDC, CNCC, CIHD, and the proposed method respectively. Here outer and inner white boxes represent vehicle before and after shadow removal respectively. In Fig. 2 (d), (e), and (f), shadows were not removed well since there are little chromaticity or strong texture component under the shadow which these methods assume. In Fig. 2 (g), the right part of the vehicle and back bumper covered with self shadow were removed. CIHD uses RGB color model and is not able to differentiate between self and cast shadows. Fig. 2 (h) shows the result of the proposed method. Most of the self



**Fig. 2.** Shadow removal results of 5 methods for test image 1

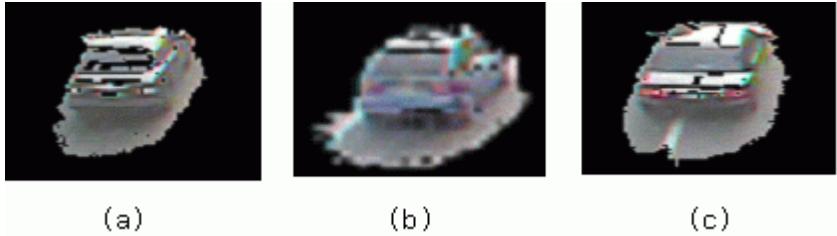


**Fig. 3.** Shadow removal results of 5 methods for test image 2

shadow was preserved since shadow-invariant  $C_1 C_2 C_3$  was used. In Fig. 3 (d), (e), and (f), outer and inner white boxes are nearly overlapped meaning that shadows were rarely removed due to the same reason as in Fig. 2 (d), (e), and (f). In Fig. 3 (g), right part of the vehicle covered with self shadow is almost removed whereas the proposed method keeps the self shadow and removes only cast shadow as shown in Fig. 3 (h).

## 4.2 A New Metric for Shadow Removal Method Evaluation

Fig. 4 (a), (b), and (c) are test images for various shadow removal method evaluation. Fig. 4 (a) and (b) correspond to Fig. 3 (a) and Fig. 2 (a) respectively. Fig. 4 (c) is added for performance evaluation.



**Fig. 4.** Test images for performance evaluation

For quantitative evaluation, we compute  $\eta$  and  $\xi$  suggested by Prati et al.[8] and a new metric  $\gamma$ . Table 1 shows manually obtained ground truth data for each image in Fig. 4. For example, Fig. 4 (a) has 1,737 foreground pixels among which 997 are vehicle pixels and 740 are shadow pixels.

**Table 1.** Ground Truth data

Imag e	GT <sub>f</sub>	GT <sub>s</sub>	GT
(a)	997	740	1,737
(b)	549	469	1,018
(c)	1,312	1,115	2,427

**Table 2.** TP values for 5 methods

imag e	NRGB		GDC		CNCC		CIHD		The propo sed	
	TP <sub>f</sub>	TP <sub>s</sub>								
(a)	757	350	963	41	816	295	517	681	534	732
(b)	525	190	537	59	506	157	251	375	405	435
(c)	1,029	576	1,266	76	1,176	337	814	1,030	1,008	1,106

Table 2 shows TP values for vehicle and shadow classified by 5 methods. Taking image (a) as an example, NRGB found 757 vehicle pixels out of 997 and 350 shadow pixels out of 740.

Table 3 shows the results of two conventional and one proposed metrics. In Table 3 (a), (b), and (c),  $\eta$ ,  $\xi$ , and  $\gamma$  are computed respectively. In Fig. 2, we saw that CIHD and the proposed yield better results than the other three, but in Table 3 (b),

**Table 3.** Metric values for 5 methods

	(a) $\eta$ (detection rate)						(b) $\xi$ (discrimination rate)					
image	NRG B	GD C	CNC C	CIH D	The proposed		NRG B	GD C	CNC C	CIH D	The proposed	
(a)	0.47	0.06	0.40	0.92	0.98		(a)	0.76	0.97	0.82	0.52	0.54
(b)	0.41	0.13	0.34	0.80	0.93		(b)	0.96	0.98	0.92	0.46	0.73
(c)	0.52	0.07	0.30	0.92	0.99		(c)	0.78	0.97	0.90	0.62	0.73

	$\gamma$ (new metric)					
image	NRGB	GDC	CNCC	CIHD	The proposed	
(a)	0.64	0.58	0.64	0.69		0.73
(b)	0.70	0.59	0.65	0.62		0.82
(c)	0.66	0.55	0.62	0.76		0.85

they have smaller  $\xi$ . That is, the method with poorer result has better performance value. This happened because NRGB, GDC, and CNCC classified most of the foreground pixels as vehicle pixels and  $\xi$  is not able to handle this situation. On the contrary, Table. 3 (c) shows that the method with better performance has better evaluation value and that the proposed is the best. Moreover the qualitative performance that can be observed in Fig. 2 and 3 is consistent with quantitative measure shown in Table 3 (c).

## 5 Conclusion and Future Research

In this paper, we proposed an invariant color-based shadow removal method and a new metric to evaluate the performance of various shadow removal methods. For shadow removal, we first label blob as either color blob or non color blob. Then we apply the proposed method to color blob and CIHD[4] to non color blob, and obtained good results. To compare the performance of various removal methods, we propose a new metric and show that this metric overcame the inconsistency problem of conventional metrics. The proposed shadow removal method has a difficulty when vehicles having self shadow are of gray tone. This is intended for future research.

## References

- [1] J. M. Stephen, J. Sumer, and D. Zoran, "Tracking Groups of People", Computer Vision and Image Understanding 80, pp.42-56, 2000
- [2] O. Javed and M. Shah, "Tracking and Object Classification for Automated Surveillance", Proc. of Seventh European Conference on Computer Vision, 2002

- [3] G. Daniel, F. Jan-Michael and K. Reinhard, "A Color Similarity Measure for Robust Shadow Removal in Real Time", Proc. of Vision, Modeling, and Visualization, pp.253-260, 2003
- [4] H. S. Choi, "A New Shadow Removal Method in Color Traffic Image Sequence ", Master's thesis, Myongji University, 2004
- [5] C. Stauffer, and W. Grimson," Adaptive Background Mixture Models for Real Time Tracking", Proc. Computer Vision and Pattern Recognition, vol. 2, pp. 246-252 , 1999
- [6] E. Salvador, A. Cavallaro, and T. Ebrahimi, "Shadow Identification and classification using invariant color models", Proc. of Acoustics, Speech, and Signal Processing, Vol. 3, pp.1545-1548, May. 2001
- [7] I. Cohen and G. Medioni, "Detecting and tracking moving objects for video surveillance", Computer Vision and Pattern Recognition, Vol. 2, pp.319-325, 1999
- [8] A. Prati, I. Mikic, Mohan M. Trivedi, and R. Cucchiara, "Detecting Moving Shadow: Algorithms and Evaluation" IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 25, pp.918-923, July. 2003

# Uncontrolled Face Recognition by Individual Stable Neural Network

Xin Geng<sup>1</sup>, Zhi-Hua Zhou<sup>2</sup>, and Honghua Dai<sup>1</sup>

<sup>1</sup> School of Engineering and Information Technology,  
Deakin University, VIC 3125, Australia  
`{xge, hdai}@deakin.edu.au`

<sup>2</sup> National Laboratory for Novel Software Technology,  
Nanjing University, Nanjing 210093, China  
`zhouzh@nju.edu.cn`

**Abstract.** There usually exist diverse variations in face images taken under uncontrolled conditions. Most previous work on face recognition focuses on particular variations and usually assume the absence of others. Such work is called *controlled face recognition*. Instead of the ‘divide and conquer’ strategy adopted by controlled face recognition, this paper presents one of the first attempts directly aiming at *uncontrolled face recognition*. The solution is based on Individual Stable Neural Network (ISNN) proposed in this paper. ISNN can map a face image into the so-called Individual Stable Space (ISS), the feature space that only expresses personal characteristics, which is the only useful information for recognition. There are no restrictions for the face images fed into ISNN. Moreover, unlike many other robust face recognition methods, ISNN does not require any extra information (such as view angle) other than the personal identities during training. These advantages of ISNN make it a very practical approach for uncontrolled face recognition. In the experiments, ISNN is tested on two large face databases with vast variations and achieves the best performance compared with several popular face recognition techniques.

## 1 Introduction

Despite the success of many face recognition systems [1,5,7,13], a lot of issues still remain to be addressed. Among those issues, perhaps the most prominent one is that most systems require the face images fed to them to satisfy certain ‘rules’, such as in a particular range of view angle, in homogeneous illumination and without any occlusions. We call such systems *controlled face recognition* systems. The control rules greatly restrict the commercialization of face recognition techniques because most real applications, such as intelligent surveillance, cannot satisfy such strict rules. What the real world needs are systems that can recognize any face images recognizable by human beings. We call such systems *uncontrolled face recognition* systems.

As a matter of fact, the developing history of face recognition techniques is the march from controlled conditions to more and more uncontrolled conditions.

Most early algorithms [1,5,7,13] can handle expression variation well but suffer from other variations. Later, a lot of methods [2,3,10,11,15] were proposed to tackle view angle and illumination variations. Recently, a few works have been emerging to remove occlusion [14] and simulate aging effect [4]. Although the treatable variations are more and more complex, most of these ingenious methods yet have to assume the absence of other possible variations. The methodology adopted by existing work appears to be ‘divide and conquer’, i.e. gradually reduce the restrictions through tackling possible variations one by one. However, in practice, a number of variations are often complicatedly interlaced. The combination of several algorithms each of which handles particular variations well will not necessarily result in a robust system against all variations.

Instead of ‘divide and conquer’, this paper presents one of the first attempts along the ‘unite and conquer’ strategy, i.e. directly target to uncontrolled face recognition. Since variations in uncontrolled face recognition might be too complex to be well handled by currently available mathematical tools, we avoid explicitly modeling different kinds of variations. Instead, we focus on the information which is useful for face recognition and try to filter out all other information. This is achieved by a multilayer neural network named Individual Stable Neural Network (ISNN).

The rest of this paper is organized as follows. In section 2, the extraction of personal characteristics is discussed. ISNN is proposed for uncontrolled face recognition in section 3. The experimental results are reported and analyzed in section 4. Finally in section 5, conclusions are drawn and the main future work is indicated.

## 2 Extraction of Personal Characteristics

The information conveyed by any face image<sup>1</sup> might be categorized into four kinds:

1. *Personal characteristics* (denoted by  $I_{personal}$ ), i.e. the characteristics that make one person look different from others;
2. *Common facial characteristics* (denoted by  $I_{facial}$ ), i.e. the characteristics shared by all faces;
3. *Face status* (denoted by  $I_{status}$ ), i.e. any changes a particular face may undergo, such as expressions, aging effects, glasses, scars, etc.;
4. *Imaging configuration* (denoted by  $I_{imaging}$ ), i.e. the conditions under which the face is imaged, such as illumination, view angle, etc..

Among them,  $I_{personal}$  is the only useful one for recognition. Thus the key step of any face recognition methods should be the extraction of  $I_{personal}$ , explicitly or inexplicitly.

The four kinds of information contained in a set of face images can be divided into two groups, i.e. *variable information* and *stable information*. Traditional

---

<sup>1</sup> Here the face image refers to normalized face image, i.e. only the face region is contained in the image.

research on face recognition mainly focuses on the variable information in a multi-personal face image set. In this case,  $I_{facial}$  is out of the game first. The goal is set as distinguishing the variation of  $I_{personal}$  from that of  $I_{status}$  and  $I_{imaging}$ . Nontrivial work naturally starts from the relatively easier cases when the variations of  $I_{status}$  and  $I_{imaging}$  are partially restricted, i.e. the cases of controlled face recognition.

Under uncontrolled conditions, the possible variations of  $I_{status}$  and  $I_{imaging}$  seem too complex to be efficiently modelled. Instead we try to ‘filter out’ them.  $I_{status}$  and  $I_{imaging}$  are always in the group of variable information, which prompts us to shift our attention to the other group, stable information. If the face images all come from a same person, then both  $I_{personal}$  and  $I_{facial}$  are stable. Noticing that  $I_{facial}$  is always stable in a face image set, we find a way to remove  $I_{facial}$  before  $I_{status}$  and  $I_{imaging}$ . Suppose the information contained in a multi-personal face image set is denoted by  $I_{multi}$ , then

$$I_{multi} = \underline{\underline{I_{status}}} + \underline{\underline{I_{imaging}}} + \underline{I_{personal}} + \underline{I_{facial}}, \quad (1)$$

where the double-underlined terms are the variable information, and the single-underlined terms are the stable information. Suppose we can construct a feature space  $F_v$  that filters out the information stable in the image set, then the information contained in the projections of the image set in  $F_v$ ,  $I_{multi}^p$ , will be

$$I_{multi}^p = \underline{\underline{I_{status}}} + \underline{\underline{I_{imaging}}} + \underline{I_{personal}}, \quad (2)$$

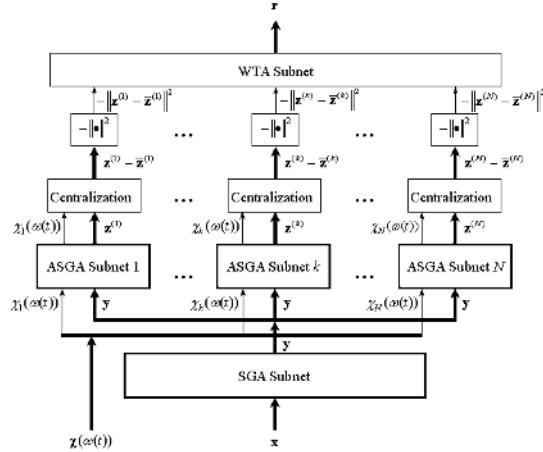
which means that  $I_{facial}$  has been removed in  $F_v$ . If subsequently the projections are divided into subsets each of which is a single-personal set, then the information contained in each subset,  $I_{sub}^p$ , will be

$$I_{sub}^p = \underline{\underline{I_{status}}} + \underline{\underline{I_{imaging}}} + \underline{I_{personal}}. \quad (3)$$

Now only  $I_{personal}$  is the stable information. If a second feature space  $F_{is}$  that filters out the variable information is constructed on the subset of a particular person, then the information contained in the projections in  $F_{is}$ ,  $I_{sub}^{pp}$ , will be only  $I_{personal}$ :

$$I_{sub}^{pp} = \underline{I_{personal}}. \quad (4)$$

Such feature space  $F_{is}$  is called Individual Stable Space (ISS) of that person because of two reasons. First, since  $I_{status}$  and  $I_{imaging}$  have been removed, all face images of that particular person are expected to be stable in  $F_{is}$ . Second, since  $I_{facial}$  has also been removed, if the face images from other persons are projected into  $F_{is}$ , the projections are expected to be unstable. Thus ISS can be used to design an uncontrolled face recognition system. The next section will describe how to map a face image into ISS and recognize it by the Individual Stable Neural Network (ISNN).



**Fig. 1.** The architecture of the ISNN for uncontrolled face recognition. The thick lines represent vector signals, and the thin lines represent scalar signals.

### 3 ISNN for Uncontrolled Face Recognition

#### 3.1 Individual Stable Neural Network

The construction of ISS involves two kinds of feature spaces. The first is the feature space that filters out the information stable in the training set (the projection from Eq. 1 to Eq. 2). The second is the feature space that filters out the information variable in the training set (the projection from Eq. 3 to Eq. 4). In ISNN, these two kinds of feature spaces are implemented by a pair of neural networks with opposite learning rules, namely SGA network [9] and ASGA network [16].

The architecture of ISNN, as shown in Fig. 1, is designed according to the extraction procedure of  $I_{personal}$  described in section 2. The raw face image  $\mathbf{x} = [x_1, x_2, \dots, x_n]$  ( $x_i$  represents the intensity of the pixels in the face region) is first input into the SGA subnet to get its projection  $\mathbf{y} = [y_1, y_2, \dots, y_p]$  in the feature space  $F_v$ . Then  $\mathbf{y}$  is input into the  $N$  (the number of different individuals) ASGA subnets together with the supervisory signal  $\chi(\omega(t)) = [\chi_1(\omega(t)), \chi_2(\omega(t)), \dots, \chi_N(\omega(t))]$ . Suppose the personal ID of a particular projection  $\mathbf{y}(t)$  is  $\omega(t)$ , then  $\chi_k(\omega(t))$  is defined by

$$\chi_k(\omega(t)) = \begin{cases} 1, & \text{when } \omega(t) = k; \\ 0, & \text{when } \omega(t) \neq k. \end{cases} \quad (5)$$

The output of the  $k$ -th ASGA subnet,  $\mathbf{z}^{(k)} = [z_1^{(k)}, z_2^{(k)}, \dots, z_m^{(k)}]$ , will be the projection in the ISS of person  $k$ . After centralization and negative normalization, the  $N$  scalar signals  $-\|\mathbf{z}^{(k)} - \bar{\mathbf{z}}^{(k)}\|^2$ ,  $k = 1 \dots N$  are sent to a winner-take-all (WTA) subnet to choose the largest one.

The SGA network has  $p$  parallel neurons each of which is associated with a weight vector  $\mathbf{w}_j$ . The learning rule of the SGA network is given by

$$\Delta \mathbf{w}_j(t-1) = \alpha_1 y_j(t)[\mathbf{x}(t) - y_j(t)\mathbf{w}_j(t-1) - 2 \sum_{i < j} y_i(t)\mathbf{w}_i(t-1)], \quad (6)$$

where  $y_j(t) = \mathbf{w}_j^T(t-1)\mathbf{x}(t)$  and  $0 < \alpha_1 < 1$  is the learning rate. As proved by Oja [8], for  $t \rightarrow \infty$ , the vectors  $\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_p$  will converge to the principal components of the input data stream. As the first step of uncontrolled face recognition, the utility of the SGA network is to remove the common facial characteristics  $I_{facial}$  because the feature space spanned by principal components mainly reserves variable information while  $I_{facial}$  is stable information.

The ASGA network uses the opposite learning rule of the SGA network. It can be viewed as an anti-Hebbian version of SGA. Since in the second stage of the extraction of  $I_{personal}$ , the projections in the face space need to be divided according to personal IDs, a supervisory signal  $\chi_k(\omega(t))$  should be integrated into the learning rule, which is given by

$$\begin{aligned} \Delta \mathbf{w}_j^{(k)}(t-1) = & -\alpha_2 \chi_k(\omega(t)) z_j^{(k)}(t)[\mathbf{y}(t) - z_j^{(k)}(t)\mathbf{w}_j^{(k)}(t-1)/\|\mathbf{w}_j^{(k)}(t-1)\|^2 \\ & - 2 \sum_{i < j} z_i^{(k)}(t)\mathbf{w}_i^{(k)}(t-1)], \end{aligned} \quad (7)$$

where  $z_j^{(k)}(t) = \mathbf{w}_j^{(k)T}(t-1)\mathbf{y}(t)$  and  $0 < \alpha_2 < 1$  is the learning rate. It was proved [17] that for  $t \rightarrow \infty$ ,  $\mathbf{w}_j^{(k)}$  will converge to the least variable components of the input data. Such components are called minor components [16]. Readers are referred to [9] and [16] for more details on the SGA and ASGA network. Just as principal components retaining the information variable in the data set, minor components retain the information stable in the data set. In the subset, as shown in Eq. 3, only  $I_{personal}$  is stable information, thus the output feature  $\mathbf{z}^{(k)}$  can be viewed as the projection in ISS.

In the training phase of ISNN, all the training face images and the corresponding personal IDs are input into the initialized ISNN to update the weights. After convergence, all the training images and IDs go through the ISNN again without updating the SGA and ASGA subnets to calculate the mean vectors  $\bar{\mathbf{z}}^{(k)}, k = 1, \dots, N$ . Note that in the whole training procedure of ISNN, no extra information except the personal IDs is needed. This advantage is extremely important since under uncontrolled conditions, the accurate estimation of such extra information is alone a big problem. One might argue that ISNN requires to train one network for each person, thus the training procedure is inefficient for large databases. However, ISNN adopts the so-called One-Class-One-Network (OCON) structure, which has certain advantages over the All-Class-One-Network (ACON) structure, such as less hidden units, faster convergence, and better generalization [18]. Moreover, such architecture can easily benefit from distributed computing. Thus efficiency should not be a problem for ISNN.

In the testing phase, the unknown face image  $\mathbf{x}$  directly go through the ISNN. The output vector  $\mathbf{r}$  indicates in which ISS the projection is most stable, and consequently  $\mathbf{x}$  is recognized as from the individual associated to that ISS.

### 3.2 Relationship to Existing Work

The architecture of ISNN is somehow similar to the Probabilistic Decision-Based Neural Network (PDBNN) [5] because both of them adopt the OCON structure. However, the effectiveness of PDBNN relies on the ability of the mixture of Gaussians to approximate any data distribution, while the effectiveness of ISNN relies on the extraction of the only useful information for recognition, i.e.  $I_{personal}$ .

The Eigenface method [13] is similar to the SGA subnet in the ISNN. However, in case of uncontrolled face recognition, neither the purpose nor computation is the same. The purpose of the SGA subnet is to remove  $I_{facial}$  rather than find  $I_{personal}$  in the image set. The computation is changed from eigen decomposition to recursive learning because the vast possible variations in uncontrolled conditions consequentially require a large number of training images, which makes the SVD procedure of Eigenface no longer tractable.

The idea of personalized subspace was first proposed as Face Specific Subspace (FSS) [11], which constructs an eigenspace for each individual and then recognizes faces with reconstruction error. FSS is actually similar to the ASGA subnets in the ISNN. There are two main advantages of ISNN over FSS. The first is that ISNN also filters out  $I_{facial}$  while FSS doesn't. The second is that ISNN can explicitly get the projections in the ISS. This provides much more room for further improvements.

The ‘unite and conquer’ strategy was once adopted in controlled face recognition by the Bayesian face recognition method [6]. However, in uncontrolled face recognition, the possible cases of both extrapersonal and intrapersonal differences will exponentially grow to an unmanageable size. On the other hand, ISNN avoids directly modeling different kinds of information and instead tries to filter out all useless information.

Fisherface [1] tries to find a global feature space that maximizes the ratio of the extrapersonal difference and the intrapersonal difference. However, single linear subspace might not be powerful enough for uncontrolled face recognition. Thus ISNN uses multiple personalized subspaces to compensate the deficiency of single linear subspace.

## 4 Experiments

### 4.1 Methodology

In the experiments, ISNN is compared with those closely related methods described in section 3.2 and some of their variants by three-fold cross validation.

Two databases are used in the experiments. The first is the CMU PIE database [12]. The face images are greatly different in pose, illumination and expression. Note that although the images in this database are obtained under controlled



**Fig. 2.** Typical face images from (a) the CMU PIE database, (b) the UFR face database

conditions, the additional information (pose, illumination and expression) is not used to train ISNN. Thus it can be viewed as an uncontrolled face database. There are totally 38,707 images from 68 individuals used in our experiments. The normalized face image has  $66 \times 46$  pixels. The other data set is used to test the algorithms in another case: fewer individuals but with more variations. We have collected 23,978 images from 14 individuals through a web camera to compose the UFR face database<sup>2</sup>. This database attempts to simulate most possible variations (pose, illumination, expression, occlusion, device noise, and inaccurate face detection) in real face recognition applications. The cropped face image has  $55 \times 42$  pixels. Some typical face images are shown in Fig. 2.

There are remarkable illumination and pose variations in both databases. Among the methods described in section 3.2, only Eigenface is not specially designed to deal with illumination variation. It has been reported that discarding the first few eigenfaces will endow Eigenface with certain ability to handle illumination variation [1]. This is tested here by discarding the first three eigenfaces, which is denoted by Eigenface-3. As for the pose variation, except for ISNN, PDBNN and FSS, none of the other methods is designed for the multi-view case. So we extend a multi-view version for each of them in a way similar to the View-based Eigenface [10] (abbreviated as V-Eigenface). The multi-view algorithms are denoted by V-Bayes, V-Fisherface and V-(Eigenface-3) respectively.

In the experiments, the parameters of PDBNN are empirically determined through several trials. When the best performance is observed, the number of Gaussians for each individual is set to 6, the learning rate for the Gaussian centers is set to  $10^{-6}$ , the learning rate for the variance is set to  $10^{-4}$ , the learning rate for the threshold is set to 0.05, and the penalty function for the threshold is the sigmoid function. As for the Bayes method, similar to [6], we randomly sample 1,000 intrapersonal difference images and 4,000 extrapersonal difference images as the training set. If not explicitly stated, the number of principal components  $p$  is set to 50, and the number of minor components  $m$  is set to 20. The initial weight vectors  $\mathbf{w}_j(0)$  in the SGA or ASGA subnet are set to random orthogonal unit vectors. The learning rates in Eqs. 6 and 7 are set as  $\alpha_1 = \alpha_2 = 0.01$ .

<sup>2</sup> This database will soon be publicly available.

**Table 1.** Recognition Rates (in %) From Rank 1 to Rank 3 on the PIE Database

Methods	Without Pose Inf.			With Pose Inf.		
	Rank 1	Rank 2	Rank 3	Rank 1	Rank 2	Rank 3
ISNN	<b>94.16</b>	<b>96.59</b>	<b>97.32</b>	<b>94.16</b>	<b>96.59</b>	<b>97.32</b>
PDBNN	81.70	88.44	91.45	81.70	88.44	91.45
FSS	89.30	92.62	94.04	89.30	92.62	94.04
Bayes	18.58	25.23	31.36	18.58	25.23	31.36
Fisherface	57.96	67.25	72.89	57.96	67.25	72.89
Eigenface	30.27	39.54	45.84	30.27	39.54	45.84
Eigenface-3	45.31	55.24	61.10	45.31	55.24	61.10
V-Bayes	N/A	N/A	N/A	48.86	61.33	68.56
V-Fisherface	N/A	N/A	N/A	89.88	93.36	94.77
V-Eigenface	N/A	N/A	N/A	65.92	72.18	75.58
V-(Eigenface-3)	N/A	N/A	N/A	85.11	88.85	90.60

## 4.2 Results

The recognition rates from rank 1 to rank 3 on the PIE database are tabulated in Table 1. The 11 algorithms are compared in two cases: with and without pose information. The best performance in each case is bolded. Note that the first 7 algorithms do not use pose information, so the results in the two cases are same.

When pose information is not available, the best performance is achieved by ISNN, which is about 5% higher in rank 1 rate than the runner-up, FSS. The superiority of ISNN over FSS mainly comes from the SGA subnet of the ISNN, which removes  $I_{facial}$ . It is also worth mentioning that FSS uses a 50-dimensional subspace while ISNN only uses a 20-dimensional subspace to describe each person. Thus ISNN is much faster and requires less storage. PDBNN performs worse than both ISNN and FSS because under uncontrolled condition, the distribution of the face images is so complicated that the gradient descent learning of PDBNN will tend to fall into local optimization. The Bayes method results in poor performance, which is not surprising since the sampled difference images are only a small portion of all possible differences. Fisherface performs best among the three single-subspace methods. Finally, Eigenface-3 performs much better than Eigenface. It can be found that there is a remarkable gap between the recognition rates of the best three methods (ISNN, FSS and PDBNN) and those of the others, which indicates that the personalized approach might be a suitable solution to the problem of uncontrolled face recognition.

When pose information is given, ISNN still performs the best. This is impressive because it does not use the additional information which has been exploited by the view-based algorithms. With certain ability to handle the pose variation, all the view-based variants make remarkable improvements over the corresponding original algorithms. Among them, V-Fisherface achieves the highest recognition rate, which marginally exceeds that of FSS. But in practice, the pose information is not always available, especially under uncontrolled conditions. This greatly enlarges the superiority of ISNN over V-Fisherface.

**Table 2.** Recognition Rates (in %) From Rank 1 to Rank 3 on the UFR Database

Methods	Without Pose Inf.			With Pose Inf.		
	Rank 1	Rank 2	Rank 3	Rank 1	Rank 2	Rank 3
ISNN	<b>98.65</b>	<b>99.51</b>	<b>99.79</b>	<b>98.65</b>	<b>99.51</b>	<b>99.79</b>
PDBNN	96.84	99.04	99.54	96.84	99.04	99.54
FSS	96.79	98.52	98.99	96.79	98.52	98.99
Bayes	39.52	59.96	73.80	39.52	59.96	73.80
Fisherface	91.32	96.56	98.10	91.32	96.56	98.10
Eigenface	68.18	80.63	86.62	68.18	80.63	86.62
Eigenface-3	75.24	86.15	90.06	75.24	86.15	90.06
V-Bayes	N/A	N/A	N/A	53.71	74.76	85.78
V-Fisherface	N/A	N/A	N/A	96.41	98.56	99.26
V-Eigenface	N/A	N/A	N/A	76.01	86.05	90.26
V-(Eigenface-3)	N/A	N/A	N/A	83.56	90.60	93.06

The recognition rates on the UFR database are tabulated in Table 2. With much fewer classes, although more variations are presented, almost all algorithms achieve better performances than those on the PIE database. The comparative results are similar with those in Table 1. When pose information is not available, ISNN is still the best one. PDBNN achieves a good performance just next to that of ISNN, and better than that of FSS. This might be because that with fewer classes, the mixture of Gaussians learned by PDBNN is enough to separate different classes. When pose information is given, there is still no other algorithm exceeds ISNN. Again V-Fisherface achieves the best performance among the view-based algorithms.

## 5 Conclusions

This paper presents one of the first approaches toward uncontrolled face recognition. The main contributions includes: (1) The ISS is proposed as a general framework for uncontrolled face recognition; (2) ISNN is designed as a neural network implementation of the ISS; (3) The first uncontrolled face database UFR is introduced.

The implementation of ISS is not limited to ISNN. As mentioned above, the ISS-based approach can be viewed as a general framework for uncontrolled face recognition. Other novel subspace methods, including both linear and nonlinear ones, might be developed to implement ISS. This will be one of our major future work following this paper.

## Acknowledgements

Part of the work was done when X. Geng was at the LAMDA group, Nanjing University. Z.-H. Zhou was partially supported by the Fok Ying Tung Education Foundation (91067) and the EYTP of MOE of China.

## References

1. Belhumeur, P.N., Hespanha, J.P., Kriegman, D.J.: Eigenfaces vs. fisherfaces: recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1997, 19(7): 711-720.
2. Geng, X. Zhou, Z.-H.: Image region selection and ensemble for face recognition. *Journal of Computer Science and Technology*, 2006, 21(1): 116-125.
3. Gross, R., Matthews, I., Baker, S.: Appearance-based face recognition and light-fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2004, 26(4): 449-465.
4. Lanitis, A., Taylor, C.J., Cootes, T.F.: Toward automatic simulation of aging effects on face Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2002, 24(4): 442-455.
5. Lin, S.H., Kung, S.Y., Li, L.J.: Face recognition/detection by probabilistic decision-based neural network. *IEEE Transactions on Neural Networks*, 1997, 8(1): 114-132.
6. Moghaddam, B., Jebara, T., Pentland, A.: Bayesian face recognition. *Pattern Recognition*, 2000, 33(11): 1771-1782.
7. Moghaddam, B., Pentland, A.: Probabilistic visual learning for object representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1997, 19(7): 696-710.
8. Oja, E.: Subspace Methods of Pattern Recognition, England: Research Studies Press and John Wiley & Sons, 1983.
9. Oja, E.: Principal components, minor components, and linear neural networks. *Neural Networks*, 1992, 5: 927-935.
10. Pentland, A., Moghaddam, B., Starner, T.: View-based and modular eigenspaces for face recognition. In: *Proceedings of Conference on Computer Vision and Pattern Recognition*, Seattle, WA, 1994, pp. 84-91.
11. Shan, S., Gao, W.: Face identification based on face-specific subspace. *International Journal of Imaging and System Technology*, Special issue on face processing, analysis and synthesis, 2003, 13(1): 23-32.
12. Sim, T., Baker, S., Bsat, M.: The CMU pose, illumination, and expression database. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2003, 25(12): 1615 - 1618.
13. Turk, M., Pentland, A.: Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 1991, 3(1): 71-86.
14. Wu, C., Liu, C., Shum, H.Y., Xu, Y.Q., Zhang, Z.: Automatic eyeglasses removal from face images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2004, 26(3): 322-336.
15. Wu, J., Zhou, Z.-H.: Face recognition with one training image per person. *Pattern Recognition Letters*, 2002, 23(14): 1711-1719.
16. Xu, L., Krzyzak, A., Oja, E.: Neural nets for dual subspace pattern recognition method. *International Journal of Neural Systems*, 1991, 2: 169-184.
17. Xu, L., Oja, E., Suen, C.: Modified Hebbian learning for curve and surface fitting. *Neural Networks*, 1992, 5(3), 441-457.
18. Zhao, W., Chellappa, R., Phillips, P. J., Rosenfeld, A.: Face recognition: a literature survey. *Acm Computing Surveys*, 2003, 35(4): 399-459.

# Fuzzy Velocity-Based Temporal Dependency for SVM-Driven Realistic Facial Animation

Pith Xie<sup>1,2</sup>, Yiqiang Chen<sup>1</sup>, Junfa Liu<sup>1</sup>, and Dongrong Xiao<sup>2</sup>

<sup>1</sup> Institute of Computing Technology, Chinese Academy of Sciences,  
Postfach 10 00 80,  
Beijing, China

pithxie@yahoo.com, yqchen@ict.ac.cn, jfliu@shict.cn

<sup>2</sup> Department of Information Science and Communication, Nanjing University  
of Information Science and Technology,  
Postfach 21 00 44,  
Nanjing, China  
xiaodongrong44@163.com

**Abstract.** Driving a realistic facial animation with Support Vector Machine(SVM) requires determining the shape-to-wrinkle correspondence, which includes not only spatial dependency, but also temporal dependency. A few available frameworks(e.g., *Recurrent Neural Network* and *Long Short-Term Memory*), represent temporal dependency as the dependency of output on position input series, which however may bring about spatial redundancy in some cases. We argue that temporal dependency should be represented as the dependency of output on velocity input series. Besides, due to the weak temporal dependency between shape change and wrinkle change, we put forward *Fuzzy Embedding* to convert velocity into fuzzy velocity. The shape-wrinkle synthesis demonstrates that, in determining the temporal dependency between wrinkle change and shape change, fuzzy velocity provides more valuable information than velocity and thus enhances the degree of the realism effectively.

## 1 Introduction

Driving a realistic facial animation requires uncovering the facial dynamics, which determines the shape-to-wrinkle correspondence. The ways to represent facial dynamics can be divided into three groups: image-based methods [1,2,3], physically-based methods [4,5] and machine learning method [6].

Avoiding tedious and manual process, machine learning method provide a straightforward way to uncover facial dynamics, which will subsequently be used to synthesize rich expressions. Zhu et al. [6], representing shape change as Facial Animation Parameters(FAP) and wrinkle change as *Eigen Expression Ratio Image*(EigenERI), train a SVM to learn the spatial dependency between FAP and EigenERI. Due to the fact that both shape and wrinkle change temporally, there exists temporal dependency between FAP and EigenERI in addition.

Many domains in machine learning involve discovering dependencies and structure over time. To capture the temporal dependency present in time series,

a variety of frameworks have been constructed by extending standard learning models, e.g., neural network. Jordan M.I. [7] proposed *Recurrent Network* to learn short-term temporal dependency. Hochreiter, S. and Schmidhuber, J. [8] proposed *Long Short-Term Memory*(LSTM) to learn long-term temporal dependency. Based on the fact that Hidden Markov Model(HMM) can encode long-term temporal dependency, Bengio, Y. and Frasconi, P. [9] proposed Input Output HMM (IOHMM) to learn long-term temporal dependency.

All those frameworks encode temporal dependency as various recurrent networks and succeed in solving the specific problem they aim at. As a whole, those frameworks are stimulated only by position input, that is to say, the position output depends on the position input series. However, position input series brings about spatial redundancy while changing slowly over time. In that case, those frameworks will no longer be sensitive to temporal dependency.

According to observation, both shape and wrinkle change slowly over time. Therefore, it is necessary to reduce spatial redundancy brought about by shape change. In this paper, temporal dependency is considered as the dependency of output on velocity input series while spatial dependency as the dependency of output on current position input. Velocity, representing temporal change hidden in the position input series, can effectively reduce the spatial redundancy. Furthermore, to capture the weak temporal dependency between wrinkle change and shape change, we propose *Fuzzy Embedding* to transform velocity into fuzzy velocity.

The paper is organized as follows. In section 2, we introduce temporal dependency and present our approach to capture weak temporal dependency. In section 3, we provide result of Mackey-Glass experiment to support our argument. In section 4, we provide demonstration of shape-wrinkle synthesis to support our argument. Section 5 gives the conclusion.

## 2 Temporal Dependency

To capture the correspondence between two time series, learning model should take into account not only spatial dependency, but also temporal dependency. Given an input time series  $X(t)$  and an output time series  $Y(t)$ , for example, spatial dependency means that current position output  $Y(t)$  depends on current position input  $X(t)$  while temporal dependency means that current position output  $y(t)$  depends on not only current position input  $x(t)$ , but also previous position input  $x(t - \Delta t), x(t - 2\Delta t), \dots$

### 2.1 Spatial Redundancy

Position input describes a stationary spatial state. While varying rapidly over time, position input series represents temporal dependency effectively. While varying slowly over time, however, position input series involves much repeated spatial content and thus brings about spatial redundancy.

For example, given an 1-dimension position input series: [5.2], [5.6], [5.4], [5.2], the repeated spatial content can be approximately quantized into value 5 due to

the big proportion it occupies in all position inputs. When position input series varies slowly over time, repeated spatial content inside it will result in spatial redundancy which finally acts as a noise. Though normalization can reduce global spatial redundancy, it still reserves local spatial redundancy.

## 2.2 Velocity

Reflecting the temporal change of position input, velocity reduces repeated spatial content, i.e. spatial redundancy. So we can represent temporal dependency as the dependency of output on velocity input series.

Given a time series with a fixed interval  $\Delta t$ , the velocity at time point  $t$  can be approximated by the following calculation:

$$R(t) = \frac{X(t) - X(t - \Delta t)}{\Delta t} \quad (1)$$

where  $R(t)$  represents the velocity of the position input  $X(t)$ .

Based on Cantor Set, velocity exhibits as exact thresholds. In the case such as shape-to-wrinkle mapping, however, the temporal dependency of wrinkle change on shape change is weak. Therefore, wrinkle change depends on fuzzy velocity of shape change providing an approximate description more than velocity of shape change.

## 2.3 Fuzzy Velocity

Fuzzy Mathematics describes vague concept with membership function. Mapping infinite domain to the closed interval  $[0, 1]$ , membership function describes concept too roughly to lose much descriptive information, which however is valuable and necessary in machine learning.

Therefore, we propose *Fuzzy Embedding*, substituting for membership function, to transform velocity into fuzzy velocity so as to capture detailed fuzzy concept. Due to the operation of clustering, fuzzy velocity is vaguer than velocity. Ranging between  $[0, +\infty)$ , fuzzy velocity is more detailed than the grade of membership. The algorithm is given as follows:

1. Compute velocity of each position input in time series. Assumed these velocities amount to  $n$ ;
2. Get all velocities together as a velocity set. The *sum-of-squared-error criterion*[11] is used here as a criterion function. Denote the number of clusters as  $k$  and the corresponding *sum-of-squared-error criterion* as  $J(k)$ . Then set  $k = 1$  and cluster velocity set with K-Means;
  - If  $k \geq \frac{n}{100}$  or  $\frac{J(k)}{J(k-1)} > \epsilon$ , then goto 4;
  - Else goto 3.
3. Set  $k = k + 1$  and cluster velocity set with K-Means.
4. Denote the centers of clusters as  $m_1, m_2, \dots, m_k$ , then map each velocity  $R(t)$  to  $k$ -dimension fuzzy velocity  $R'(t)$ :

$$R'(t) = [\| R(t) - m_1 \|, \| R(t) - m_2 \|, \dots, \| R(t) - m_k \|] \quad (2)$$

The condition  $k \geq \frac{n}{100}$  ensure that the average volume of clusters be no less than 100 elements, while  $\epsilon$  is set in advance to ensure that the result of clustering can reflect the distribution of elements well.  $\| R(t) - m \|$  represents the Euclid distance between  $R(t)$  and cluster center  $m$ . The fuzzy velocity  $R'(t)$ , resulting from *Fuzzy Embedding*, reflects the similarity between velocity and each cluster.

### 3 Mackey-Glass Experiment

The intensity of temporal dependency is the degree to which the position output depends on the position input series. Mackey-Glass equation is often used in some research[12, 13] to generate stationary time series, which can simulate the transfer of the intensity of temporal dependency with several parameters being adjusted.

Mackey-Glass equation is expressed as the following formula:

$$\frac{dx(t)}{dt} = \frac{a \bullet x(t - d \bullet \Delta t)}{1 + x(t - d \bullet \Delta t)^{10}} - b \bullet x(t) \quad (3)$$

$a$ ,  $b$ ,  $d$  and  $\Delta t$  are parameters of this equation. Among them, we fix the following three parameters:  $a = 2$ ,  $b = 1$ ,  $\Delta t = 1ms$ . Parameter  $d$  represents the time-delay. The smaller  $d$  is, the greater role the previous elements play in determining current element, and the stronger the hidden temporal dependency is. On the contrary, the bigger  $d$  is, the less role the previous elements play in determining current element, and the weaker the hidden temporal dependency is.

Initially,  $\forall t \leq 0, x(t) = 9$ . Then the equation is iterated for 4005 times with an interval of  $\Delta t$  to generate a time series consists of 4005 elements. With time window size being fixed to be 5ms, we can acquire an example set consisting of 4000 temporally ordered input-output pairs, where the position input can be expressed as the following 5-dimension vector:

$$X_t = [x(t - 5), x(t - 4), \dots, x(t - 1)] \quad (4)$$

and the output can be expressed as the following 1-dimension vector:

$$Y_t = [x(t)] \quad (5)$$

In the experiment, we first train three SVMs: Standard SVM, Velocity SVM and Fuzzy Velocity SVM, then test their generalization performance.

Standard SVM takes into account only spatial dependency, which is stimulated by one and only position input. Velocity SVM takes into account both spatial dependency and temporal dependency, which is stimulated by position input as well as previous velocity input series. Fuzzy Velocity SVM also takes into account both spatial dependency and temporal dependency, which however is stimulated by position input as well as previous fuzzy velocity input series. In the following, we give detailed procedures to train and test these SVMs.

### 3.1 Standard SVM

First, normalize the example set to generate a new example set;

Second, a training set consisting of the former 2000 examples in the new example set is used to train a SVM to learn the mapping  $F : X_t \rightarrow Y_t$ ;

Third, a test set consisting of the latter 2000 examples in the new example set is used to test the generalization performance of above SVM.

### 3.2 Velocity SVM

First, incorporate three successive velocities  $V_{t-2}, V_{t-1}, V_t$  into the input of the example set, and generate new input  $X'_t$  as follows:

$$X'_t = [V_{t-2}, V_{t-1}, V_t, X_t] \quad (6)$$

Second, normalize the example set to generate a new example set;

Third, a training set consisting of the former 2000 examples in the new example set is used to train a SVM to learn the mapping  $F : X'_t \rightarrow Y'_t$ ;

Fourth, a test set consisting of the latter 2000 examples in the new example set is used to test the generalization performance of above SVM.

### 3.3 Fuzzy Velocity SVM

First, convert  $V_t$  to fuzzy velocity  $V'_t$  with *Fuzzy Embedding* under the constraint  $\epsilon = 0.99$ ;

Second, incorporate three successive fuzzy velocities  $V'_{t-2}, V'_{t-1}, V'_t$  into the input of the training set and test set, and generate new input  $X''_t$  as follows:

$$X''_t = [V'_{t-2}, V'_{t-1}, V'_t, X_t] \quad (7)$$

Third, normalize the example set to generate a new example set;

Fourth, a training set consisting of the former 2000 examples in the new example set is used to train a SVM to learn the mapping  $F : X''_t \rightarrow Y''_t$ .

Fifth, a test set consisting of the latter 2000 examples in the new example set is used to test the generalization performance of above SVM.

### 3.4 Experimental Results

The experiment is repeated on 16 example sets generated from Mackey-Glass equations with different  $d$ . Thus, we acquire 16 results as Table. 1 to compare the 3 SVMs. From left to right, column 1 stands for the value of  $d$ ; column 2, 3 and 4 stand for the value of Mean Squared Error(MSE); column 5, 6 and 7 stand for the value of Squared Correlation Coefficient(SCC). MSE reflects the local fitting effect while SCC reflects the global fitting effect. The smaller MSE is, the better the local fitting effect is. The bigger SCC is, the better the global fitting effect is.

The intensity of temporal dependency decreases gradually while  $d$  increases step by step(see Table. 1). When  $d$  increases from 1 to 5, the temporal dependency keeps strong and Velocity SVM performs the best. When  $d$  increases from

**Table 1.** Experimental results from Mackey-Glass on Standard SVM(S-SVM), Velocity SVM(V-SVM) and Fuzzy Velocity SVM(FV-SVM)

parameter	MSE			SCC		
	d	S-SVM	V-SVM	FV-SVM	S-SVM	V-SVM
1	0.0786	0.0405	0.0596	0.755	0.865	0.78
2	0.101	0.0404	0.0783	0.756	0.873	0.714
3	0.104	0.0475	0.0851	0.696	0.839	0.691
4	0.229	0.168	0.203	0.0007	0.163	0.0557
5	0.293	0.258	0.257	0.0003	0.0115	0.0105
6	0.28	0.265	0.269	0.0136	0.018	0.0221
7	0.242	0.236	0.241	0.0695	0.072	0.0771
8	0.203	0.198	0.202	0.16	0.165	0.168
9	0.173	0.17	0.171	0.252	0.255	0.264
10	0.145	0.142	0.142	0.348	0.352	0.361
11	0.13	0.129	0.128	0.395	0.398	0.406
12	0.115	0.115	0.114	0.451	0.451	0.458
13	0.103	0.103	0.102	0.502	0.502	0.508
14	0.0906	0.0906	0.0895	0.559	0.559	0.565
15	0.0819	0.0816	0.0806	0.596	0.597	0.601
16	0.073	0.0729	0.0718	0.637	0.637	0.642

6 to 16, the temporal dependency becomes weak and Fuzzy Velocity SVM performs the best on the whole. This demonstrates that fuzzy velocity, as opposed to velocity, provides more valuable information in determining the weak temporal dependency.

## 4 Shape-Wrinkle Synthesis

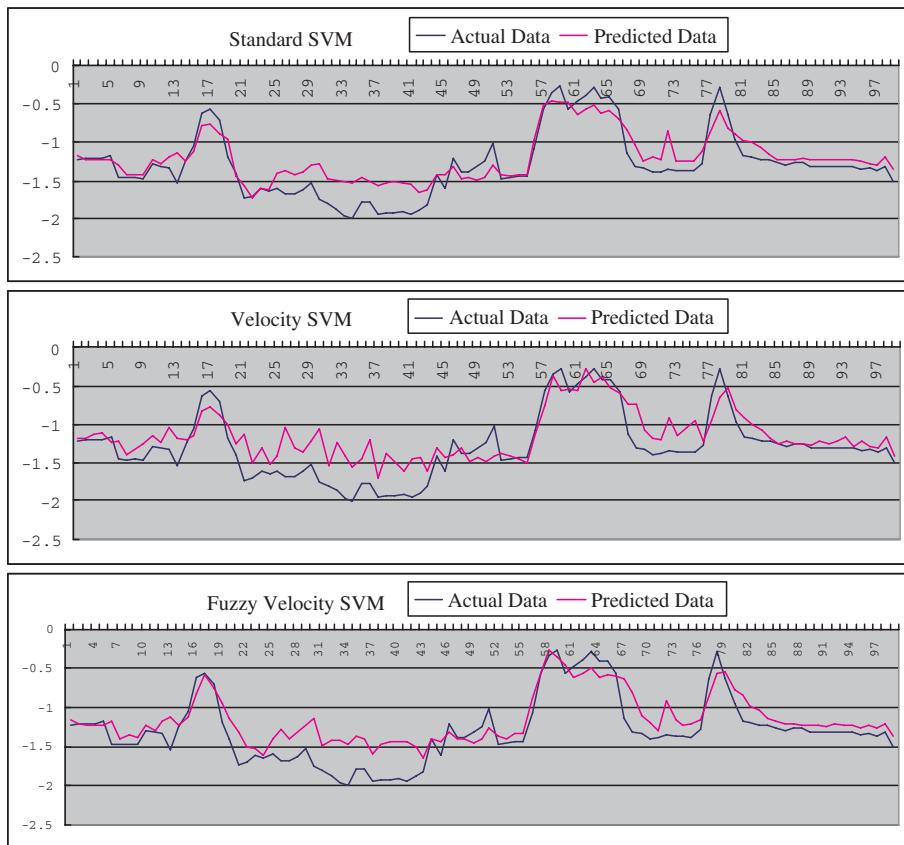
In shape-wrinkle synthesis, we represent shape change as FAP and wrinkle change as EigenERI. According to physiology, there exists causality between them which can be considered as mapping. However, there exists more or less temporal dependency between EigenERI and FAP, which belongs to weak dependency.

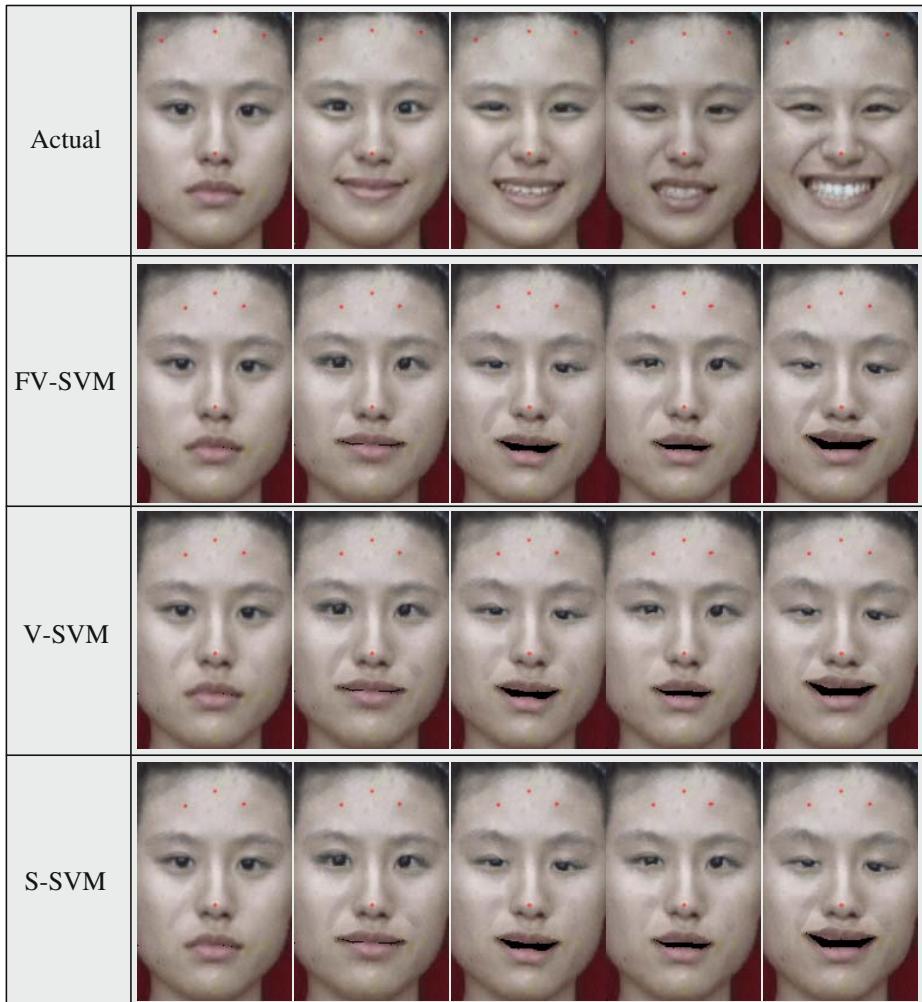
We capture a 1840-frame video containing 16 expressions with the resolution of  $130 \times 200$  pixels. In accordance with MPEG-4, we label 35 feature points and 3 rectificative points on human face. Firstly, we align all images according to rectificative points. Then we capture 43-dimension FAP by optical flow and compute ERI. High dimension make ERI difficult to be used in machine learning, therefore we extract 10-dimension EigenERI from ERI through Principal Component Analysis(PCA).

We derive a example set containing 1840 examples from 1840-frame video, where the input stands for FAP and the output stands for EigenERI. Similar to Mackey-Glass experiment, we compare three SVMs: Standard SVM, Velocity SVM and Fuzzy Velocity SVM. The experimental procedures differs from the Mackey-Glass experiment in two ways. First,  $\epsilon$  in *Fuzzy Embedding* is set to be

**Table 2.** Experimental results from shape-wrinkle synthesis on Standard SVM(S-SVM), Velocity SVM(V-SVM) and Fuzzy Velocity SVM(FV-SVM)

EigenERI index	MSE			SCC		
	S-SVM	V-SVM	FV-SVM	S-SVM	V-SVM	FV-SVM
1	0.073	0.0694	0.0641	0.858	0.867	0.876
2	0.0676	0.0643	0.0593	0.858	0.8666	0.876
3	0.0643	0.0613	0.0565	0.856	0.864	0.875
4	0.0699	0.0666	0.0616	0.856	0.865	0.874
5	0.0775	0.0739	0.0682	0.856	0.864	0.874
6	0.0468	0.0443	0.0407	0.862	0.871	0.881
7	0.0657	0.0624	0.0578	0.862	0.87	0.879
8	0.0727	0.069	0.0641	0.862	0.871	0.879
9	0.065	0.0615	0.0569	0.861	0.87	0.879
10	0.0555	0.052	0.0486	0.867	0.877	0.884

**Fig. 1.** Fitting curves generated from Standard SVM, Velocity SVM and Fuzzy Velocity SVM



**Fig. 2.** From top to bottom: true facial animation, facial animation driven by Fuzzy Velocity SVM, facial animation driven by Velocity SVM, facial animation driven by Standard SVM

0.999. Second, the test set is just the whole new example set while the training set is derived from the new example set by sampling one every three examples.

Table. 2 compares the generalization performance of the 3 SVMs. From left to right, column 1 stands for the index of component in EigenERI; column 2, 3 and 4 stand for the value of Mean Squared Error(MSE); column 5, 6 and 7 stand for the value of Squared Correlation Coefficient(SCC). As shown in Table. 2, Fuzzy Velocity SVM performs the best while Standard SVM performs the worst.

Figure. 1 shows the fitting curves of the first component in EigenERI, which are generated from Standard SVM, Velocity SVM and Fuzzy Velocity SVM

during the same interval. As shown in Figure. 1, Fuzzy Velocity SVM performs the best while Standard SVM performs the worst. Figure. 2 compares actual facial animation and the facial animations driven by Standard SVM, Velocity SVM and Fuzzy Velocity SVM. As demonstrated in Figure. 2, Fuzzy Velocity SVM performs the best while Standard SVM performs the worst.

## 5 Conclusion

To avoid spatial redundancy, we argue that output should depend on previous velocity input series and current position input, which individually provide temporal information and spatial information. When the temporal dependency is weak, however, output depends on fuzzy velocity input more than velocity input. In that case, we propose *Fuzzy Embedding* to convert velocity input into fuzzy velocity input.

As demonstrated in the shape-wrinkle synthesis, fuzzy velocity, providing more valuable information than fuzzy velocity in determining the temporal dependency between wrinkle change and shape change, can enhance the degree of the realism effectively. At this stage, we directly add previous velocity input series into current input to memorize short-term temporal dependency, which only takes into account the fixed-sized temporal dependency. In future, we will attempt to construct recurrent network driven by velocity or fuzzy velocity so as to learn unfixed-sized temporal dependency.

## Acknowledgements

This work was supported by the National Natural Science Foundation of China (grant 60303018, 60575032, 60473043), Shanghai Qingmingxing Project (05QMX 1447) and Beijing Multimedia and Intelligent Software Technology Laboratory. Thanks to the reviewer's comments.

## References

1. Cyriaque Kouadio, Pierre Poulin, Pierre Lachapelle: Real-time facial animation based upon a bank of 3D facial expression. Proc. Computer Animation'98, pp:128-136, 1998
2. Moiza. G, Tal. A, Shimshoni. I, Barnett. D, Moses. Y: Reconstruction of movies of facial expressions. IEEE ICCV 2001, Page(s):8 - 15, 2001
3. F. Pighin, J. Hecker, D. Lischinski, R. Szeliski, D. H. Salesin: Synthesizing Realistic Facial Expressions from Photographs. SIGGRAPH 1998, pp. 75-84, 1998
4. Terzopoulos D, Waters K: Physically-Based Facial Modeling and Animation. Journal of Visualization and Computer Animation, Vol. 1, pp. 73-80, 1990
5. Goudeaux, K., Tsuhan Chen, Shyue-Wu Wang, Jen-Duo Liu: Principal component analysis for facial animation. Proc. ICASSP'01. Page(s):1501-1504 vol.3.2001
6. Wenhui Zhu, Yiqiang Chen, Yanfeng Sun, Baocai Yin and Dalong Jiang: SVM-Based Facial Texture Driving for Realistic Expression Synthesis. ICIG 2004. Dec.18-20,2004

7. Jordan, M.I.(1986).Serial order: A parallel distributed processing approach. Institute for Cognitive Science Report 8604.University of California,San Diego
8. Hochreiter, S., Schmidhuber, J. (1997): Long short-term memory. Neural Computation, 9(8), 1735–1780
9. Bengio, Y., Fransconi, P. (1995): An input-output HMM architecture. In J.D. Cowan, G. Tesauro and J. Alspector (Eds.), Advances in Neural Information Processing Systems 7 (pp.427- 434), Cambridge, MA: MIT Press
10. J.L. Elman: Finding structure in time. Cognitive Science, 14:179–211, 1990
11. Richard O. Duda, Peter E. Hart, David G. Stork: Pattern Classification (2nd Edition), John Wiley and Sons, 2001
12. Ming-Wei Chang, Chih-Jen Lin, Weng,R.C.-H: Analysis of switching dynamics with competing support vector machines. IEEE Transactions on Neural Networks, 15(3),720 - 727
13. J. Kohlmorgen, S. Lemm, G. Rätsch, and K.-R. Müller: Analysis of nonstationary time series by mixtures of self-organizing predictors. In Proceedings of IEEE Neural Networks for Signal Processing Workshop, pages 85-94, 2000
14. V. Vapnik: Statistical Learning Theory. John Wiley, 1998
15. E. Hüllermeier: Fuzzy methods in machine learning. and data mining: Status and prospects. Fuzzy Sets and Systems, vol. 156, pp.387-406, 2005
16. A. Joshi et.al: On Neurobiological, Neurofuzzy, Machine Learning and Pattern Recognition Techniques. IEEE Transaction on Neural Networks, vol. 8, no.1, pp. 18-31, Jan.1997
17. Ian Cloete, Jacobus van Zyl: A machine learning framework for fuzzy set covering algorithms. SMC(4):3199-3203,2004

# Re-ordering Methods in Adaptive Rank-Based Re-indexing Scheme

Kang Soo You<sup>1</sup>, Jae Ho Choi<sup>2</sup>, and Hoon Sung Kwak<sup>2</sup>

<sup>1</sup> School of Liberal Arts, Jeonju University, Jeonju, 561-756, Korea  
you.kangsoo@gmail.com

<sup>2</sup> Department of Computer Engineering, Chonbuk National University,  
Jeonju, 561-756, Korea  
{wave, hskwak}@chonbuk.ac.kr

**Abstract.** We apply the adaptive ranking methods in preprocessing for lossless image compression. We suggest four phase methods to determine priority in same co-occurrence frequency on a row as rank-based re-indexing of index image. Firstly, the element located at first has the priority rank in the co-occurrence frequency matrix. Secondly, the element located at the main diagonal axis has the priority rank. Thirdly, considering all co-occurrence counts in a row and a weighted function according to distance among elements, the nearest element to the highest one has the priority rank. Finally, this method compromises the third method with the second method to decide the priority. As the result of the experiment, the proposed methods showed efficiency on compression ratio than conventional re-indexing algorithms.

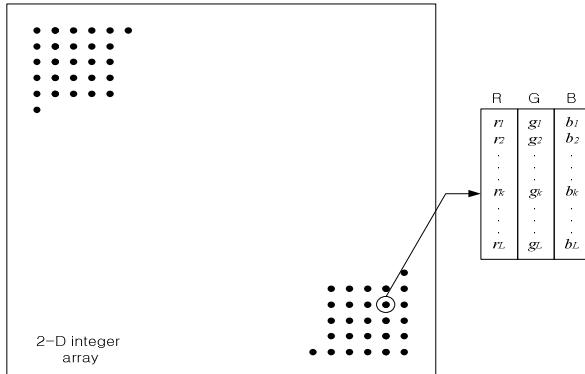
**Keywords:** Re-ordering, Re-indexing, Lossless image compression.

## 1 Introduction

The lossless image compression through the re-indexing of index images is efficient [1]. Such index images have been diffused widely through recent internet. There are some examples as many famous corporation logos, computer graphic images and medical images. Specially, medical treatment image, the art works image demand to be stored without any loss during its compression. The image which satisfies this condition at internet is used in general GIF file form.

Figure 1 shows the element of index image is composed with constant matrix and color map [2]. An index image stores colors as an array of indices into the colrmap. The color map should be an  $L$  row by 3 column matrix. The columns contain red, green, and blue intensities respectively. All entries should be between 0 and 1 inclusive.

From earlier 2000's, the particular scheme for re-indexing color mapped images was proposed by Zeng *et al.* [3]. It is based on increasing the efficiency of lossless compression. Its goal is to reduce the overall difference of index values of adjacent pixels. In addition to, Pinho *et al.* [4] proposed an algorithm which improves Zeng's algorithm by the method of more reducing the overall difference of index values of adjacent pixels.



**Fig. 1.** Element of 2-D index image : constant instant and color map

In this paper, we propose a new adaptive method which used RIAC(ranked re-indexing with arithmetic coding) [5] to reduce bits per rate, then we propose the methods which determine the co-occurrence frequency rank to have same count on a co-occurrence frequency matrix of each row consisted of the pixel-pair those are the neighborhood each other.

The proposed methods deal with index color images. Therefore, it has an advantage of processing index color image without any converting into intensity-based image. Also it can be implemented simply so time complexity is  $O(M^2 \log M)$ .

This paper is organized as follows. In section 2, we give a brief overview of conventional re-indexing scheme. And in section 3, we show how to convert an original image into rank-based image adaptively. Then, we present several decision priority ranking co-occurrence counts. In section 4, we present some result from our simulation comparing Zeng's and Pinho's algorithm to verify the proposed methods. Finally we summarize our results in section 5 and conclude this paper.

## 2 Re-indexing Scheme

### 2.1 Zeng's Algorithm

Zeng's algorithm, first, calculates the cross-counts  $C(S_i, S_j)$  for each pair of symbol  $S_i$  and  $S_j$  based on the initial index image and the cumulative cross-counts as equation (1) for each symbol  $S_i$ .

$$C_i = \sum_{j=0, j \neq i}^{M-1} C(S_i, S_j) \quad (1)$$

And then, the algorithm chooses the symbol  $S_{\max}$  that has the largest cumulative cross-counts, denotes it as  $L_0$ , puts  $L_0$  in a symbol pool  $P$  that will consist of

spatially ordered symbols , i.e.,  $P = \{L_0\}$ . The size of  $P$  denotes as  $N$ , and set  $N = 1$ . A new unassigned symbol is entered in  $P$  only from either the left and or the right end.

In order to choose the most suitable symbol that will put the current end position of the pool  $P$ , the maximal unassigned symbol is chosen. It is the reason that difference between neighbored indices in replaced with index image. A candidate symbol entering  $P$  is chosen by equation (2).

$$D_i = \sum_{j=0}^N w(N, j)C(S_i, L_j) \quad (2)$$

$$w(N, j) = \log_2 \left( 1 + \frac{1}{d(N, j)} \right) \quad (3)$$

Through the algorithm steps, assign integers  $0, 1, \dots, M - 1$  to the spatially ordered symbols in the pool  $P$  in left-to-right or right-to-left order. A re-indexed image can be obtained by replacing the initial index value  $i$  with the new index value assigned to  $S_i$ .

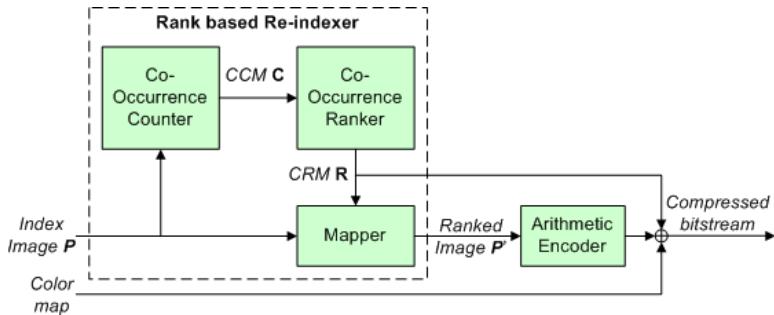
## 2.2 Pinho's Algorithm

Pinho *et al.* [4] proposed an improved Zeng's algorithm to have more entropy efficiency. In order to reduce the overall difference of index values of adjacent pixels, this algorithm modifies Zeng's algorithm as equation (4).

$$\begin{aligned} D_{L,R}(S_i, N) &= \sum_{j=0}^{N-1} \log_2 P(j+1)C(S_i, L_j) \\ &\quad - \sum_{j=0}^{N-1} \log_2 P(j+2)C(S_i, L_j) \\ &= \sum_{j=0}^{N-1} \log_2 \frac{P(j+1)}{P(j+2)} C(S_i, L_j) \end{aligned} \quad (4)$$

## 2.3 RIAC Algorithm

It is started by counting co-occurrence frequency in initial index image. Using the relationship between two adjacent pixels, a co-occurrence count matrix is made. The size of matrix is a max number used index of image. Counted all pixels' indices, co-occurrence count rank matrix is made. The result of rank image is generated by referencing both the rank matrix and initial image. The algorithm has enhanced arithmetic coding because its symbols have skewed distributions or little variance. RIAC algorithm is displayed as figure 2.

**Fig. 2.** Encoder of RIAC

### 3 The Proposed Algorithm

#### 3.1 Adaptive RIAC Algorithm

Both Zeng's and Pinho's algorithms mentioned earlier are to reduce difference in adjacent indices, after counting co-occurrence frequency of indices pairs. The best re-indexing algorithm can be obtained by heuristic strategy. We cannot expect better compression efficiency with those algorithms, because re-indexing images' and initial images' indices have same number of index.

In this paper, therefore, we propose an algorithm, which is an adaptive improving method of RIAC algorithm, to have better compression efficiency in entropy coding. If co-occurrence count matrix has same count values, we also suggest several deciding priority methods with assuming that there is no same rank about same counting value. The proposed adaptive RIAC algorithm is follows:

1. 1-D permutation  $P = (P_0, P_1, \dots, P_m)$  indicates index values of each pixel. Here  $P_0$  is a virtual pixel value for processing the first pixel in an initial image and the lowest index value  $I_1$ . Each entity  $P_i$  ( $1 \leq i < n \times m$ ) represents a  $i$ -th index value scanned row by row.
2. In the first step,  $C^1$ 's entities of Co-occurrence Count Matrix are initialized with '0'. The index pair  $P^i$  is defined as  $P_j \{P_j | P_j \in P, j \leq i\}$  from  $P$ .  $R^1$ 's entities of Co-occurrence Rank Matrix are updated by referencing  $C^1$ . And then the first rank is obtained by referencing  $P^1 = (P_0, P_1)$  and  $R^1$ . Now  $C^2$  of Co-occurrence Count Matrix updates a count number of co-occurrence on  $P^1 = (P_0, P_1)$ . Also  $R^2$  of Co-occurrence Rank Matrix is updated by referencing  $C^2$ .
3. In the second step, the second rank is obtained by  $(P_1, P_2)$ . As the first step,  $C^2$  updates a count number of co-occurrence on  $(P_1, P_2)$ . If  $P^1 = P^2$ ,  $C^3$  accumulates these count values. Here  $C$  and  $R$  represent Co-occurrence Count

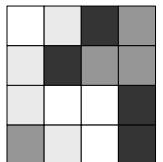
Matrix and Co-occurrence Rank Matrix respectively. An entity,  $c^i(j, k)$  of  $c^i$  satisfies as equation (5).

$$c^i(j, k) = \{c^i(j, k) \mid 0 \leq k \leq M\}, \quad (5)$$

$$c^i(j, k) \in C^i$$

The rank image from original image is obtained, as the last index pair,  $P^{n \times m} = (P_{(n-1) \times (m-1)}, P_{n \times m})$  is processed.

Figure 3(a) displays a sample image with four colors,  $M=4$ . Figure 3(b) and 3(c) display the index image from figure 3(a) and the rank image converted by the proposed adaptive method with 3(b). Figure 4 and figure 5 display the first and second step in the proposed algorithm.



(a) sample image

3	2	0	1
2	0	1	1
2	3	3	0
1	2	3	0

(b) index image

4	3	1	3
3	1	1	3
2	4	4	3
1	1	2	1

(c) rank image

**Fig. 3.** Sample index image with 4 colors

$I_j$	0	1	2	3
0	0	0	0	0
1	0	0	0	0
2	0	0	0	0
3	0	0	0	0

(a) CCM

$I_j$	0	1	2	3
0	1	2	3	4
1	1	2	3	4
2	1	2	3	4
3	1	2	3	4

(b) CRM

4			

(c) the first rank

**Fig. 4.** The related matrixes for generation the first pixel in the rank image from Fig. 1

$I_j$	0	1	2	3
0	0	0	0	1
1	0	0	0	0
2	0	0	0	0
3	0	0	0	0

(a) CCM

$I_j$	0	1	2	3
0	2	3	4	1
1	1	2	3	4
2	1	2	3	4
3	1	2	3	4

(b) CRM

4	3		

(c) the second rank

**Fig. 5.** The related matrixes for generation the second pixel in the rank image from Fig. 1

### 3.2 Deciding Priority of Rank

In order to decide the ranking from co-occurrence count matrix CCM, we give high rank to the first coming index like original index matrix. Then, there are four distinct phases to get co-occurrence count rank matrix CRM. These four phases are considered to set the collision when there is same rank value in the matrix of neighboring rank index. The four methods at each four phase are following. Figure 6 shows the results of distance measuring in each method.

#### **Method 1**

At the figure 4(a) and figure 5(a), the method 1 lines up the elements on each row by the descending series order. This phase, the case where the values that are identical, in order to decide the ranking of the elements, the first coming element has the priority like this figure 4(b) and figure 5(b).

#### **Method 2**

Similarly to method 1, method 2 gives high priority to the nearest element from main diagonal axis of unit matrix. This method used the characteristics of square matrix which has  $256 \times 256$  cells of which the 256 color images that are used in experiments. If the index images' color number is  $M$ , then the rank matrix size should be  $M \times M$ . Consequently the co-occurrence frequency value of element which exists in the location that corresponds the Kronecker's delta  $\delta^{i,j} (i = j)$  has the highest and the nearer the higher. The formula which calculates the distance of element valued '1' from the unit matrix, where in this formula,  $M$  represents the number of color, then the max value is 256.

$$\hat{C}_{ij} = C_{ij} + \frac{M - 1 - |i - j|}{M} \quad (6)$$

#### **Method 3**

Let the highest co-occurrence frequency number be  $j$ , and  $j+k$  be the second, then the rank of  $j$  is 1, and  $j+k$  is 2. If  $j-1$  or  $j+1$  are identical, then the overall rank is  $j[1], j-1[2]$  or  $j+1[2], j+k-1[3]$  or  $j+k+1[3]$ , where the value inside the '[]' denotes a decided rank and  $j$  or  $k$  are random column, then  $j, k, M$  and  $N$  satisfy  $\{j, k | 0 \leq j < M-1, j \in N\}$ . Method 3 can be represented as following.

$$\begin{aligned} \hat{C}_{ij} &= C_{ij} + \frac{\sum_{n=1}^M C_{in} \omega(j-n)}{\sum_{l=1}^N C_{lj}} \\ \omega(n) &= \left(1 - \frac{|n|}{M-1}\right)^2 \end{aligned} \quad (7)$$

#### **Method 4**

Method 4 is the trade off between method 2 and method 3. In equation (7), the method 2 is used when the denominator grows greater than the threshold  $T$ . Also in

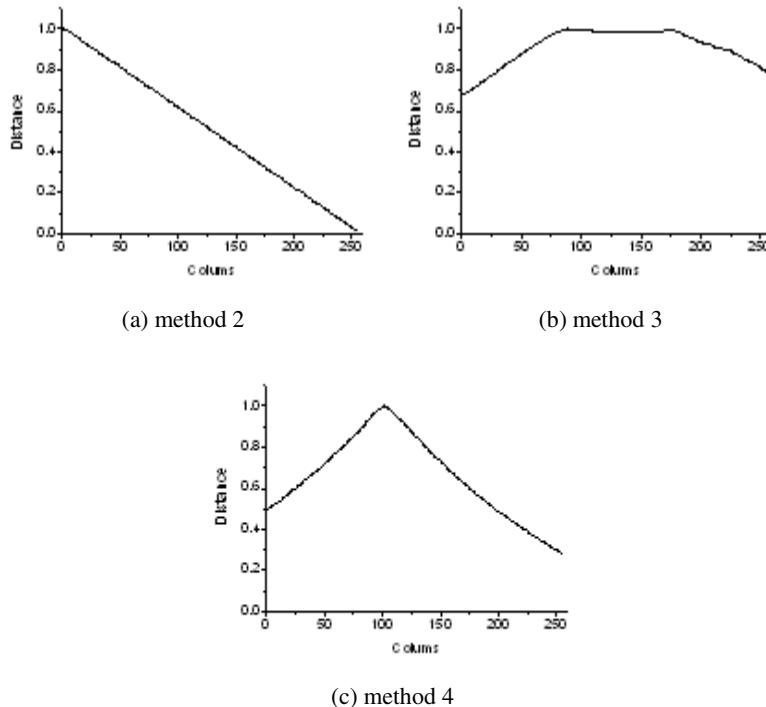
equation (7), the method 3 is used when the denominator decreases less than or equal to the threshold  $T$ . The denominator in equation (7) is the measured color co-occurrence frequency value.

$$\sum_{l=1}^N C_{lj} \begin{cases} > T, & \text{Method 2} \\ \leq T, & \text{Method 3} \end{cases} \quad (8)$$

## 4 Performance Evaluation

The images used for a simulation are two sets, all based on eight bits palette. In particular one set collects photo images of real life scenes and one set collects synthetic graphic images. All images have only limited number of colors (12~256). Also these images have used for test of conventional re-indexing algorithm [6].

Simulation was performed to compare the performance of the proposed algorithm with that of Zeng's and Pinho's algorithm. Compression Ratio ( $CR$ ) is used to calculate bits per pixel (bpp) to evaluate the compression performance. Table 1 and 2



**Fig. 6.** The distance measuring in each method

display the simulation results with synthetic images and photo images of real life scenes, respectively.

As demonstrated in table 1 and table 2, the proposed algorithm needs lower bit rates than other coding schemes. Here, ARIAC-1, ARIAC-2, ARIAC-3 and ARIAC-4 means adaptive RIAC algorithm with each the way deciding priority, as mentioned earlier. Figure 7 shows the histograms of initial and rank image in Clegg and Lena image respectively. Especially, two images' histogram is displayed in the top 10 bins contained almost pixels. After all, we can expect the higher efficiency in the compression ratio because of more data redundancy.

**Table 1.** The results of simulation with synthetic images

Images	Zeng	Pinho	The Proposed Method				unit : bpp
			ARIAC-1	ARIAC-2	ARIAC-3	ARIAC-4	
clegg	5.863	5.456	4.175	4.159	4.166	4.161	
cwheel	3.058	2.878	2.857	2.843	2.845	2.845	
descent	2.943	2.854	2.816	2.678	2.758	2.755	
fractal	6.193	5.828	5.268	5.236	5.245	5.235	
frymire	3.619	3.376	2.414	2.408	2.409	2.409	
party8	0.318	0.318	0.289	0.287	0.287	0.287	
serrano	3.393	3.273	2.379	2.369	2.373	2.373	
yahoo	1.798	1.789	1.816	1.753	1.761	1.761	
Average	3.398	3.222	2.752	2.728	2.731	2.728	

**Table 2.** The results of simulation with real images

Images	Zeng	Pinho	The Proposed Methods				unit : bpp
			ARIAC-1	ARIAC-2	ARIAC-3	ARIAC-4	
airplane	5.056	4.445	4.282	4.232	4.234	4.223	
anemone	5.806	4.966	4.326	4.296	4.301	4.296	
baboon	7.097	6.496	6.101	6.058	6.075	6.052	
boat	6.048	5.823	5.321	5.273	5.278	5.269	
house	5.180	4.854	4.801	4.716	4.706	4.704	
lena	5.710	5.049	4.055	4.036	4.042	4.041	
monarch	4.325	3.917	3.233	3.217	3.222	3.221	
peppers	5.544	5.019	4.016	3.989	3.995	3.994	
Average	5.596	5.071	4.517	4.477	4.482	4.475	

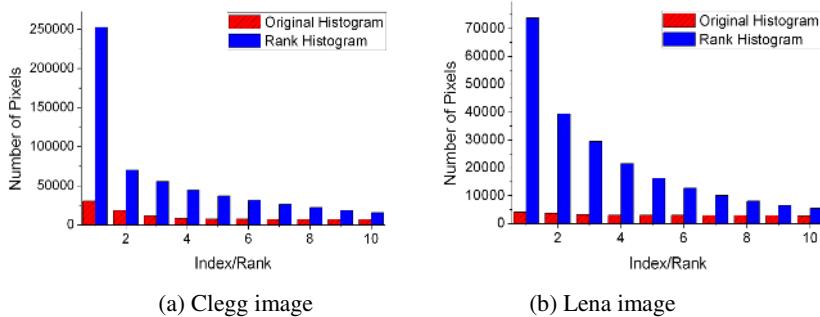


Fig. 7. The histogram of top 10 bins in Clegg and Lena image

## 5 Conclusions

This paper used the idea that no lost of quality is co-occurred while re-indexing the palette information of an image. After measuring the co-occurrence frequency between the two pixels' index and detecting the ranking against all contiguous pixel pairs, we proposed the algorithm which changes an initially presented the index images with the rank images. The rank image which is exchanged like this, applied an arithmetic coding.

Specially, in order to decide the ranking which followed by the co-occurrence frequency the case of four branches are defined and applied in ARIAC algorithm. This method consider that the number of same rank of index surrounding a pixel pair will be able to increase frequently while the quality of the image is not changed. The algorithm proposed from this study, is compared with the Zeng's algorithm and the Pinho's algorithm.

From the proposed ARIAC algorithm which we got the result evenly up to 0.670 bpp and 0.494 bpp, rate saving than conventional algorithms. Also from actual collar image of index mode efficiency showed as of 1.121 bpp and 0.596 bpp evenly.

Consequently, in the medical image field which does not permit the loss of image and the satellite image field which is used from military remote sensing application, the proposed algorithm will be able to contribute on lossless compression.

## References

1. Battiatto, S. Gallo, G. Impoco G. and Stanco, F.: An Efficient Re-Indexing Algorithm for Color-Mapped Images, IEEE Transactions on Image Processing, Vol. 13 No. 11 (2001) 104–108
2. Li, Z.N. Drew, M.S.: Fundamentals of Multimedia, New Jersey: Prentice Hall (2004)
3. Zeng, W. Li, J. and Lei, S.: An Efficient Color Re-Indexing Scheme for Palette-based Compression, Proc. IEEE int. Image Processing, ICIP-2000., Vol. III (2000) 476–479

4. Pinho, A.J. and Neves, A.J.R.: A Survey on Palette Reordering Methods for Improving the Compression of Color-Indexed Images, *IEEE Transactions on Image Processing*, Vol. 13, No. 11 (2004) 1411–1418
5. You, K.S. Lee, H.J. Jang, E.S. Kwak, H.S.: An Efficient Lossless Compression Algorithm using Arithmetic Coding for Indexed Color Image, *KICS on Communication Theory, Signal Processing, Information Security*, Vol. 30 No. 1C (2005) 35–43
6. Pinho, A.J.: <ftp://ftp.ieeta.pt/%7Eap/images>

# Use of Nested K-Means for Robust Head Location in Visual Surveillance System

Hyun Jea Joo, Bong Won Jang, Sedai Suman, and Phill Kyu Rhee

Dept. of Computer Science & Engineering

Inha University, Yong-Hyun Dong

Incheon, South Korea

{hj0235, bictory7, suman}@im.inha.ac.kr, pkrhee@inha.ac.kr

**Abstract.** This paper presents a head detection method for frontal face detection. We use motion segmentation algorithm that makes use of differencing to detect moving people's head. The novelty of this paper comes from adaptive frame differencing, detecting edge lines and restoration, finding the head area and cutting the head candidate. Moreover, we adopt nested K-means algorithm for finding head regions. Our system applies the statistical modeling of face and non – face classes and classifies multiple frontal face images with the Bayesian Discriminating Features (BDF) method to verify. Finally experimental results (using capture diverse image sources for 13 frames per second during 20 seconds and having 260 images per person) shows the feasibility of the differencing based head and Nested K-means Detection method.

## 1 Introduction

Many visual surveillance systems have adopted camera capture system to detect a lot of typical objects or analyze their motion and patterns. As huge size images, we need only objects active candidate regions. Because their big images bring about long time to calculate and detect objects. Therefore, it is the key to find segmentation domains for using real time applications. Because of them, many systems adopt head detection and tracking methods. [1][2][3][4][9][10][11] And they adopt color – based methods to find head regions. [1][3][4][9] However many people have different face color and illumination in various environment. Therefore many systems do not define all face color values. They have solved the restricted condition. And they select the shape modeling like as ellipse. [12][14] But, these applications have disadvantage to solve noise effect. When the noise is contained the original image, the ellipse model turn to huge space region and image distortion. For overcoming this defect, some methods have used motion segmentation algorithm. Motion segmentation in visual surveillance system falls into three divisions. [7]

1) Usage of environment map. These algorithms are very universal. First, they make statistical background. Then, they perform subtraction of current image frame from environment map. [9][10] .This method is simple and gives good result in restricted environment. But it is very weak when the light circumstance is changed like as the

direct rays of the sun. So the first consideration in this case is to make accurate environment which seems difficult to achieve.

2) Frame differencing. Many approaches adopt difference between two or three frames in image sequence. However, an improved performance is achieved by three frame differencing. Frame differencing is very robust method to continuing environmental changes. [11] Because of the differencing result, it is fragile to restore the original edges and to extract the active region. Therefore, the advantages of them are based on making activate candidate regions.

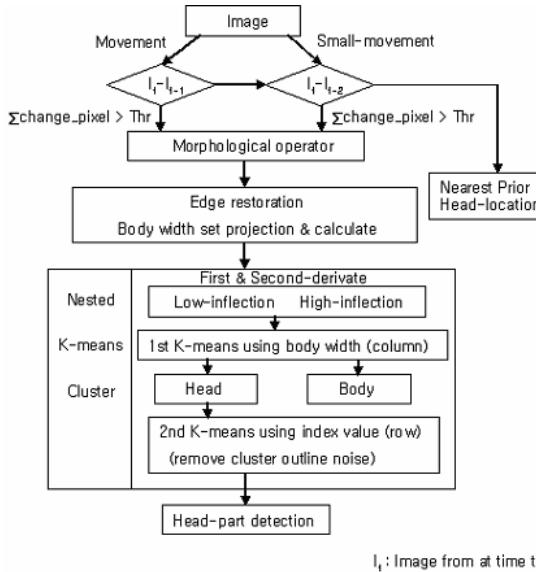
3) Optical flow. They select flow vectors of moving objects in an image sequence. The results apply for gait and activity analysis. [12] But they have defects in complex and sensitive noise effects and do not perform in real time video.

Because of unrestricted surrounding changes in real time, frame differencing method is used to improve accuracy and to minimize multi-resolution search so high head detection rates employing minimum multi resolution searches and reducing overall computational time is aimed to be achieved here. Edge detection and restoration challenge is overcome by using image evaluation, preprocessing, interpolation and gradient operator. Then, active head regions are trimmed resulting in reduced search spaces in shorter time. Although the segmentation is not always correct, result of head detection time is found to be robust in real time surveillance. We select the threshold based on the expected distance with manual input between the people and camera by interpolation method. Moreover, only head detection has been taken into account for this research. So we perform head detection using two clusters K-means algorithm followed by head and body histogram analysis. To detect heads of people, we adopt 13 frames per second frame rate for single person in indoor scene. The camera is fixed 1.6 meter above the ground. The people can move over 25 square meters area in front of camera with their face pointing towards it. Our approaches are divided into several steps. Section 2 presents system architecture. Section 3 deals with image evaluation, image preprocessing, edge detection and restoration. Section 4 presents head detection and face detection using Bayesian method. Section 5 brings up experiments. Finally, sections 6 describe the conclusion and future extensions.

## 2 Real Time Head Detection System Architecture

Our purposed system architecture is shown in Fig. 1. Input images are of  $320 \times 240$  sizes. Then the difference between the current image (N) frame and previous frame (N-1) is calculated and evaluated. The difference is then compared with certain threshold. The system proceeds towards edge detection only when the difference is significant enough to find the body segment. But when the result is not changed i.e. difference is less than the threshold value (person is not moving), current frame (N) is compared with the second previous (N-2) and difference is evaluated. Therefore the maximum difference is measured between two frames by comparing three image frames. If again the difference is not significant then previously segmented head part is used for face detection .Otherwise other steps are proceeded out which results in new head segmentation. The steps include edge detection or restoration. Here accurate edge line across the body is found by performing gradient operators and interpolation, detail of which is discussed in section 3. Next, the projection of horizontal distance

between the edges is calculated for each row. Local minima and maxima point is calculated on the projection vector to find the head and body location respectively. To overcome the error due to noise, the projection vector is clustered around the two regions (one for head and one for body) using K-means algorithm. After the head detection region is extracted from the image, Bayesian Discriminating feature method is used to search the target's face in the region to verify.



**Fig. 1.** Architecture of the proposed methodology

### 3 Movement Detection, Edge Detection and Restoration

#### 3.1 Movement Detection

Our proposed method adopts image evaluation for movement detection, i.e. finding people that are moving or not moving. We think people can move over 25 square meters area. When the people get far from camera, human's head is very small for example 25 pixels  $\times$  25 pixels. However there are many head size in end point like as hair style, man or woman and so on. Since we use Bayesian discrimination method for face detection and it uses minimum of 16 $\times$ 16 pixel window for statistical analysis. Threshold is calculated assuming that the person should at least move his head for the noticeable change in the successive image frames. The border line of 16 $\times$ 16 window contains 16 $\times$ 4 pixels and the approach is to find the accurate edge of human. Therefore the threshold is found to be  $\Delta\text{Thr}_{\text{total}} = 16 \times 3$  (i.e. discarding the lower edge of window)

$$\text{Comp}(x,y) = \begin{cases} 255 & \text{Set}(x, y) < \Delta\text{Thr}_{in} \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where

$$\text{Set}(x,y) = \begin{cases} N(x,y) - N_{-1}(x,y) & \Delta\text{Sum}_{\text{change}} \geq \Delta\text{Thr}_{\text{total}} \\ N(x,y) - N_{-2}(x,y) & \text{otherwise} \end{cases}$$

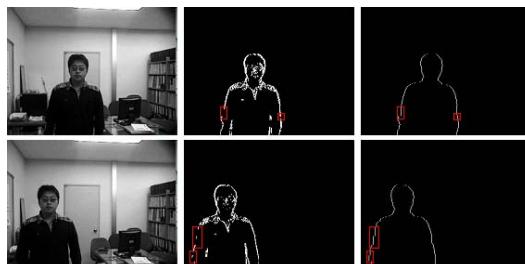
$$\Delta\text{Sum}_{\text{change}} = \frac{1}{255 * \text{Total}} \sum_{n=1}^{\text{total}} \text{Comp}(x,y)$$

$\text{Comp}(x, y)$  is the thresholding operator, and thresholds the difference image  $\text{Set}(x, y)$  into only two extreme gray color values (255 and 0). Then, opening morphological operator is used to reduce the noise effect due to thresholding.

$N(x, y)$  is the image sequence indicating the  $N_{th}$  intensity value of  $(x, y)$ . These values are resolved into white and black for finding edge line. However when  $\Delta\text{Sum}_{\text{change}}$  value is less than  $\Delta\text{Thr}_{\text{total}}$  current frame is regarded as non-moving. In this case the previous nearest head segmentation image is used as active region.

### 3.2 Edge Restoration

Edge is an important issue to detect a particular shape of an object. Human is symmetric shaped especially for frontal view. As frame differencing creates a lot of noise and there is no predefined way to detect noise. To overcome this, edge restoration is employed. After preprocessing and evaluation ,the edge image is searched for the left and right row point to find the first edge pixel .The column points for right and left edge is stored as  $\text{Left}_{\text{first}}(x,y)$  and  $\text{Right}_{\text{first}}(x,y)$  respectively. The spikes of the edges due to sudden illumination change are removed by comparing with prior column values and restoring.



**Fig. 2.** Original images (left), differencing images and preprocessed images (center), edge restoration images (right)

$$E_{\text{left}}(x_i, y) = \begin{cases} (\text{Left}_{\text{first}}(x_i, y) = \text{Left}_{\text{first}}(x_{i-1}, y)) & \text{Dev}(x, y), J_n(x, y) \geq \text{Thr}_{\text{out}} \\ (\text{Left}_{\text{first}}(x_i, y)) & \text{otherwise} \end{cases} \quad (2)$$

where

$$\text{Dev}_{\text{left}}(x, y) = (\text{Left}_{\text{first}}(x_i, y) - \text{Left}_{\text{first}}(x_{i-1}, y))$$

$E_{\text{right}}(x_i, y)$ ,  $\text{Dev}_{\text{right}}(x, y)$  are the same ways.

$$\text{Center}_{n-1}(x, y) = \frac{1}{n-1} \sum_{i=1}^{n-1} \frac{1}{2} (\text{Left}_{\text{first}}(x_i, y_i) + \text{Right}_{\text{first}}(x_i, y_i)) \quad (3)$$

$$J_n(x,y) = \frac{1}{2} (Left_{first}(x_n, y_n) + Right_{first}(x_n, y_n)) - Center_{n-1}(x,y)$$

This is done by checking two states (vertical and horizontal) of  $N_{th}$  column values. At first, the differencing between current row and prior row's column values is calculated. This value is based on gradient differencing. Secondly the prior vertical value is checked. Although the people are restricted to move within a square, the center vertical values hardly change. It is very easy to find edge line of symmetrical objects like people, ball etc because if edge of single side can be found then it is always possible to predict the edge of other side of object.

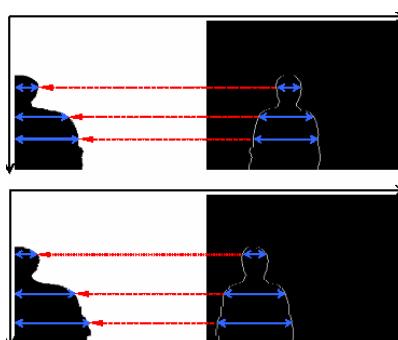
So if  $Dev_{left}(x, y)$  and  $J_n(x, y)$  are less than threshold  $Thr_{out}$  then the current edge is regarded as correct edge of nearest people. Otherwise the edge is considered as noise lines and hence it is replaced with the prior column values of the particular row. Some successive points for both columns are considered determining the presence or absence of noise. Noise is detected in two ways: distance and median. For distance method, if there is abrupt change between successive points, noise is detected; otherwise, noise free as shown in equation (2). For median method, the center point between left and right position is determined. If this center point much differs from threshold, we consider this as noise as shown in equation (3). If noise is detected noise-removing technique is applied.

#### 4 Head Region Segmentation Using Nested K-Means Algorithm

To reduce the search space for face detection, first active region of head is found by the method of head segmentation using vertical projection of edge contours and K-means algorithm. Then face is detected using Bayesian discriminant analysis method by searching the active region.

Here edge contours of the person are extracted by the method described in section 3.2. Then Body width set projection of the edge contours is calculated as shown in figure 3. Then local minima and maxima are calculated on the projection vector in order to find the expected location of head and body respectively.

$$RP(x_i) = Dev_{right}(y) - Dev_{left}(y) \quad (4)$$



**Fig. 3.** Edge image (left), Body width set projection (right)

Then first and second derivative of projection vector is calculated to find local minima and maxima. The definitions about second derivative states that; If the first derivative  $RP'(x_i)$  is positive (+), then the function  $RP(x_i)$  is increasing ( $\uparrow$ ) and  $RP'(x_i)$  is negative (-), then the function  $RP(x_i)$  is decreasing ( $\downarrow$ ). Also, If the second derivative  $RP''(x_i)$  is positive (+), then the function  $RP(x_i)$  is concave up ( $\cup$ ) and  $RP''(x_i)$  is negative (-), then the function  $RP(x_i)$  is concave down ( $\cap$ ). It is assumed that  $y=RP(x)$  is a twice-differentiable function with  $RP''(c)=0$ .

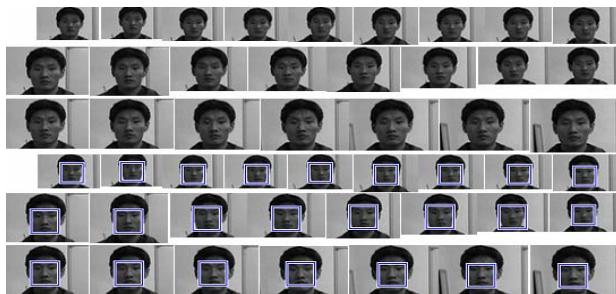
$$(5) \quad \begin{aligned} f''(x) &: y'', \frac{d^2y}{dx^2}, \frac{d^2}{dx^2}f(x) \\ RP'(x_i) &= \frac{f(x_i+h) - f(x_i)}{h} \\ RP''(x_i) &= \frac{d^2y}{dx^2} = \frac{d}{dx}\left(\frac{dy}{dx}\right) = \lim_{h \rightarrow 0} \frac{f'(x_i+h) - f'(x_i)}{h} \end{aligned}$$

Therefore if  $RP''(d) < 0$  then  $RP(x_i)$  has a relative maximum value at  $x=d$ . It means body center ( $\text{Max}_d(y)$ ) is in body cluster. In same ways, if  $RP''(c) > 0$  then  $RP(x_i)$  has a relative minimum value at  $x=c$ . So it is near to head center ( $\text{Min}_c(y)$ ) and in head cluster, too.  $\text{Max}_d(y)$ ,  $\text{Min}_c(y)$  points, that is the center of clusters is used to calculate the similarity:

$$(6) \quad \begin{aligned} D_b &= \| \text{Max}_d(y) - RP(x_i) \|^2 \\ D_h &= \| \text{Min}_c(y) - RP(x_i) \|^2 \\ G(x, y) &= \begin{cases} 0 & D_b \geq D_h \text{ Head part} \\ 1 & \text{otherwise Body part} \end{cases} \\ \mu_{head} &= \frac{\sum x}{\text{count}(x)} \quad \text{forall } x : G(x, y) = 0 \\ \mu_{body} &= \frac{\sum x}{\text{count}(x)} \quad \text{forall } x : G(x, y) = 1 \\ d_b &= \| x - \mu_{body} \|^2 \\ d_h &= \| x - \mu_{head} \|^2 \\ g(x, y) &= \begin{cases} 0 & d_b \leq d_h \text{ Head part} \\ 1 & \text{otherwise Body part} \end{cases} \end{aligned}$$

Nested k-means algorithm is implemented for the purpose of head and body segmentation. Clustering is done in two direction of projection vector. At first local minima and maxima is found in the projection vector  $RP(x)$ . The operator is defined (5) Then the vector is clustered to search head and body location using the  $y$  value using initial centroid as calculated minima and maxima. (6) The distance between the head cluster's center and body cluster center point is calculated for every entry in projection vector. If distance to the head cluster is greater than that of body cluster then the point is shifted to body cluster. The first stage of clustering separates head and body parts but error due to noise still exists. That is there are some of the body parts which are still-misclustered as head part. To reduce this noise, second level of clustering is needed .Second level of clustering is done on  $x$  direction of  $RP(x)$ . i.e. on the index value of the projection vector taking centroid result of first level clustering

as initial centroid. Here also the distance between the head cluster's center and body cluster center point is calculated for every entry in projection vector. If distance to the head cluster is greater than that of body cluster then the point is shifted to body cluster. So in second stage of clustering any outline point(x, y) (related to body) which mis-clustered to head part is corrected. It is found that this method increases the detection rate, because second level of clustering corrects most of the noise occurred in first level of clustering. And Figure 4 shows the result images with face detection using Nested K-means algorithm.



**Fig. 4.** Result images with head detection

## 5 Experiments

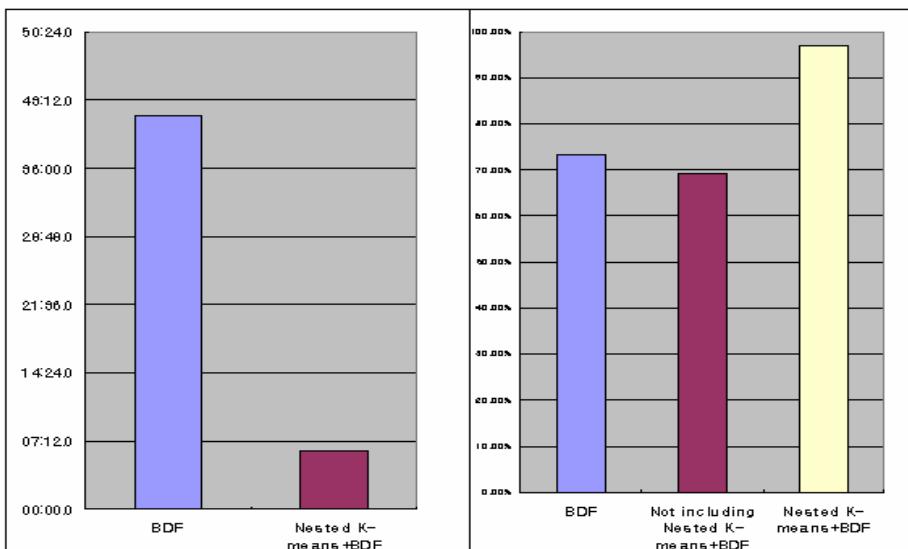
Our proposed method is tested using two series of experiments; first face detection without Nested K-means, second is face detection with head segmentation. A cheap LG USB camera is used as a video capturing device. The sequence frames of 320 x 240 pixels were acquired at 13 fps in 2.0GHz Pentium 4 PC computer and stored in the temporary folder in hard disk. Database is separated into 3 groups A, B and C. Group A contains normal people without any glasses and disguise form. Group B group includes people wearing glass. The last group C contains the people having beard and wearing glass in changing illumination condition. One group consists of 5 people dataset, and each person has twenty second movie clips at 13 fps. So each group has 1300 images sequence Table 1 shows the experimental results. After the head segmentation, the system finds Bayesian classifier to discriminate face and non face part of image. FERET database is used to learn the face Bayesian face model where as no-face Bayesian model is generated from the CMU database set.

Table 1 shows the face detection result for different scenario. Here it can see that face detection without segmentation gives high accuracy rate but it found to be time consuming. The third row of table 1 shows the result of our proposed method, which uses head segmentation and k-means for noise reduction.. The result of our proposed method is found to be superior in terms of accuracy 97% and very fast in terms of speed. The experiment was conducted, but that time not using k-means for noise reduction. The recognition rate in this case is found to be very poor. Hence the credit of higher recognition rate goes to the error reduction technique employed by k-means algorithm. And Table 1 shows the result of experiment that's shows the accuracy of

the head location. Same database and experiments were conducted. The table suggest that our method is almost perfect to find the head location with the accuracy higher than 97%. But, when we do not contain the nested K-means algorithm, the result is very low because our experiment in real-time visual surveillance system contains many noise effects like as changing illumination.

**Table 1.** Experiments about face detection rate about head region segmentation methods

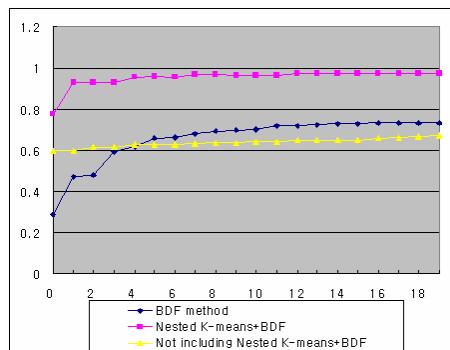
	Experiment division	Database A group	Database B group	Database C group	Average total Database
Non-including Nested k-means head detection +BDF	Face detection rate	73.29%	68.79%	65.87%	69.32%
		97.29%	97.34%	97.21%	97.28%
Head seg +BDF (proposed method)	Accurate Head Location rate	93.33%	92.14%	91.19%	92.22%
		98.07%	97.73%	99.23%	98.34%



**Fig. 5.** Time consumption (left), Average face detection rate (right)

Figure 5 shows the variation of time consumption and accuracy rates across various methods. It can be seen that the proposed method (Head segmentation +BDF (Bayesian Discriminant features [8]) method based face detection) have high detection rate with less response time.

Figure 6 displays the ROC curve, shows the relationship between false alarm and face detection rate.



**Fig. 6.** ROC curve

## 6 Conclusion

We have proposed real time head region detection based on difference of image frames from active camera. The main component of the system is the use of vertical projection of edge contours to find head location and Nested K-means algorithm for error minimization during differencing and threshold. Overall the system shows low computational cost and high detection rate which suits for the real time system. Accurate head detection rate of about 97% is achieved while minimizing the computational cost and time.

The future enhancement to the system can be to overcome the occlusion and to find automatic threshold for frames evaluation using multiple cameras.

## References

- [1] Maolin Chen and Seokcheol Kee, "Head tracking with shape modeling and detection"
- [2] G. L. Foresti, "A Real-Time System for Video Surveillance of Unattended Outdoor Environments", IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, VOL. 8, NO. 6, OCTOBER 1998
- [3] Tao Yang, Quan Pan, Jing Li, Yongmei Cheng, Chunhui Zhao, "Real-Time Head Tracking System With an Active Camera\*", Proceedings of the 5" World Congress on Intelligent Control and Automation, June 15-19, 2004, Hangzhou, PR China
- [4] Weimin Huang, Ruijing Luo, Haihong, Beng Hai Lee and Meneka Rajapakse " Real Time Head Tracking and Face and Eyes Detection, Proceedings of IEEE TENCON'02
- [5] Mi-Suen Lee," Detecting People in Cluttered Indoor Scenes"

- [6] How-Lung Eng, Junxian Wang, Alvin H. Kam and Wei-Yun Yau," A Bayesian framework for robust human detection and occlusion handling using human shape model"
- [7] Weiming Hu, Tieniu Tan, Fellow, IEEE, Liang Wang, and Steve Maybank "A Survey on Visual Surveillance of Object Motion and Behaviors", IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART C: APPLICATIONS AND REVIEWS, VOL. 34, NO. 3, AUGUST 2004
- [8] Chengjun Liu, Member, IEEE," A Bayesian Discriminating Features Method for Face Detection", IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 25, NO. 6, JUNE 2003
- [9] I. Haritaoglu, D. Harwood, and L. S. Davis, "W : Real-time surveillance of people and their activities," IEEE Trans. Pattern Anal. Machine Intell., vol. 22, pp. 809–830, Aug. 2000.
- [10] S. McKenna, S. Jabri, Z. Duric, A. Rosenfeld, and H. Wechsler, "Tracking groups of people," Comput. Vis. Image Understanding, vol. 80, no. 1, pp. 42–56, 2000.
- [11] A. J. Lipton, H. Fujiyoshi, and R. S. Patil, "Moving target classification and tracking from real-time video," in Proc. IEEE Workshop Applications of Computer Vision, 1998, pp. 8–14.
- [12] D. Meyer, J. Denzler, and H. Niemann, "Model based extraction of articulated objects in image sequences for gait analysis," in Proc. IEEE Int. Conf. Image Processing, 1998, pp. 78–81.
- [13] S. Mulassiotis and M. G. Strintzis, "REAL-TIME HEAD TRACKING AND 3D POSE ESTIMATION FROM RANGE DATA", 2003 IEEE
- [14] Harsh Nanda and Kikuo Fujimura, "A Robust Elliptical Head Tracker", Proceedings of the Sixth IEEE International Conference on Automatic Face and Gesture Recognition (FGR'04)

# Appearance Based Multiple Agent Tracking Under Complex Occlusions

Prithwijit Guha<sup>1</sup>, Amitabha Mukerjee<sup>2</sup>, and K.S. Venkatesh<sup>1</sup>

<sup>1</sup> Department of Electrical Engineering,  
Indian Institute of Technology, Kanpur,  
Kanpur - 208016, Uttar Pradesh  
[{pguha, venkats}@iitk.ac.in](mailto:{pguha, venkats}@iitk.ac.in)

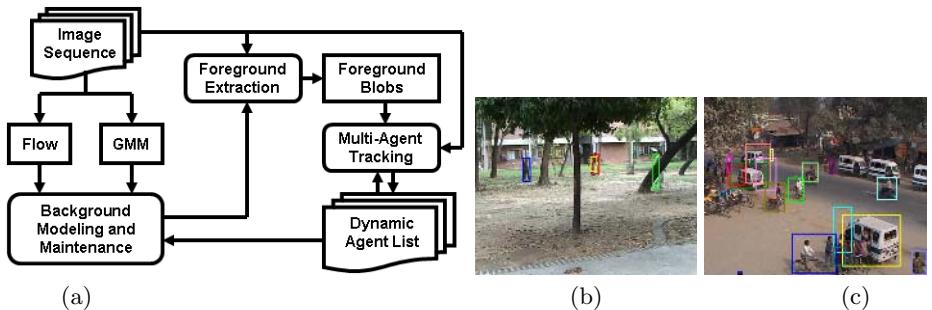
<sup>2</sup> Department of Computer Science & Engineering,  
Indian Institute of Technology, Kanpur,  
Kanpur - 208016, Uttar Pradesh  
[amit@cse.iitk.ac.in](mailto:amit@cse.iitk.ac.in)

**Abstract.** Agents entering the field of view can undergo two different forms of occlusions, either caused by crowding or due to obstructions by background objects at finite distances from the camera. This work aims at identifying the nature of occlusions encountered in multi-agent tracking by using a set of qualitative primitives derived on the basis of the Persistence Hypothesis - objects continue to exist even when hidden from view. We construct predicates describing a comprehensive set of possible occlusion primitives including entry/exit, partial or complete occlusions by background objects, crowding and algorithm failures resulting from track loss. Instantiation of these primitives followed by selective agent feature updates enables us to develop an effective scheme for tracking multiple agents in relatively unconstrained environments. The agents are primarily detected as foreground blobs and are characterized by their centroid trajectory and a non-parametric appearance model learned over the associated pixel co-ordinate and color space. The agents are tracked through a three stage process of motion based prediction, agent-blob association with occlusion primitive identification and appearance model aided agent localization for the occluded ones. The occluded agents are localized within associated foreground regions by a process of iterative foreground pixel assignment to agents followed by their centroid update. Satisfactory tracking performance is observed by employing the proposed algorithm on a traffic video sequence containing complex multi-agent interactions.

## 1 Introduction

An algorithm for tracking multiple agents in a (quasi) static monocular surveillance setup is reported. A naive approach to this problem deals with tracking blobs obtained from the process of background subtraction [1]. However, such blobs do not necessarily correspond to individual agents, as the agents can form a group and get detected as a single blob or an agent can be detected as multiple blobs due to occlusions. The W4 system [2] differentiates people from other

objects by shape and motion cues and tracks them under occlusions by constructing appearance models and detecting body parts. Several researchers [3,4] have employed particle filtering along with prior shape and motion models for multi-person tracking in cluttered scenes. Recently, Zhao et al. [5] proposed a Bayesian approach for tracking multiple persons under occlusions by computing MCMC based MAP estimates with prior information about camera model and human appearance along with a ground plane assumption. McKenna et al. [6], on the other hand, presents a color based tracking algorithm that performs in relatively unconstrained environments and works at three levels of abstraction, viz. regions, people and groups.



**Fig. 1.** Illustrating the proposed algorithm. (a) The functional block diagram of the algorithm; Cases of occlusions. (b) Partial occlusion: agent occluded by tree is visible as two fragmented blobs; in this state, the agent is recognized, but its visual characteristics are not updated. (c) Crowding: multiple agents merge to form a single blob.

This work extends our earlier approach to surveillance event primitive guided occlusion handling in multi-agent tracking [7]. Firstly, we propose to learn the background model with a single frame latency thereby identifying agents that have come to a stasis or have suddenly started moving. The temporal pixel intensity distribution is continuously learned as a mixture of Gaussians [8] and the moving pixels are delineated by motion detection using [9]. More so, we employ a higher layer tracking feedback to avoid learning the agent regions as background which have come to a stasis. Secondly, this work represents the agent appearance as a non-parametric model constructed from its associated pixel coordinates and color space. We detect a set of qualitative *occlusion primitives* by associating predicted agent regions with extracted foreground blobs. Thus, a given agent can have six possible states with respect to other agents and fixed objects: *isolation*, *partial occlusion* (several foreground objects in one agent region), *crowding* (several agents in the same foreground region), *disappearance* (complete occlusion or track loss), *entrance* (initial appearance of an agent) and *exit* (disappearing near the image boundaries) - these constitute the occlusion primitives. The occluded agents are further localized in these blobs through a process of iterative agent pixel assignment followed by pixel likelihood weighted centroid update. Our algorithm makes no constraining assumptions regarding

the agent shape and motion models, ground plane etc. Some salient strengths of the proposed scheme are the following.

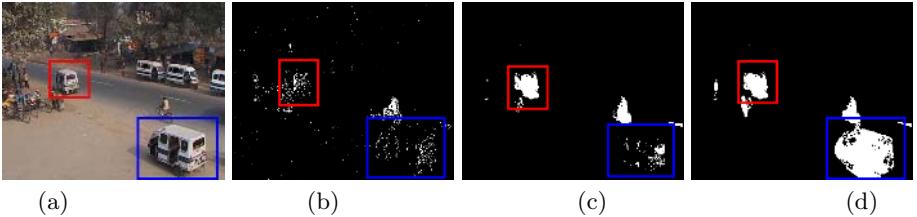
- (a) Ability to distinguish a variety of problematic situations.
- (b) Using the above information to decide on the advisability of feature updates.
- (c) Inherent ability of recognizing failure situations, should they occur.
- (d) Ability to automatically track restoration at a later time
- (e) No constraining assumptions related to agent shape and motion models, ground plane, etc.

The functional block diagram of the proposed algorithm along with a snapshot of the results for certain typical occlusion cases are shown in figure 1 for illustration purposes. The next section presents the underlying work on foreground extraction, and section 3 develops the algorithm for multi-agent tracking and subsequent occlusion primitive identification. The experimental results of multi-agent tracking are presented in Section 4 before concluding in section 5.

## 2 Foreground Extraction

Agents are identified as foreground regions based on one of two kinds of evidence: first, as regions of change with respect to a learned background model; and second, as regions exhibiting motion. Learning the background model in presence of agents is a challenging problem in itself. Several approaches have been proposed to incrementally learn the background scene model in the presence of agents. The most commonly adopted algorithms include the computation of median [2] or fitting (temporally evolving) Gaussian mixture models [10,8] on the temporal pixel color histogram of the image sequence. These approaches continuously learn the multi-modal mixture models with the assumption that the moving objects appear at a certain pixel only temporarily and the *true* background remains accessible to the system more frequently leading to higher weight of the corresponding mode. However, such an approach is prone to transient errors persisting over a number of frames (depending on the learning rate), resulting in two types of errors. First, if agents learned as part of the background suddenly start moving, ghosts and holes appear in the foreground segmentation. Second, when a moving agent comes to stasis, it is eventually learned as a part of the background, which may not be desirable in itself, and also in the transition period, objects interacting with it would not be identified. Both these problems are averted in the present approach by combining background-model and motion evidence, and updating based on tracking / previous motion-history feedback.

Generally, the background model  $\mathcal{B}_t$  at the  $t^{th}$  instant is selectively updated based on the classification results of the  $t^{th}$  frame  $\Omega_t$ . Classification based on  $\mathcal{B}_{t-1}$  first results in a set of foreground pixels  $\mathbf{F}_b(t) \subset \Omega_t$ . Next, an inter-frame motion estimation [9] is performed between  $\Omega_t$  and  $\Omega_{t+1}$  to delineate the set of moving foreground pixels  $\mathbf{F}_m(t) \subset \Omega_t$ . This results in single-frame latency that helps us in identifying the regions that suddenly start moving or come to a stop.



**Fig. 2.** Results of foreground detection. (a) The *tempo* in the red bounding box starts moving (Frame 1628), and the *tempo* highlighted by blue has already come to rest; Foreground extraction results (b) using only per pixel Gaussian mixture model with traditional exponential forgetting; (c) with motion detection only ( $\mathbf{F}_m$ ) and (d) combining motion evidence with tracking feedback ( $\mathbf{F}_t$ ).

Pixels identified as both foreground and moving are clearly identified as agent pixels. Among the mismatched pixels, moving pixels not identified as foreground, are denoted as  $\mathbf{F}_{hole}(t) = \mathbf{F}_m(t) - \mathbf{F}_b(t)$ . On the other hand, the set of non-moving pixels in  $\mathbf{F}_b(t)$ , is given by  $\mathbf{F}_{ghost}(t) = \mathbf{F}_b(t) - \mathbf{F}_m$  and is identified as a possible background candidate. However, these non-moving ghost pixels may contain actual agent regions which have not shown up in the optical flow, or where an agent has actually come to a stasis. Using information from the motion history and tracking (discussed in Section 3) we delineate the set of agents pixels that are known to have come to rest,  $\mathbf{F}_0(t) \subset \mathbf{F}_{ghost}(t)$ . The set of agent pixels that emerge from this analysis is defined as  $\mathbf{F}(t) = (\mathbf{F}_b(t) - \mathbf{F}_{ghost}(t)) \cup \mathbf{F}_{hole}(t) \cup \mathbf{F}_0(t)$ . Now, the complement of  $\mathbf{F}(t)$  is used to update the background model to  $\mathcal{B}_t$ . The set of detected foreground pixels  $\mathbf{F}(t)$  is further subjected to shadow removal (based on the criteria of equality among sub-unity intensity modulations in the 3 color channels), neighborhood voting, followed by connected component analysis to obtain the set of disjoint foreground blobs  $\mathcal{F}(t) = \{f_i(t)\}_{i=1}^{n_t}$ . These blobs are further used for multi-agent tracking and associated O-primitive identification at the next higher layer of visual processing.

### 3 Multiple Agent Tracking

The proposed multi-agent tracking algorithm works by associating the foreground blobs to the predicted agent regions and label several characteristic occlusion behaviors according to the **Persistence Hypothesis**: Objects continue to exist even when hidden from view. The agent-blob association is performed over an *active* set  $\mathcal{S}_A(t) = \{\mathcal{A}_j(t)\}_{j=1}^{m_t}$  containing the agents tracked till the  $t^{\text{th}}$  instant and also a *putative* set  $\mathcal{S}_P(t-1)$  of agents which have disappeared within the viewing window. The system initializes itself with empty sets and the agents are added (removed) as they (dis)appear in (from) the field of view. The proposed approach is a three stage process. Initially, the centroids of the pixel sets of the agents in  $\mathcal{S}_A(t-1)$  are predicted in the current frame  $\Omega_t$ . This is followed by evaluating the agent-blob associations leading to the identification of occlusion primitives. Finally, at the correction stage, the occlusion state

and agent-blob association information are used to localize the agents further in  $\Omega_t$  along with selective feature updates. The processes of agent characterization, occlusion primitive identification and agent localization under occlusions are further illustrated in the following subsections.

### 3.1 Agent Characterization

An agent is primarily detected as an extracted foreground region whose features characterize the same. The  $j^{th}$  agent  $\mathcal{A}_j(t)$  at the  $t^{th}$  instant is represented by its occupied pixel set  $\alpha_j(t)$ , the appearance model  $\mathbf{a}_j(\mathbf{u}, t)$ , the pixel visibility map  $\mathbf{v}_j(t)$  and the  $\tau$ -length trajectory  $\{\mathbf{c}_j(t), \dots, \mathbf{c}_j(t - \tau + 1)\}$  of the centroid of  $\alpha_j(t)$ . The appearance  $\mathbf{a}_j(\mathbf{u}, t)$  ( $\mathbf{u}$  is the  $(3 + 2) - D$  pixel color and position vector) is constructed as a non-parametric model with a Gaussian kernel learned over the 5-dimensional position and color space of the pixels in  $\alpha_j(t)$ . Here, the pixel positions are expressed in a local co-ordinate system centered at  $\mathbf{c}_j(t)$ .

$$\mathbf{a}_j(\mathbf{u}, t) = \frac{|\Sigma_j(t)|^{-\frac{1}{2}}}{K_j(t)(2\pi)^{\frac{5}{2}}} \sum_{k=1}^{K_j(t)} \exp\left(-\frac{1}{2}(\mathbf{u} - \mathbf{u}_k(j, t))^T \Sigma_j^{-1}(t)(\mathbf{u} - \mathbf{u}_k(j, t))\right) \quad (1)$$

$$\mathbf{u}_k(j, t) = [r_k(j, t), g_k(j, t), b_k(j, t), (\mathbf{x}_k(j, t) - \mathbf{c}_j(t))^T]^T; \mathbf{x}_k(j, t) \in \alpha_j(t) \quad (2)$$

where,  $K_j(t)$  is the number of pixels in  $\alpha_j(t)$ , ( $r_k(j, t)$ ,  $g_k(j, t)$ ,  $b_k(j, t)$ ) are the respective red, green and blue color components of the pixel  $x_k(j, t)$  and  $\Sigma_j(t) = \{\sigma_x^2(j, t), \sigma_y^2(j, t), \sigma_r^2(j, t), \sigma_g^2(j, t), \sigma_b^2(j, t)\}$  is the diagonal matrix specifying the bandwidth of the Gaussian kernel.

We identify the agent to be in one of the states of occlusion, viz. *isolation* (no occlusion), *partial occlusion* (fragmented/distorted agent appearance due to view obstruction by a background object, e.g. a pole), *crowding* (two or more agents detected in the same foreground region), *disappearance* (agent-blob association failure or complete occlusion by a background object, e.g. a thick tree) and *exit* (agent moving out of image region). However, in some cases the states of partial occlusion and crowding can happen simultaneously. The pixel set  $\alpha_j(t)$  and the appearance model  $\mathbf{a}_j(\mathbf{u}, t)$  of  $\mathcal{A}_j(t)$  are initialized from its first appearance as a foreground blob. Under partial occlusions or crowding, the agent is only partially visible and is often not localized exactly. Thus, both  $\alpha_j(t)$  and  $\mathbf{a}_j(\mathbf{u}, t)$  are updated only under the state of isolation.

### 3.2 Identifying Occlusion Primitives

The motion histories of the agents in  $\mathcal{S}_A(t - 1)$  are used to predict their centroids  $\{\mathbf{c}_j^*(t)\}_{j=1}^{m_{t-1}}$  and hence the pixel sets occupied by them as  $\{\alpha_j^*(t)\}_{j=1}^{m_{t-1}}$ . The extent of association between a predicted agent region  $\alpha_j^*(t)$  for an agent in  $\mathcal{S}_A(t - 1)$  and the foreground blob  $\mathbf{f}_i(t)$  is estimated by constructing a thresholded *localization confidence matrix*  $\Theta_{AF}(t)$  and an *attribution confidence matrix*  $\Psi_{FA}(t)$ . These confidences are computed by a fractional overlap measure  $\gamma(\omega_1, \omega_2) = \frac{|\omega_1 \cap \omega_2|}{|\omega_1|}$  signifying the fraction of the region  $\omega_1$  overlapped with  $\omega_2$ .

$$\Theta_{AF}[j, i](t) = \begin{cases} 1; & \gamma(\alpha_j^*(t), \mathbf{f}_i(t)) \geq \eta_A \\ 0; & \text{Otherwise} \end{cases} \quad (3)$$

$$\Psi_{FA}[i, j](t) = \begin{cases} 1; & \gamma(\mathbf{f}_i(t), \alpha_j^*(t)) \geq \eta_F \\ 0; & \text{Otherwise} \end{cases} \quad (4)$$

where the thresholds  $\eta_A$  and  $\eta_F$  signify the extent of allowable localization and attribution confidences. The number of foreground regions attributed to the  $j^{th}$  agent ( $\Theta_A[j](t) = \sum_{i=1}^{n_t} \Theta_{AF}[j, i](t)$  and  $\Psi_A[j](t) = \sum_{i=1}^{n_t} \Psi_{FA}[i, j](t)$ ) and agents localized in  $F_i(t)$  ( $\Theta_F[i](t) = \sum_{j=1}^{m_t} \Theta_{AF}[j, i](t)$  and  $\Psi_F[i](t) = \sum_{j=1}^{m_t} \Psi_{FA}[i, j](t)$ ) are further computed from these matrices to identify the occlusion primitives.

The  $j^{th}$  agent in  $\mathcal{S}_A(t-1)$  is **isolated** or unoccluded ( $\mathcal{O}(I)[j, t]$ ), if the localization confidence is significantly high and the associated foreground blob is not overlapped with other agents. However, when the agent **disappears** ( $\mathcal{O}(D)[j, t]$ ) both localization and attribution confidences fall below  $\eta_A$  and  $\eta_F$  signifying very poor or no association of the agent to any foreground blob. In case of **partial occlusions** ( $\mathcal{O}(P)[j, t]$ ), the attribution confidence of one or more foreground blobs to the  $j^{th}$  agent remains high, although the localization confidence falls significantly. On the other hand, while in a **crowd** ( $\mathcal{O}(C)[j, t]$ ), the localization confidence of the  $j^{th}$  agent in the crowded blob (overlapped with more than one agent) remains high although the attribution confidence of that blob to the agent remains low. Thus the four Boolean predicates for these occlusion primitives can be constructed as follows.

$$\mathcal{O}(I)[j, t] = \exists i [\Theta_{AF}[j, i](t) = 1] \wedge [\Theta_F[i](t) = 1] \quad (5)$$

$$\mathcal{O}(D)[j, t] = [\Theta_A[j](t) = 0] \wedge [\Psi_A[j](t) = 0] \quad (6)$$

$$\mathcal{O}(P)[j, t] = \forall i [\Psi_{FA}[i, j](t) = 1] \wedge [\Theta_F[i](t) = 1] \wedge [\Psi_A[j](t) \geq 1] \quad (7)$$

$$\mathcal{O}(C)[j, t] = \exists i [\Theta_{AF}[j, i] = 1] \wedge [\Theta_F[i](t) > 1] \quad (8)$$

To obtain the current active set  $\mathcal{S}_A(t)$ , updates are applied to pixel set, appearance and trajectory of individual agents under *isolation*, but only to the trajectory of agents under *partial occlusions* and *crowding*. Agents that have *disappeared* are moved from the active set to the putative set. This enables the system to remain updated with agent features while keeping track of them.

The **entry/reappearance** of an agent is attributed to the existence of a foreground blob  $\mathbf{f}_i(t)$  in the scene having no association with any agent from  $\mathcal{S}_A(t-1)$  and is thus detected as  $\mathcal{O}(N)_i(t) = [\Theta_F[i](t) = 0] \wedge [\Psi_F[i](t) = 0]$ . The features of the new blob  $\mathbf{f}_i(t)$  are matched against those in  $\mathcal{S}_P(t-1)$  to search for the reappearance of agents. If a match is found, the agent is moved from  $\mathcal{S}_P(t-1)$  to  $\mathcal{S}_A(t)$  and a *reappearance* ( $\mathcal{O}(R)[j, t]$ ) is noted. Otherwise, a new agent is added to  $\mathcal{S}_A(t)$  and the system detects an *entrance* ( $\mathcal{O}(E)[j, t]$ ). Similarly, an agent is declared to **exit** the scene ( $\mathcal{O}(X)[j, t]$ ), if its motion predicted region lies outside the image region and is thus removed from the active set.

### 3.3 Agent Localization Under Occlusions

The thresholded localization and attribution confidence matrices provides us with the agent-blob association information which is employed to further localize the agents under partial occlusions or crowding or both. Let us consider the general case where the agents  $\{\mathcal{A}_p(t)\}_{p=1}^{P_t}$  are associated with the foreground blobs  $\{\mathbf{f}_q(t)\}_{q=1}^{Q_t}$  where some of the agents are partially occluded by background objects while being in a crowd. The task of localizing the agents in these blobs is formulated in a k-means clustering framework. The blob pixels are assigned to one of the agents according to their membership in the corresponding agent appearance models and the agent centroids are updated iteratively with each new assignment.

Consider the pixel  $\mathbf{x}_k(t) \in \bigcup_{q=1}^{Q_t} \mathbf{f}_q(t)$  using which we form the 5-dimensional pixel co-ordinate and color vector  $\mathbf{u}_k(t)$  as given in equations 2. This pixel is labeled as  $l_k^{(s)}(t)$  according to its likelihood  $\lambda_k^{(s)}(t)$  in the  $s^{th}$  iteration of belonging in one of the agent appearance models.

$$\lambda_k(t) = \max_{p=1}^{P_t} \mathbf{a}_p(\mathbf{u}_k(t), t) \quad (9)$$

$$l_k^{(s)}(t) = \operatorname{argmax}_{p=1}^{P_t} \mathbf{a}_p(\mathbf{u}_k(t), t) \quad (10)$$

$$\mathbf{u}_k(p, t) = [r_k(t), g_k(t), b_k(t), (\mathbf{x}_k(t) - \mathbf{c}_p^{(s)}(t))^T]^T \quad (11)$$

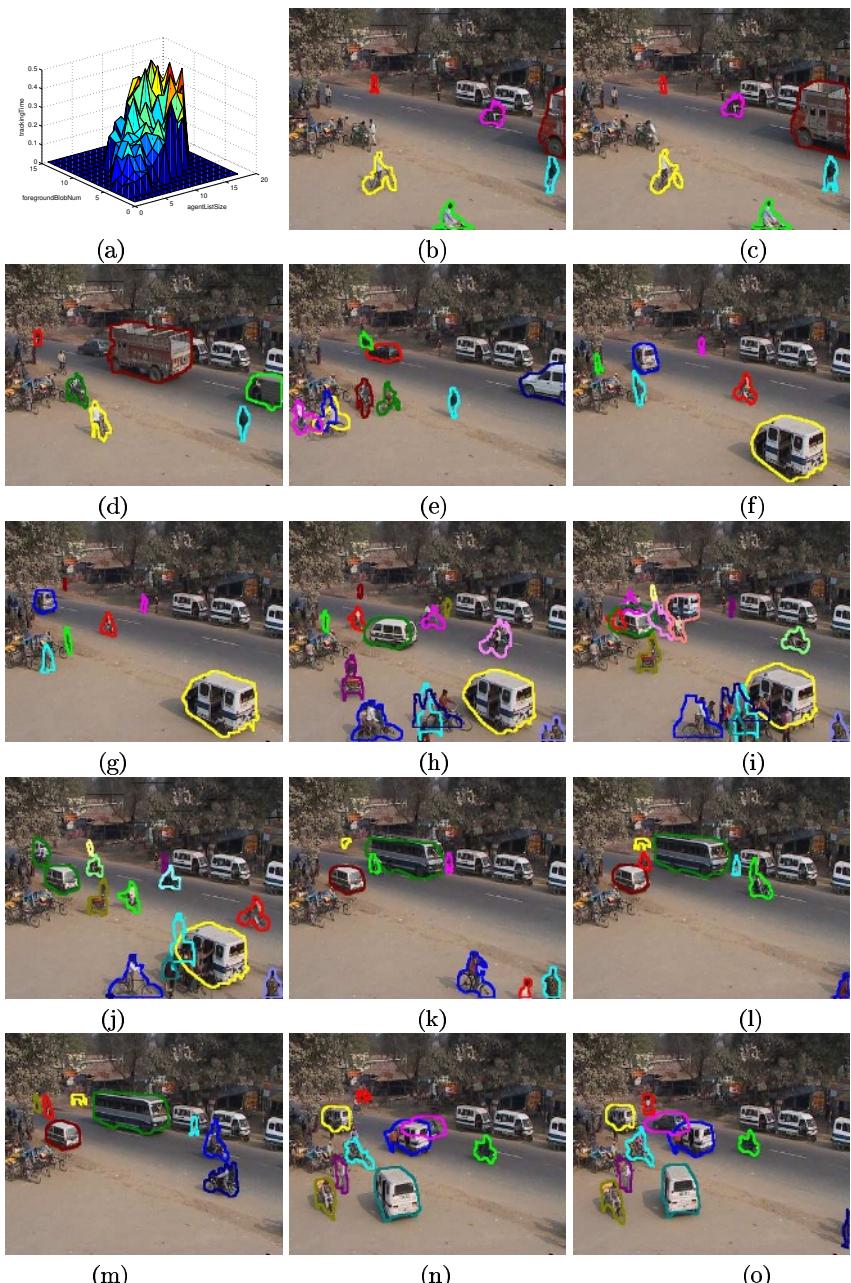
Where,  $r_k(t)$ ,  $g_k(t)$  and  $b_k(t)$  are the respective red, green and blue color components of the pixel at  $\mathbf{x}_k(t)$  and  $\mathbf{c}_j^{(s)}(t)$  is the agent centroid at the  $s^{th}$  iteration. The agent centroids are thus updated as the likelihood weighted sum of the member pixels as,

$$\mathbf{c}_p^{(s+1)}(t) = \mathbf{c}_p^{(s)}(t) + \frac{\sum_{k \in L_p^{(s)}(t)} (\mathbf{x}_k(t) - \mathbf{c}_p^{(s)}(t)) \lambda_k(t)}{\sum_{k \in L_p^{(s)}(t)} \lambda_k(t)} \quad (12)$$

Where,  $L_p^{(s)}(t) = \{k | l_k^{(s)}(t) = p\}$  is the set of pixels labeled to be belonging to the  $p^{th}$  agent in the  $s^{th}$  iteration. The new centroids obtained after the iterations are used to update the agent trajectory but not the appearance and pixel set.

## 4 Results

Experiments are performed on a traffic surveillance video of 30 minutes duration (acquired under almost constant ambient illumination conditions) consisting of a wide variety of vehicles like bikes, rickshaw, cars, heavy vehicles etc along with people and animals. The background modeling is performed by learning pixel-wise mixture of Gaussians over the *RGB* color space with a learning rate of  $\alpha = 0.01$  and a  $3 \times 3$  diagonal co-variance matrix  $\Sigma_{init} = \{4.0, 4.0, 4.0\}$ . The foreground extraction is performed with inter-frame motion information and selective model update with higher layer agent position feedback. Comparative results of foreground extraction are shown in figure 2, Section 2.



**Fig. 3.** Multi-agent tracking on traffic surveillance video. (a) Surface plot of tracking time with respect to number of active agents and foreground blobs; (b)-(o) Results of tracking in the surveillance video. The vehicles and people are shown to be tracked under complex crowding conditions, e.g. crowd boards a tempo, vehicles crossing and overtaking each other and men etc.

The overall tracking performance varies, as the thresholds  $\eta_A$  and  $\eta_F$  are changed. It is evident from equations 3 and 4 that, as the thresholds  $\eta_A$  and  $\eta_F$  are increased, the detection rates of correspondences between predicted agent regions and foreground blobs reduce and thus the rate of track loss increases. On the other hand, too low values of these thresholds would increase the number of false detections of the occlusion primitives. Thus, to achieve optimal performances, we have chosen  $\eta_A = \eta_F = 0.51$ . The kernel widths of the appearance models were empirically selected as  $\sigma_x = \sigma_y = 2.0$  for pixel co-ordinates and  $\sigma_r = \sigma_g = \sigma_b = 3.0$  for the pixel color components.

Multiple agents in the traffic video are tracked with occlusion primitive identification. The tracking performance of the  $j^{th}$  agent at the  $t^{th}$  instant is evaluated by the fraction of the ground-truth region of the same ( $\mathbf{g}_j(t)$ ) overlapped with the region  $\alpha_j(t)$ , localized by the proposed algorithm and is thus given by the quantity  $\gamma(\mathbf{g}_j(t), \alpha_j(t))$ . Hence, if there are  $m_g(t)$  number of agents present in the ground-truth marked images at the  $t^{th}$ , instant, then the overall performance  $\mathcal{P}$  for a video of  $T$  frames is given by,

$$\mathcal{P} = \frac{1}{T} \sum_{t=1}^T \frac{1}{m_g(t)} \sum_{j=1}^{m_g(t)} \gamma(\mathbf{g}_j(t), \alpha_j(t)) \quad (13)$$

The above measure of overall performance  $\mathcal{P}$  signifies the average fraction of the actual agent regions (or ground-truth regions) localized by the tracking algorithm in a certain video sequence. In our case, an overall tracking performance of 61.37% was observed.

The tracking time largely depends on the number of agents in the active and putative set along with the number of foreground blobs. The variation of the tracking time with respect to these two factors is shown in figure 3(a). It is worth noting, that an estimate of the algorithm execution time with respect to crowding can also be obtained from this graph. The results of tracking in the traffic surveillance video are shown in figure 3(b)-(o).

## 5 Conclusion

In this paper, we have proposed an algorithm for multiple agent tracking identifying occlusions with a comprehensive set of qualitative primitives. The system was found to track multiple agents satisfactorily in several complex situations. An agent is primarily detected as a change mask by the process of background subtraction. The surveillance system maintains a set of different agents, where the agent appearance (a non-parametric model learned over pixel co-ordinate and color space) and motion history form the signature of each. Our algorithm processes a dynamic set of agent signatures in an intelligent manner to identify a variety of occlusion primitives such as isolation, track loss, partial or complete occlusions, crowding and entry/exit. The proposed scheme is not restricted by any prior agent shape/motion models or ground plane assumptions and thus performs satisfactorily in relatively unconstrained environments.

This work reports a significant component of our ongoing project on semantic analysis of surveillance videos. Further enhancements to the multi-agent tracking algorithm aims at marrying Kalman filter based motion prediction along with an objective function based agent localization methodology for agent centroid update. The current algorithm is able to identify the reliability of feature updates along with various event primitives like crowding, partial/complete occlusions, and entry/exit. The future goals include parsing of temporal sequences of detected agent appearance features and occlusion primitives for the discovery and recognition of agent and scene activities thereby improving the efficiency of the surveillance system.

## References

1. Wren, C.R., Azarbayejani, A., Darell, T., Pentland, A.: Pfnder: Real-time tracking of the human body. *Pattern Analysis and Machine Intelligence* **19** (1997) 780–785
2. Haritaoglu, I., Harwood, D., Davis, L.: W4 : Real time surveillance of people and their activities. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **22** (2000) 809–830
3. Needham, C., Boyle, R.: Tracking multiple sports players through occlusion, congestion and scale. In: *Proceedings of the 12th British Machine Vision Conference*. (2001) 93–102
4. Isard, M., Blake, A.: Condensation - conditional density propagation for visual tracking. *International Journal of Computer Vision* **29** (1998) 5–28
5. Zhao, T., Nevatia, R.: Tracking multiple humans in crowded environments. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Volume 2. (2004) 406–413
6. McKenna, S., Jabri, S., Duric, Z., Rosenfeld, A.: Tracking groups of people. *Computer Vision and Image Understanding* **80** (2000)
7. Guha, P., Mukerjee, A., Venkatesh, K.: Efficient occlusion handling for multiple agent tracking with surveillance event primitives. In: *The Second Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*. (2005)
8. Zivkovic, Z.: Improved adaptive gaussian mixture model for background subtraction. In: *Proceedings of the 17th International Conference on Pattern Recognition*. Volume 2. (2004) 28–31
9. Proesmans, M., Gool, L.V., Pauwels, E., Osterlinck, A.: Determination of optical flow and its discontinuities using non-linear diffusion. In: *The 3rd European Conference on Computer Vision*. Volume 2. (1994) 295–304
10. Stauffer, C., Grimson, W.: Adaptive background mixture models for real-time tracking. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Volume 2. (1999) 246–252

# Variable Duration Motion Texture for Human Motion Modeling

Tianyu Huang, Fengxia Li, Shouyi Zhan, and Jianyuan Min

School of Computer Science and Technology,  
Beijing Institute of Technology, Beijing, 100081, China

**Abstract.** Statistical model is an effective method for character motion modeling. In this paper, a variable duration motion texture is proposed to represent complex human motion that is statistically similar to the original captured motion data. The motion texture is defined as a three-level structure with motion abstracts, motons and their distribution. The motion texture is modeled by a Semi-SLDS (Semi-Switching Linear Dynamic System), which provides an intuitive framework for describing the continuous but nonlinear dynamics of human motion. To explicitly incorporate duration modeling capability, the Semi-SLDS is adopted to improve SLDS by replacing the Markov switching layer with semi-Markov model. In addition, the proposed approach is proved flexible and effective by several motion applications, namely motion synthesis, motion recognition and motion compression.

## 1 Introduction

Motion capture is an increasingly popular approach of generating realistic character animation, which is highly applied in computer animation and computer vision. Due to the complex and rich dynamic behaviors of human motion, capturing motion data directly from real actors and mapping them to computer character becomes an effective way to produce high quality motion. To preserve the local and global dynamics of the character motion, the statistical model is a potential approach of modeling realistic human motion.

Motion texture is a successful approach in solving the problem of editing captured motion by learning motion dynamics from motion capture data. In [1], the motion texture is defined as a set of motion textons and their statistical distribution, which are presented by LDSs and a transition matrix respectively. In our paper, the motion texture is learned by Semi-SLDS — a modified version of SLDS, which takes into account the variable duration to achieve more realistic motion. Given an example, a rhythmic athlete attempts to keep jogging for certain duration to be ready for a graceful jump. In this case, Semi-SLDS model tries to keep the trends of jogging, instead of switching to next state of jumping directly. However, The SLDS, limited by Markov assumption, is inappropriate to effectively explain the regularity of duration data. In addition, in order to deal with motion at frame-level interactively, a three-level motion texture is adopted for motion modeling. By analogy to the waveform-word-distribution hierarchy in speech and

pixel-texton-distribution hierarchy [2] in image, the frame-motion-distribution hierarchy presents an increasing abstract motion statistic description.

The paper proceeds with a presentation of related work in section 2. Section 3 gives a general view of the three-level models of motion texture in details. Section 4 describes the statistical model Semi-SLDS, and learning and inference in motion texture. In section 5, the three-level motion texture applications, namely as motion synthesis, motion recognition and motion compression are provided. We make a discussion and present future work in Section 6.

## 2 Related Work

Statistical models have been studied in a variety of correlative research fields, such as computer vision [3],[4],[5], computer graphics [1],[6],[7],[8],[9], tracking, speech recognition [10],[11],[12]. In this section, the motion research on statistical models is expatiated in two aspects: linear dynamic system and nonlinear dynamic system.

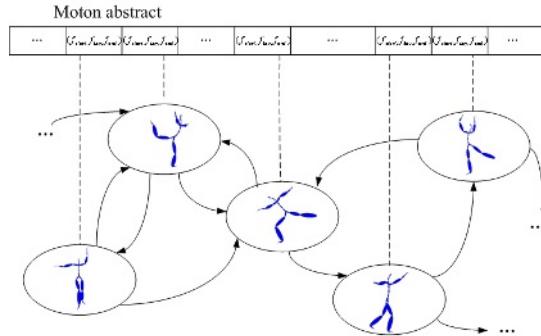
Modeling the primitives (textons or motons) with LDS was related to several recent works. Soatto, Doretto, and Wu [6] posed the problems of modeling, learning, recognizing and synthesizing dynamic textures. Schodl, Szeliski, Salesin and Essa [13] presented video textures for analysis and synthesis. Yuan, Wen, Liu, and Shum [14] improved on synthesizing dynamic Texture with closed-loop LDS. These LDS applications in computer vision model the temporal behavior as samples of an underlying continuous process, and were effective for spatially coherent textures. Bregler [15] used second-order dynamic systems to represent the dynamical categories of human motion, which only recognized simple gait. Li, Wang, Shun [1] confined each LDS element as a "texton" that was constrained to start and end at specific key-frame.

Many approaches have been proposed to model complex motion with nonlinear systems. Reference [8], [9] described motion dynamics by a Hidden Markov Model (HMM), with which the motion primitives cannot be edited explicitly because they were represented as hidden states. North, Blake, Isard, and Rittscher [7] learnt multiple classes of motions by combining EM [16] and condensation. Reference [4],[5] devised approximate inference methods to learn a SLDS model. Ren, Patrick, Efros, Hodgins and Rehg [17] addressed that each LDS component in an SLDS model of motion defined a family of trajectories with linear dynamics. But with the Markov assumption, the inference results of SLDS were restricted to geometric distribution without considering duration factor. The same problem was previously addressed in [10],[11], [12] and solved by several HMM extensions, which provided enhanced duration modeling. The switching hidden semi-Markov model [18] is a similar hierarchical model with our modeling method in discrete state space. Generally, A SLDS model can be viewed as a generalization of a HMM in which each switching state is associated with a LDS instead of a Gaussian distribution over the output space [5]. Durational extensions to SLDS are closely related to the model structure and extensions described in [19].

In our work, a three-level statistical model is designed for modeling character motion with rich dynamics flexibly and effectively. A Semi-SLDS model with enhanced duration modeling capabilities is used to learn the captured motion data.

### 3 Three-Level Motion Texture

We propose a three-level motion texture, i.e. *moton abstract – moton – moton distribution*, to model character motions. In this model, a Semi-SLDS is adopted to present the statistical models of characteristic motion. We assume that there are  $n$  motons  $M_t(t \in [1, n])$ , and each one can represent one cluster of statistical similar motion segments. The property of moton, called moton abstract, is represented by its key-frames. Moton distribution can be described by the probability that a moton is switched to another. Our three-level motion texture characterizes the dynamic and stochastic nature of the figure motion as illustrated in Fig.1. First, the specific key-frame in any motons can be accessed efficiently by the clue of moton abstract. Secondly, we use a Semi-SLDS to capture the local linear dynamics and model the global non-linear dynamics. Thirdly, we use motons to describe the repeated patterns in the stochastic process. The presentation of each level of motion texture is shown as follow:



**Fig. 1.** A three-level motion texture model with moton abstracts, motons and their distribution

(1) Moton abstract of  $M_t$  is represented by a triple  $(f_{start}, f_{key}, f_{end})$ , where  $f_{start}$ ,  $f_{key}$ ,  $f_{end}$  are the start frame, key frame, end frame of the moton respectively. The triple is assumed to describe the property of a moton. In our experiments, each moton is defined in the range of 50 frames to 160 frames so that local dynamics can be captured. The key frame  $f_{key}$  can be found by the algorithm introduced in [20].

(2) Each moton is represented by a model  $LDS$  defined by the following state-space model:

$$\begin{cases} X_{t+1} = AX_t + w_t, & w_t \sim N(0, Q) \\ Y_t = CX_t + v_t, & v_t \sim N(0, V) \end{cases}. \quad (1)$$

In our experiments, we model a human skeleton with 19 joints and its global translation. Variable  $Y_t$  is constructed with a 79-dimensional vector, with joint angles approximated by rotations of unit quaternion and 3-dimensional translation displacement. And we apply PCA on  $Y_t$  to represent  $X_t$  by 8-12 most significant principal vectors of  $Y_t$ .  $w_t$  and  $v_t$  are white noise.

(3) We assume that motions satisfy the first-order Markov distributions, which is represented by a transition matrix:

$$T_{i,j} \triangleq P(l_k = j | l_{k-1} = i). \quad (2)$$

Such a transition matrix indicates the probability of switching from one LDS state to another.

## 4 Semi-SLDS for Motion Modeling

### 4.1 Introduction of SLDS

A LDS is a time-series state-space model consisting of a linear Gaussian dynamics model and a linear Gaussian observation model. In a SLDS, we assume that there are  $n$  distinct LDSs represented by  $LDS \triangleq \{LDS_l | 1 \leq l \leq n\}$ , where  $L \triangleq \{l_t | 1 \leq t \leq T\}$ . In addition to a set of models, SLDS specifies two additional parameters: a multinomial distribution over the initial label and a transition matrix that defines the switching behavior between the LDSs.

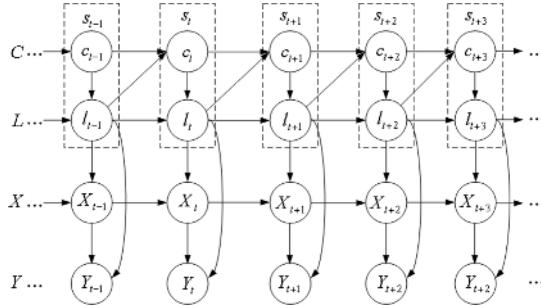
An important weakness of the conventional SLDS is its inflexibility to model state durations, because the Markov assumption is imposed on the transitions between the discrete LDSs. If  $d$  is the duration of a particular state  $S_k$ , then the probability of  $d$  is denoted by  $P_k(d) = T_{kk}^{d-1}(1 - T_{kk})$ , which results in that one step in a duration is the most probability.

As discussed in Section 1, a rhythmic athlete attempts to stay in one state for certain duration to be ready for a next action. In such a case, the geometric distribution induced in SLDS is not an appropriate choice to model the duration patterns. Hence, a SLDS model incorporated with duration modeling is required in our models.

### 4.2 Description of Semi-SLDS

A Semi-SLDS is defined as an improved SLDS model by relaxing the Markov assumption from a time-step level to a segment-level. In the Semi-SLDS model, the duration functions learn from training data. As a consequence, the Semi-SLDS model has more descriptive capabilities in modeling duration, and more robust inference capabilities than SLDS. In this subsection, the Semi-SLDS is given by the variable duration form.

Fig.2 illustrates the representation of a Semi-SLDS. As this figure shows, the top chain is a series of segments. Each segment is represented by a pair of label  $l_i$  and duration  $d_i$ , i.e.  $s_i \triangleq (l_i, d_i)$ . Within each segment a fixed LDS model  $LDS_l$  is used to generate the continuous state sequence for the duration  $d_i$ .



**Fig. 2.** Representation of a Semi-SLDS

The distribution  $\pi(l_i)$  is multinomial about the label  $l_i$  of the first segment  $s_i$ , and a  $n \times n$  matrix  $\tilde{T}$  defines the switching probability between the segment labels. Additionally, each label is associated with a fixed duration model  $D_l$ , represented as a multinomial, and the set of  $n$  duration models are denoted as  $D \triangleq \{D_l(d) | 1 \leq l \leq n\}$ .

In summary, a Semi-SLDS with duration version is defined as  $\Theta = \{\pi, \tilde{T}, D, LDS\}$ . The dynamics for the continuous hidden states and observation is similar to SLDS: a segment  $s_i$  calls the continuous hidden states  $X$  with a corresponding LDS model  $LDS_{l_i}$  for the duration  $D_i$ , then the observations are computed by given the labels  $L$  and the continuous states  $X$ . Thus, continuous motion sequences can be generated by modeling with Semi-SLDS. The current segment  $s_i = (l_i, d_i)$  transits a next segment  $s_{i+1}$  in the following manner: first, the current label  $l_i$  generates the next label  $l_{i+1}$  based on the label transition matrix  $\tilde{T}$ ; then, the next duration  $d_{i+1}$  is generated from the duration model  $D_{l_{i+1}}(d)$  for the label  $l_{i+1}$ .

An additional set of incremental variables  $C \triangleq \{c_t | 1 \leq t \leq n\}$  are defined to perform the segmentation. All  $c_t$  are initialized by zero. If the counter simply increases, then the new label is set to be the current label  $l_t$ , thus the system stays in the same segment. If the counter variable  $c_{t+1}$  is reset to zero, then a label transition occurs, thus a new segment is initialized. A new label  $l_{t+1}$  is chosen based on the label transition matrix  $\tilde{T}$ .

### 4.3 Learning and Inference in Semi-SLDS

Learning in Semi-SLDS is similar to learn in conventional SLDS, using EM algorithm [16] to obtain the maximum-likelihood parameters.

Due to more parameters in Semi-SLDS, the existing inference methods are not available. Hence, by merging multiple discrete variables into one variable, a Semi-SLDS can be converted to an equivalent SLDS. Thus the inference methods in SLDS also can be applied in Semi-SLDS.

Specifically, all possible pairs of a label  $l_t$  and a counter value  $c_t$  are merged into "lc" variables, and  $LC \triangleq \{(l, c_t) | 1 \leq l \leq 1, 1 \leq c_i \leq D_l^{max}\}$ . An equivalent

$n' \times n'$  transition matrix  $T'$ , where  $n' = \sum_{l=1}^n (D_l^{max})$  is constructed from transition matrix  $\tilde{T}$  and the transition functions  $F$ , as follows :

$$T'_{(l_i, c_i), (l_j, c_j)} = \begin{cases} \tilde{T}_{l_i, l_j}(1 - F_{l_i}(c_i)), & \text{where } c_j = 0 \\ F_{l_i}(c_i), & \text{where } l_i = l_j \text{ and } c_j = c_i + 1 \\ 0, & \text{others} \end{cases}. \quad (3)$$

where  $F_l(c_t) = 1 - (D_l(c_t) / \sum_{d=c_t}^{D_l^{max}} D_l(c_t))$ .  $D_l^{max}$  denotes the maximum duration associated with the  $l$ th model. Intuitively,  $D_l(c_t) / \sum D_l(c_t)$  denotes the probability to reset the counter variable .

In addition, the initial label distribution  $\pi'$  for the equivalent SLDS can similarly be constructed from the Semi-SLDS initial distribution  $\pi$  :

$$\pi'(l_i, c_i) = \begin{cases} \pi(l_i), & \text{if } c_i = 1 \\ 0, & \text{others} \end{cases}. \quad (4)$$

The duration models  $D$  only influence the label sequence  $L$ , and hence the ML estimates of the parameters can be evaluated from a durational representation of the label sequence  $L$ . A Poisson distribution is used as duration model in our paper.

There have been several researches of efficient approximate inference methods. In this paper an approximate Viterbi [3],[4],[5] method is used for inference. The linear efficiency can be maintained by the sparseness of the constructed SLDS matrix  $T'$  , because only a few transitions are allowed between the states in  $LC$ . The transitions include the resetting to the labels and an increment transition. Hence, we achieve an performance of  $O(TD_{max}|L|^2)$ . This complexity allows to incorporate a duration model with a large  $D_{max}$  and maintain computational efficiency.

## 5 Applications and Experiments

Our experiments were performed on a motion capture datasets including walking, jogging, running and jumping. The motion collection consists of 117 trails which contain 31160 frames with 60Hz frame rate. We have implemented the algorithm by Matlab, and it took approximately 1 hour to learn the motion texture on a Pentium M 1.73G computer with 512M memory. After learning, motion texture was modeled by totally56 motons. With the learnt motion texture, motion synthesis, motion recognition and motion compression can be efficiently realized to prove out method in this paper.

### 5.1 Motion Synthesis

With the learnt motion texture, new motion can be synthesized. Motion texture decouples global nonlinear dynamics (moton distribution) from local linear dynamics (motons), which contains the property of the dynamics (moton abstract). Accordingly, a three-step synthesis method is demonstrated as follows.

First, by the BKS-tree [21] retrieval method, the start motion and the end motion can be retrieved according to the given key-poses. Secondly, a motion path can be generated from the state space by two alterative methods, namely the Dijkstra's algorithm and the method specifying the path length. Thirdly, since we have the motion path, a motion sequence can be synthesized frame by frame with the learnt Semi-SLDS and sampled noise. To achieve smooth transition between two neighbor motions, we apply a constrained synthesis algorithm with two additional frames at the beginning of the synthesized segment.

Due to considering the duration of each state in Semi-SLDS, movement trends are kept in the synthesized motion. Fig.3 and Fig.4 show the synthesized results. Fig.3(a) and Fig.3(b) give the comparison of synthesis by SLDS and Semi-SLDS respectively. After we sketched two key-poses, running pose and walking pose, the motion texture learnt by Semi-SLDS can keep the duration of the running in two time step, then transit to walk. However, the SLDS model transits the running state to walking state directly. In Fig. 4, we gave two similar boundary frames, and a jumping pose in addition. A motion with jump key-frame can be found by BKS-tree retrieval method, and a graceful motion can be synthesized.

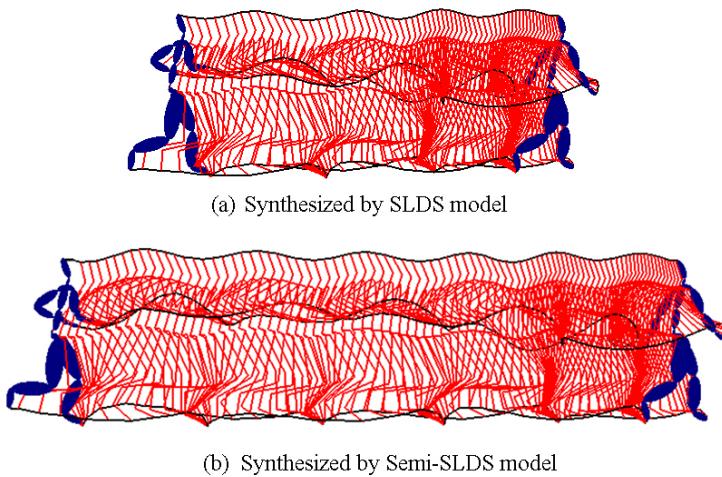
## 5.2 Motion Recognition

The second experiment explored the utility of the Semi-SLDS model in improving motion behavior recognition. We used a second-order model in our experiments. In this model, trajectories begin at an initial state that is described by a mixture of Gaussians. As the trajectory evolves, the state of the motion at time  $t$  is described by a linear combination of the state values at times  $t - 1$  and  $t - 2$  and the addition of Gaussian noise. By switching between these LDS components, the Semi-SLDS can model a system with nonlinear and non-Gaussian dynamics.

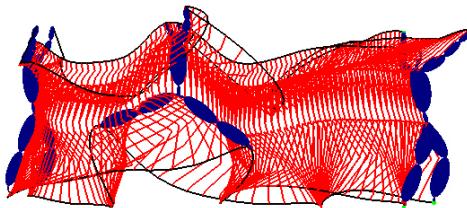
The label inference results for given motion is shown in Fig.5. Fig.5(a) shows the action recognition, and Fig.5(b) demonstrates the walking direction recognition. The experimental results show the superior recognition capabilities of the Semi-SLDS model over the original SLDS model. This is possible because Semi-SLDS uses the additional durational information to robustly discern the subtle differences.

## 5.3 Motion Compression

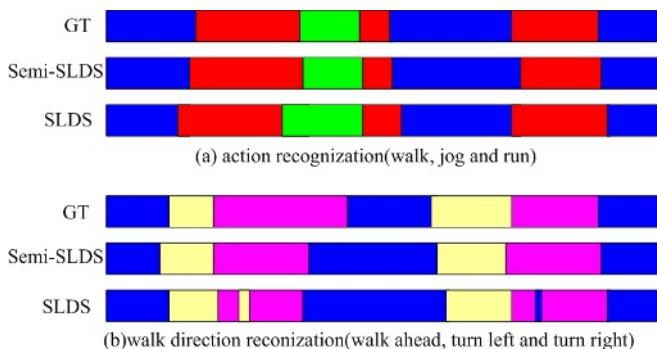
As proved above, the motion texture model can re-exhibit the original motion. Since parameters for describing the model is less than the quantity of original capture data, the motion texture model can be applied in large amount of motion compression. The storage requirement of the original dataset is  $O(mt)$  ( $m$  denotes the dimensions of motion data,  $t$  frames totally), whereas the storage for the motion texture model is  $O(mn + n^2 + nk + kt)$  ( $n$  denotes the number of learnt states,  $n \leq t$ , and  $k$  denotes the reduced dimensions of motion data,  $k \leq m$ ). In our experiments, in order to store a model of the original motion data with  $O(10^6)$  would only need the storage about  $O(10^4)$ .



**Fig. 3.** Comparison of synthesis results by SLDS model and Semi-SLDS model



**Fig. 4.** Given the specific key-pose, the proper motons can be retrieved by the BKS-tree method, and then the motion can be synthesized by learnt motion texture



**Fig. 5.** Label inference results. Estimates from Semi-SLDS and SLDS models are compared to manually ground truth (GT) labels.

## 6 Conclusion and Discussion

We demonstrated a method of motion modeling based on three-level motion texture to achieve more effective and flexible motion. The main contributions are that the access of key-frames in motion made the dynamics of motion applications in frame-level and that the adoption of Semi-SLDS modeling made the synthesized motion more realistic.

In our experiments, the motion segments for training motion texture are labeled by ground truth, which may result in the inaccurate motion. We argue that more accurate physical measure of motion can solve this problem. Our experiments were tested on a set of motions with repeated patterns. For the reason that with the increasing size of state space in Semi-SLDS, the efficiency of inference need to be improved if the testing motion set is a random motion database.

Our presented work supplies a flexible and effective way of representing dynamics of human motion in motion synthesis, motion recognition and motion compression. In the future work, we are looking forward to developing the framework favored by accurate and automatic segment for motion behaviors, which improves the capabilities of the motion texture.

## References

1. Li, Y., Wang, T., Shun, H. Y.: Motion texture: A two-level statistical Model for Character Motion Synthesis. In Proceedings of SIGGRAPH2002. (2002)
2. Zhu, S. Ch., Guo, Ch. E., Wang, Y. Zh., Xu, Z. J.: What Are Textons. Int. Journal of Computer Vision **62** (2005) 121–143
3. Pavlović, V., Rehg, J.M.: Impact of dynamic model learning on classification of human motion. In IEEE Conf. on Computer Vision and Pattern Recognition. (2000)
4. Pavlović, V., Rehg, J.M., Cham, T.J., Murphy, K.: A dynamic Bayesian network approach to figure tracking using learned dynamic models. In Intl. Conf. on Computer Vision. (1999)
5. Pavlović, V., Rehg, J.M., MacCormick, J.: Learning switching linear models of human motion. In Advances in Neural Information Processing Systems. (2000) 981–987
6. Soatto,S., Doretto, G., Wu, Y. N.: Dynamic textures. IEEE International Conference on Computer Vision. 2002 439–446
7. North, B., Blake, A., Isard, M., Rittscher, J.:Learning and classification of complex dynamics. IEEE Transactions on Pattern Analysis and Machine Intelligence. **22** (2000) 1016–1034
8. Brand, M., Hertzmann, A.: Style machines. In Proceedings of SIGGRAPH00. (2000) 183–192
9. Tanco, L. M., Hilton, A.: Realistic synthesis of novel human movements from a database of motion capture examples. IEEE Workshop on Human Motion. (2000)
10. Levinson, S. E.: Continuously variable duration hidden Markov models for automatic speech recognition. Computer Speech and Language. **1** (1990) 29–45
11. Ostendorf,M, Digalakis, V. V. and Kimball, O. A.: From hmm's to segment models: A unified view of stochastic modeling for speech recognition. IEEE Transactions on Speech and Audio Processing.**4** (1996) 360–378

12. Ferguson, J.D.: Variable duration models for speech. Symposium on the Application of Hidden Markov Models to Text and Speech. (1980) 143–179
13. Schodl, A., Szeliski, R., Salesin, D.H., Essa, I. Video textures. Proceedings of ACM SIGGRAPH 00. (2000) 489–498
14. Yuan, L., Wen, F., Liu, C., Shum, H. Y.: Synthesizing dynamic texture with closed-loop linear dynamic system. European Conference on Computer Vision. (2004) 603–601
15. Bregler, C.: Learning and recognizing human dynamics in video sequences. Int. Conf. on Computer Vision and Pattern Recognition. (1997) 568–574
16. Dempster, N. M., Laird, A. P., Rubin, D. B.: Maximum likelihood from incomplete data via the EM algorithm. **39** (1977) 185–197
17. Ren, L., Patrick, A., Efros, A., Hodgins, J., Rehg, G. M.: A data-driven approach to quantifying natural human motion. ACM Trans. on Graphics, Special Issue: Proc. of 2005 SIGGRAPH Conf. **24** (2005) 1090–1097
18. Duong, T., Bui, H., Phung, D., Vekatesh, S.: Activity recognition and abnormality detectiong with the switching hidden semi-Markov model. IEEE International Conference on Computer Vision and Pattern Recognition. (2005)
19. Oh, S. M., Rehg, J. M., Balch, T., Dellaert, F.: Learning and Inference in Parametric Switching Linear Dynamic Systems. Intl. Conf. on Computer Vision . **2** (2005) 1161–1168
20. Assa, J., Caspi, Y., Cohen, O.: Action Synopsis: Pose Selection and Illustration. ACM Transactions on Graphics. **24** (2005) 667–676
21. Huang, T. Y., Li, F.X., Zhan, S. Y.: Motion Retrieval Based on Binary Key-pose Split Tree. in press.

# A Novel Motion Blending Approach Based on Fuzzy Clustering

Zhu Xiangbin

School of Information Science and Engineering, Zhejiang Normal University,  
Jinhua Zhejiang , China  
zhu\_xb@163 . com

**Abstract.** Motion blending allows the generation of new motions by interpolation or transition between motion capture sequences, which is widely accepted as a standard technique in computer animation. But traditional blending approaches let the user choose manually the transition time and duration. This paper presents a new motion blending method for smoothly blending between two motion capture clips and automatically selecting the transition time and duration. To evaluate the effectiveness of the improved method, we have done extensive experiments. The experiment results show that the novel motion blending method is effective in smoothly blending between two motion sequences.

## 1 Introduction

Motion capture is a popular process for generating human animation. The motion capture data can be made reusable: this may mean using previous motion capture data to generate new motions so that certain requirements are met. Moreover motions can be transferred from one skeletal configuration to another so that we can animate multiple figures with the same motion, or changing the style of the motion so that the directors can have higher level control over the motion. For synthesizing new human motion, motion data must be edited to produce new motion clips to be adapted to meet the demands of a particular situation. People have developed many motion editing methods. An important one of motion editing is motion blending, which generates new motion clips by combining multiple clips according to time-varying weights<sup>[1]</sup>. It takes the data received from several motion capture passes and averages and interpolates it to provide a smoother form of motion animation.

Motion blending has several applications. For example, blending can be used to create seamless transitions between motions, allowing one to build lengthy, complicated motions out of simpler actions. Another application is interpolation, or creating motions “inbetween” the initial set to produce a parameterized space of motions<sup>[1]</sup>. Blending operations such as these have significant practical importance and have proven useful in commercial applications like video games<sup>[2][3]</sup>.

But it is not easy to blending two motion clips to produce a convincing motion clip unless the input motions are chosen very carefully. If the two motion clips are very different, a meaning motion clip almost can't be got. So in motion blending, a key problem is to find two quite similar motion frame sequences in two motion clips, used to generate the new motion.

In the exist methods, they are always to select a transition point which has the least distance between two corresponding frames according to some measure methods. But motion blending need two similar motion subsequences which usually have scores of frames. Although one frame of one motion subsequence is quite similar to one frame of the other subsequence, one motion subsequence may be not similar to the other one from a whole. This also makes the resulting produced motion to be not coherent. Therefore, we should consider the similar between two motion sequences.

We propose a novel method based on fuzzy clustering which is used to find the quite similar motion sequences in two motion clips. Our methodology consists first, in an off-line process, to computing all distances between each frame pair about the two motion subsequences. Second is to identify the candidate transitions using fuzzy clustering method. Third, for we match subsequences instead of individual frames, we should filter the difference matrix with a diagonal kernel with weights [ $w_{1,k}, \dots, w_{k-1,k}$ ]. In addition, if one distance in a subsequence is larger a threshold, the corresponding candidate transition can't be selected.

In the remainder of this paper we describe the novel motion blending method. The next section starts with an overview of motion blending. Section 3 introduces the related fuzzy clustering theory. Section 4 presents the blending algorithm that exploits fuzzy clustering theory. Finally, Section 5 demonstrates our technique in common applications and Section 6 concludes with a discussion of the advantages and limitations of the new method.

## 2 Related Work

Recent years have seen a number of algorithms for motion synthesis from motion data for reusing or altering existing motions. Motion warping<sup>[4]</sup> can be used to smoothly add small changes to a motion. Retargeting<sup>[5][6]</sup> maps the motion of a performer to a character of different proportions while retaining important constraints like footplants. Various signal processing operations<sup>[7]</sup> can be applied to motion data. Wiley et al<sup>[8]</sup> proposed the interpolation synthesis algorithm that chooses and combines most relevant motions from the database to produce animation with a specific positional goals.

Motion blending has been an important part of these research efforts. One of the earliest was due to Perlin<sup>[9]</sup>, who presented a real-time system based on procedurally generated motions. After a user manually constructed base motions, blending operations were used to create new motions and transition between existing ones. The language for constructing motions contained built-in timing and constraint models, simplifying the related blending issues. Several systems have leveraged multitarget blending to create intuitively parameterized spaces of motions<sup>[8][10][11]</sup>. Multiple research groups have used general blending (that is, blending with arbitrary weight functions) to provide continuous control of locomotion<sup>[12][13][14]</sup>. Blend-based transitions have been incorporated into various systems for graph-based motion synthesis<sup>[10][15]</sup>. Instead of averaging skeletal parameters directly, Unuma et al.<sup>[16]</sup> used weighted averages in the Fourier domain. Rose et al.<sup>[17]</sup> used spacetime optimization to create transitions that minimize joint torque. Other researchers<sup>[18][19][20]</sup> have generated transitions by concatenating motions and eliminating discontinuities with displacement

mapping techniques. Timewarping has been an important part of previous work on motion blending.

These algorithms create novel motions by cutting pieces from a motion database and reassembling them. Motion graphs<sup>[15]</sup> present a novel method for creating realistic, controllable motion, which uses a directed graph called a motion graph to generate novel motion.

However, this method is ill suited if the desired actions have short temporal span, such as “jumping” or “catching” or if the actions are to be composed: “jump and catch while running”.

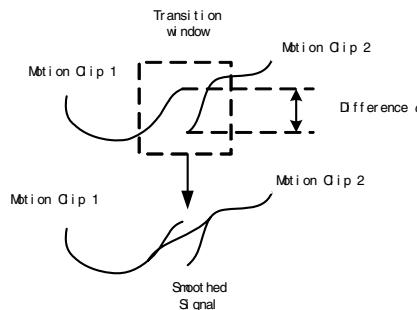
Motion concatenation also need motion blending technology. Motion transition has been achieved with both frequency and time domain methods. The key problem is to find the similar motion subsequences.

The general method is to search the correspondence frames(The distance of two frames is minimum) as a transition point. There are several of calculate methods. Kovar et al<sup>[15]</sup> measures the distance between two frames of animation in terms of point cloud driven by the skeleton. The distance is expressed as:

$$\min_{\theta, x_0, z_0} \sum_{k=1}^{n_p} w_k \| p_k - T_{\theta, x_0, z_0} p_k \| ^2 \quad (1)$$

Lee et al.<sup>[18]</sup> uses the equation 2 to compute the distance between two frames:

$$d(p_i, p_j) = \| p_{i,0} - p_{j,0} \|^2 + \sum_{k=1}^m w_k \| \log(q_{j,k}^{-1} q_{i,k}) \|^2 \quad (2)$$



**Fig. 1.** Smooth transition

Although we can find the transition point, there is also some difference between the two frames. If one motion jumped directly to another motion at transition point, it is not coherent yet. Rose et al.<sup>[17]</sup> uses easy-in/easy-out weight function to make the smooth transition . Define  $\alpha=0.5\cos(\beta\pi)+0.5$ , in the transition interval, the transition subsequences is a linear interpolation of the two motion subsequences. Arikan et al.<sup>[19]</sup> distributes the difference among these frames in the transition window to get a smooth transition, illustrated at fig.1. The algorithm interpolates the difference  $d$  and adds the result to original signal for getting the smooth signal.

### 3 Fuzzy Clustering

Clustering involves the task of dividing data points into homogeneous classes or clusters so that items in the same class are as similar as possible and items in different classes are as dissimilar as possible. Depending on the data and the application, different types of similarity measures may be used to identify classes, where the similarity measure controls how the clusters are formed. Some examples of values that can be used as similarity measures include distance, connectivity, and intensity.

A fuzzy set has elements that have a varying degree of membership in that set, i.e., the degree of membership has a value in the closed interval [0, 1]. In contrast, for a classical, or crisp set, membership has no degree. An element is either a member or it is not a member. Let  $X$  be a space of points (i.e. all possible elements of concern in a particular context). Let  $X$  be called the universe of discourse. The membership function of a fuzzy set  $A$  in  $X$ ,  $\mu_A: X \rightarrow [0, 1]$ .

A fuzzy relation is a fuzzy equivalence relation if the reflexive, symmetric, and transitive properties are satisfied. The fuzzy equivalence relation can be used for fuzzy clustering<sup>[21][22]</sup>. Multiplying by itself can get the equivalence relation of a fuzzy relation  $R$ . For example,  $R \circ R = R^2$ ,  $R^2 \circ R^2 = R^4$ , ... When  $R^{2k} = R^k$ ,  $R^k$  is an equivalence relation. The following fuzzy relation is an equivalence relation because it is reflexive, symmetric, and transitive properties. It satisfies  $R \circ R = R^2$ .

$$R = \begin{bmatrix} 1 & 0.48 & 0.62 & 0.41 & 0.47 \\ 0.48 & 1 & 0.48 & 0.41 & 0.47 \\ 0.62 & 0.48 & 1 & 0.41 & 0.47 \\ 0.41 & 0.41 & 0.41 & 1 & 0.41 \\ 0.47 & 0.47 & 0.47 & 0.41 & 1 \end{bmatrix}$$

A standard (non-fuzzy or crisp) equivalence relation  $R_\lambda$  can be obtained from a fuzzy equivalence relation, say  $R_e$ , by applying a  $\lambda$ -cut operation on the fuzzy relation  $R_e$ . The crisp relation  $R_\lambda$  is defined as  $R_\lambda = \{(i, j) | \mu_{R_e}(i, j) \geq \lambda\}$ , i.e., if the strength of the relation between  $i$  and  $j$  is greater than or equal to  $\lambda$ , it is replaced by 1, and 0 otherwise. Depending on the values of  $\lambda$ , outputs can be classified differently. For example, if  $\lambda=0.48$ , the crisp equivalence relation becomes:

$$R = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

and the outputs are classified as  $\{x_1, x_2, x_3\}, \{x_4\}, \{x_5\}$ .

### 4 Motion Blending

To do motion blending, we first need to get a small window, in which a transition will be operated on two video clips. There are a lot of approaches to get this window.

#### 4.1 Locating Candidate Transitions Using Fuzzy Clustering

Motion Capture data is typically represented as vectors of parameters specifying the root position and joint rotations of a skeleton on each frame. One might attempt to locate transition points by computing some vector norm to measure the difference between poses at each pair of frames. To find places where motions are similar, we introduce a distance metric for comparing frames of motion. This is accomplished through a scalar function  $d(F_i, F_j)$  that defines the distance between two frames  $F_i$  and  $F_j$ . We use the same distance function as in [18].

$$d(F_i, F_j) = \| p_{i,0} - p_{j,0} \|^2 + \sum_{k=1}^m w_k \| \log(q_{j,k}^{-1} q_{i,k}) \|^2$$

So we can a fuzzy relation  $R$ :

$$R = \begin{bmatrix} r_{00} & r_{01} & \cdots & r_{0j} & \cdots & r_{0(n-2)} & r_{0(n-1)} \\ r_{10} & r_{11} & \cdots & r_{1j} & \cdots & r_{1(n-2)} & r_{1(n-1)} \\ \vdots & \vdots & & \vdots & & \vdots & \vdots \\ r_{i0} & r_{i1} & \cdots & r_{ij} & \cdots & r_{i(n-2)} & r_{i(n-1)} \\ \vdots & \vdots & & \vdots & & \vdots & \vdots \\ r_{(n-2)0} & r_{(n-2)1} & \cdots & r_{(n-2)j} & \cdots & r_{(n-2)(n-2)} & r_{(n-2)(n-1)} \\ r_{(n-1)0} & r_{(n-1)1} & \cdots & r_{(n-1)j} & \cdots & r_{(n-1)(n-2)} & r_{(n-1)(n-1)} \end{bmatrix} \quad (4)$$

where  $r_{i,j} = (\text{MaxDistance} - d(F_i, F_j)) / \text{MaxDistance}$

$$\text{MaxDistance} = \max(d(F_i, F_j)) \quad 0 \leq i \leq n-1, 0 \leq j \leq n-1$$

$n$  is the number of frames in two motion clips that is the range to search match frames.

Thus, we can use the above fuzzy clustering to get some clusters. For these clusters which elements is larger 1 all is the candidate transition point.

#### 4.2 Selecting Transition Points

For these candidate clusters, we need select the ideal transition point. In this paper, we first determine the best match point in two clips, and then create a blending window by extending a certain size in two directions (Fig.2). The minimum in the distance function

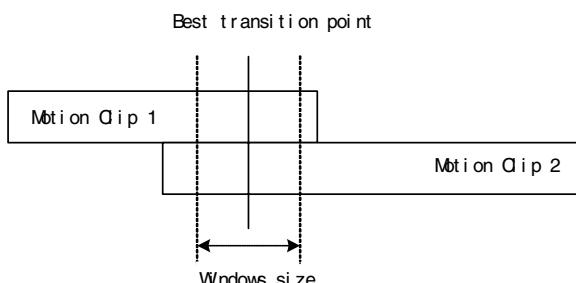


Fig. 2. Transition window

does not necessarily imply a high-quality transition. It only implies a transition better than its neighbors and can't assure the small window is the best blending range. So we need computer the difference of the temporally adjacent frames within some weighted window must be similar to each other. In other words, we match subsequences instead of individual frames. Such a subsequence match can be achieved by filtering the difference matrix with a diagonal kernel with weights  $[w_{-m}, \dots, w_{m-1}]$ ,

$$d_{i,j} = \sum_{k=-m}^{m-1} w_k d_{(i+k)(j+k)} \quad (5)$$

$$\text{where } w_k = \begin{cases} 2\left(\frac{p+1}{2m}\right)^3 - 3\left(\frac{p+1}{2m}\right)^2 + 1 & p \leq m \\ 1 - 2\left(\frac{p+1}{2m}\right)^3 + 3\left(\frac{p+1}{2m}\right)^2 - 1 & p > m \end{cases}$$

Moreover, each distance  $d_{i,j}$  in the small window should be below an empirically determined threshold. This is to prevent the new motion clip from dithering at some frames. User will want to set the threshold themselves to pick an acceptable tradeoff between having good transitions and having high connectivity.

Window size should be changed properly if the distance of two clips is very large. To make the new clip contains the specific actions from the origin video clips, which we are interested in, the match point, can not be in the interested area in both clips. So region constraints are added to make the result in the appropriate boundary. The search region should be in the ending part of the first clip, and in the beginning part of the second clip.

### 4.3 Creating Transitions

As soon as the transition window is determined, we can make the transition now. The principle is to make the transition from one side of window to the other side window, by using different weights for the two clips. The root position  $\mathbf{p}'$  during the transition is a linear interpolation of the root positions  $\mathbf{p1}$  from motion clip M1 and the root positions  $\mathbf{p2}$  from the other motion clip M2:

$$\mathbf{p}'_k = \alpha \mathbf{p1}_{i-m+k} + (1-\alpha) \mathbf{p2}_{j-m+k} \quad (6)$$

where  $k \in [0, 2m]$  and  $\alpha$  is defined as:

$$\alpha = 2\left(\frac{k+1}{2m}\right)^3 - 3\left(\frac{k+1}{2m}\right)^2 + 1 \quad (7)$$

Similarly, each joint orientation  $\mathbf{q}'$  in the transition is a spherical linear interpolation of the corresponding joint orientations  $\mathbf{q1}$  in M1 and  $\mathbf{q2}$  in M2<sup>[15]</sup>:

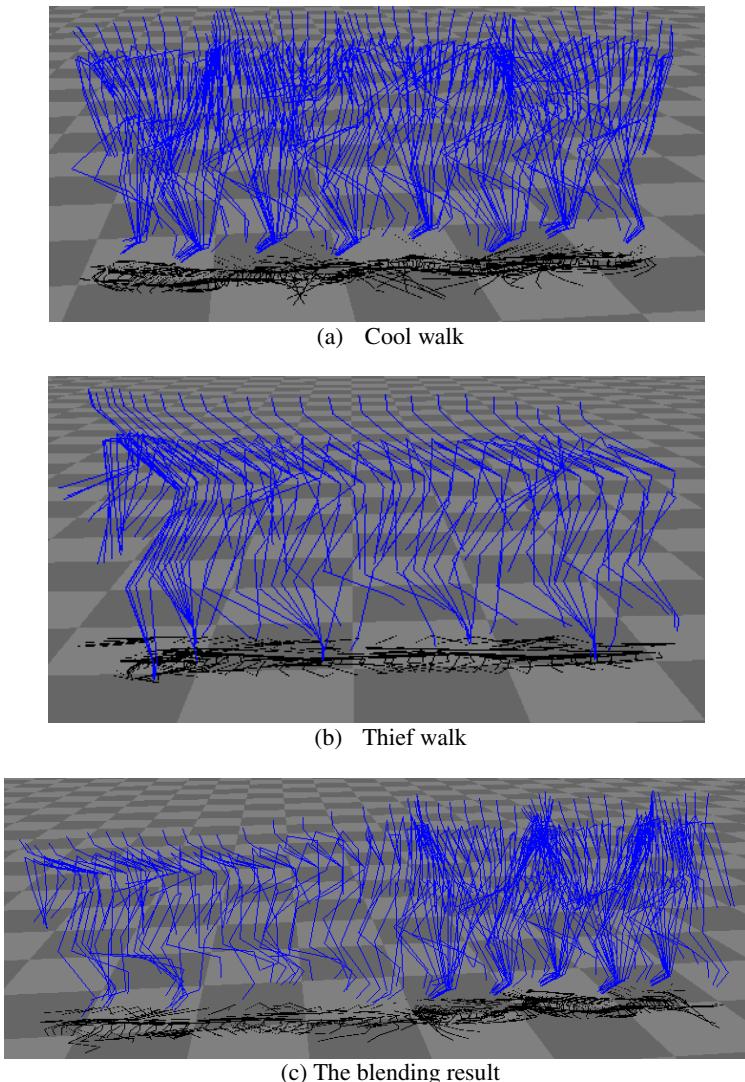
$$\mathbf{q}'_k = \text{slerp}(\mathbf{q2}_{j-m+k}, \mathbf{q1}_{i-m+k}, \alpha) \quad (8)$$

In the transition duration,  $\alpha$  changes from 1 to 0 linearly according to the frame position in the transition window. So that  $1-\alpha$  changes from 0 to 1 in a cosine curve, which may bring us much more smooth transition than the simple linear one can. After the transition, the data will be recentered in X-axis and Y-axis for better view.

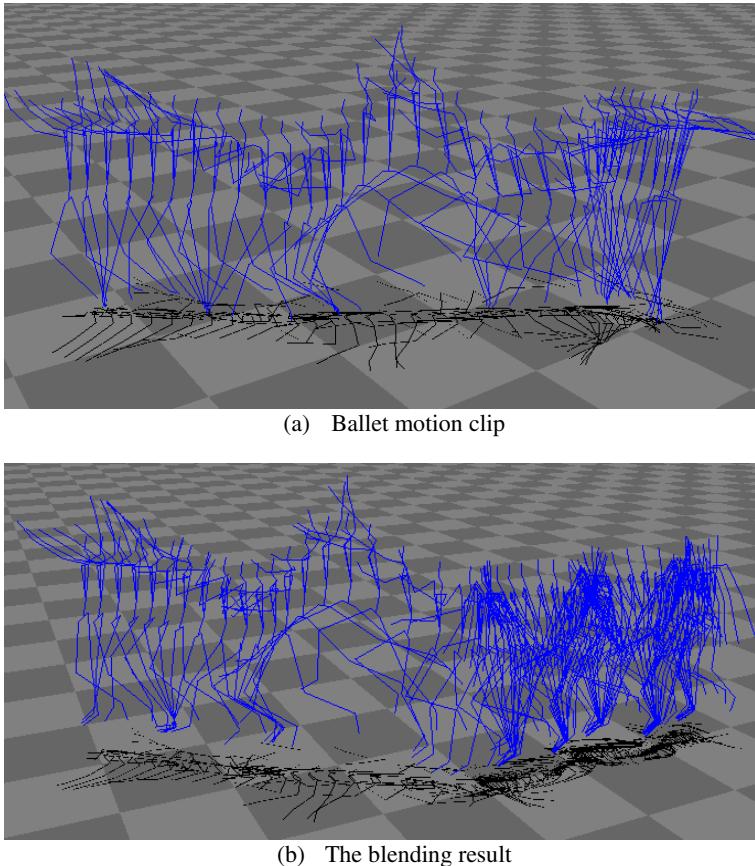
## 5 Results

We have used the motion blending approach of this paper to create seamless transitions between motions. Fig.3 and Fig.4 show our experiment results.

Fig.3(a) shows a cool walk and Fig.3(b) is a thief walk. Fig.3 (c) shows the effect of the transition between cool walk and thief walk by using our method.



**Fig. 3.** The motion blending between cool walk and thief walk



**Fig. 4.** The motion blending between cool walk and ballet

Fig.4(a) shows a ballet motion. Fig.4 (b) shows the effect of the transition between cool walk and ballet by using our method. This result also shows that the range for searching match frames must be carefully selected in motion clips. In some time, the blending result may lose some important motion elements. The ballet motion showed in Fig.4(a) has a round motion element.

## 6 Conclusion and Future Work

In this paper we present a general method for motion blending that could be used to create seamless transition among motion clips. Animators can use this approach to produce a new move sequence.

The implementation in this paper is not complicated, but it still gets a nice result. The only apparent drawback is that the thresholds of fuzzy clusters must be specified by hand, since different kinds of motions have different fidelity requirements.

## References

1. Lucas Kovar and Michael Gleicher. Flexible Automatic Motion Blending with Registration Curves. Proceedings of 2003 Symposium on Computer Animation. July 2003
2. M. Mizuguchi, J. Buchanan, and T. Calvert. Data driven motion transitions for interactive games. In Eurographics 2001 Short Presentations, September 2001.
3. S. Theodore. Understanding animation blending. Game Developer, pages 30–35, May 2002.
4. A. Witkin and Z. Popović. Motion warping. In Proceedings of ACM SIGGRAPH 95, Annual Conference Series, pages 105– 108. ACM SIGGRAPH, August 1995.
5. M. Gleicher. Retargeting motion to new characters. In Proceedings Of ACM SIGGRAPH 98, Annual Conference Series, pages 33–42. ACM SIGGRAPH, July 1998.
6. J. Lee and S. Y. Shin. A hierarchical approach to interactive motion editing for human-like figures. In Proceedings of ACM SIGGRAPH 99, Annual Conference Series, pages 39– 48. ACM SIGGRAPH, August 1999.
7. A. Bruderlin and L. Williams. Motion signal processing. In Proceedings of ACM SIGGRAPH 95, Annual Conference Series, pages 97–104. ACM SIGGRAPH, August 1995.
8. D. Wiley and J. Hahn. Interpolation synthesis of articulated figure motion. IEEE Computer Graphics and Application, 17(6):39–45, 1997.
9. K. Perlin. Real time responsive animation with personality. IEEE Transactions on Visualization and Computer Graphics, 1(1):5–15, March 1995.
10. C. Rose, M. Cohen, and B. Bodenheimer. Verbs and adverbs: Multidimensional motion interpolation. IEEE Computer Graphics and Application, 18(5):32–40, 1998.
11. C. Rose, P. Sloan, and M. Cohen. Artist-directed inversekinematics using radial basis function interpolation. Proceedings of Eurographics 2001, 20(3), 2001.
12. S. Guo and J. Robege. A high-level control mechanism for human locomotion based on parametric frame space interpolation. In Proceedings of Eurographics Workshop on Computer Animation and Simulation '96, 1996.
13. G. Ashraf and K. C. Wong. Generating consistent motion transition via decoupled frame-space interpolation. Computer Graphics Forum, 2000.
14. S. I. Park, H. J. Shin, and S. Y. Shin. On-line locomotion generation based on motion blending. In Proceedings of ACM SIGGRAPH Symposium on Computer Animation 2002. ACM SIGGRAPH, July 2002.
15. L. Kovar, M. Gleicher, and F. Pighin. Motion graphs. In Proceedings of ACM SIGGRAPH 2002, Annual Conference Series. ACM SIGGRAPH, July 2002.
16. M. Unuma, K. Anjo, and T. Tekeuchi. Fourier principles for emotion-based human figure animation. In Proceedings of ACM SIGGRAPH 95, Annual Conference Series, pages 91– 96. ACM SIGGRAPH, 1995.
17. C. Rose, B. Guenter, B. Bodenheimer, and M. Cohen. Efficient generation of motion transitions using spacetime constraints. In Proceedings of ACM SIGGRAPH 1996, Annual Conference Series, pages 147–154. ACM SIGGRAPH, August 1996
18. J. Lee, J. Chai, P. Reitsma, J. Hodgins, and N. Pollard. Interactive control of avatars animated with human motion data. In Proceedings of ACM SIGGRAPH 2002, Annual Conference Series. ACM SIGGRAPH, July 2002.
19. Arikan and D. A. Forsyth. Interactive motion generation from examples. In Proceedings of ACM SIGGRAPH 2002, Annual Conference Series. ACM SIGGRAPH, July 2002.

20. Katherine Pullen and Christoph Bregler. Motion capture assisted animation: Texturing and synthesis. In Proceedings of ACM SIGGRAPH 2002, Annual Conference Series. ACM SIGGRAPH, July 2002.
21. S. Tamura, S. Higuchi, and K. Tanaka. Pattern Classification Based on Fuzzy Relations, IEEE Transactions on Systems, Man, and Cybernetics, Vol(1), No.1, Jan, 1971.
22. J. Bezdeck and D. Harris. Fuzzy Partitions and Relations: An Axiomatic Basis for Clustering, Fuzzy Sets and Systems (1) pp 111-127, 1978.

# Efficient Optimization of Inpainting Scheme and Line Scratch Detection for Old Film Restoration

Seong-Whan Kim and Ki-Hong Ko

Department of Computer Science, University of Seoul, Jeon-Nong-Dong, Seoul, Korea  
swkim7@uos.ac.kr and jedigo@venus.uos.ac.kr

**Abstract.** Old films usually have typical damages from dirt, scratch, and scribbling. These damages make an image degradation of vertical line scratches or blotches in frames. This paper proposes an efficient line scratch detection technique and efficient inpainting method based on MSE (mean square error) to fill the identified line scratch areas. Previous line scratch detection algorithms can only detect full column line scratches; however, we found that partial line scratch should also be identified for better film restoration. We identify line scratches using block-by-block inspection; thereby we can detect partial line scratches. After identifying the line scratches, we use a modified inpainting scheme, which uses MSE measure to compute gradient vector of the inpainting regions. In our experiment with old Korean films, we show that our scheme gives better video quality with much reduced computational complexity.

**Keywords:** Inpainting, Line Scratch, Isophote line, MSE, and Restoration.

## 1 Introduction

Film damages can be global over entire film area, or it can be localized in a frame. Color fading, shake, and flicker are the typical example of global film damages. Dirt, scratch, film grain noise, film tear, and digital drop-out are the example of localized damages. Old films usually have typical damages from dirt, scratch, and scribbling, and these damages can be shown in the form of vertical line scratches or blotches in frames. Currently, film restoration is usually done by specially trained film engineers. Because the restoration process of film usually requires large amount of film data to be restored, the automatic restoration or pre-processing for restoration is required [1]. In this paper, we propose an efficient line scratch detection technique and MSE (mean square error) based inpainting method to fill the identified line scratch areas. Previous line scratch detection algorithms can only detect full column line scratches. However, old film academics also require partial line scratch be identified for better film restoration. From this investigation, we modified the line scratch detection algorithm using block based approaches, to detect partial column line scratches. After identifying the line scratches, we use a modified inpainting scheme, which uses MSE (mean square error) measure to compute gradient vector of the inpainting regions. We experiment our line scratch detection and restoration scheme with two old Korean films, and we show better restoration results than traditional line scratch and inpainting based schemes.

## 2 Related Works

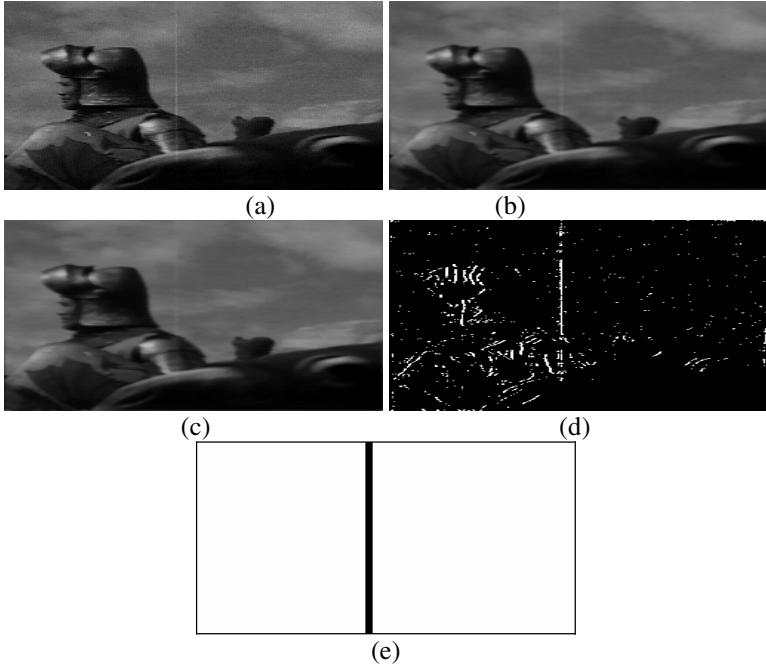
Film damages can be global or localized. Color fading, shake, and flicker are the typical example of global film damages. For restoration of global film damages such as color fading, shake, and flicker, we can use (1) motion estimation based approach with multiple frames, (2) histogram matching method, and (3) LMMSE (linear minimum mean-square error) [1, 2]. Dirt, scratch, film grain noise, film tear, and digital drop out are the localized damages, and it is more frequent than global damages. Localized damage occurs because of dusts, scribbling, scratch and furious flames from film projector. The most important localized damages are blotch and line scratch. For the blotches, they are usually spotted randomly all around entire frame because of dust and scribbling nature [3]. The blotches can be detected from motion vector estimation using previous and next frame of film. Line scratches are bright or dark intensity-oriented line, which is more or less vertically over much of the image. They may be caused when material from some particle is smeared vertically over the film material in the projector or by the abrasion of the film as it passes over some particle caught in the mechanism. Unlike blotch, a line scratch persists in the same spatial location for more than one frame. It can not be characterized as a temporal discontinuity, but be characterized analyzing each column characteristic in a frame [4].

Kokaram proposed a line scratch detection scheme: To detect line scratches for image as shown in Figure 1(a), the scheme computes the difference between 1x5 vertically oriented Gaussian filtered image as shown in Figure 1(b) and 5x1 horizontally median filtered image as shown in Figure 1(c). Figure 1(d) is a difference image from Figure 1(b) to Figure 1(c). After setting threshold for the difference images, the scheme uses Hough transform to find the candidate for line scratch.

For each candidate line scratch, the scheme computes a brightness distribution, and compares the distribution with the line profile  $L_n^{(p)}(i, j)$  to determine that the candidate line is true line scratches [4].  $L_n^{(p)}(i, j)$  is a  $P$  th line profile on  $(i, j)$  is coordinates of  $n$  th frame as defined in Equation (1), where  $b_p$  is the brightness of the central portion of the line,  $k_p$  is the decay of the line profile such that  $0.05 < k_p < 0.95$ ,  $w_p$  is the line width, and  $m_p$ ,  $c_p$  are the slope and intercept with the horizontal edge of the image which defines the orientation of the  $P$  th straight line profile. The resulting line scratch detection for Figure 1(a) is shown in Figure 1(e).

$$L_n^{(p)}(i, j) = b_p k_p^{|i - (m_p j + c_p)|} \cos\left(\frac{3\pi|i - (m_p j + c_p)|}{2w_p}\right) \quad (1)$$

Another approach to detect line scratch is the energy based model proposed by Vitulano [5]. It uses only the column index of the input image. It uses only the column index of the input image.  $L_n^{(p)}$  is now a simple sinusoid as equation (2), where  $b_p$  is the brightness of the central portion of the line,  $w_p$  is the line width, and  $c_p$  is the column position of the line.

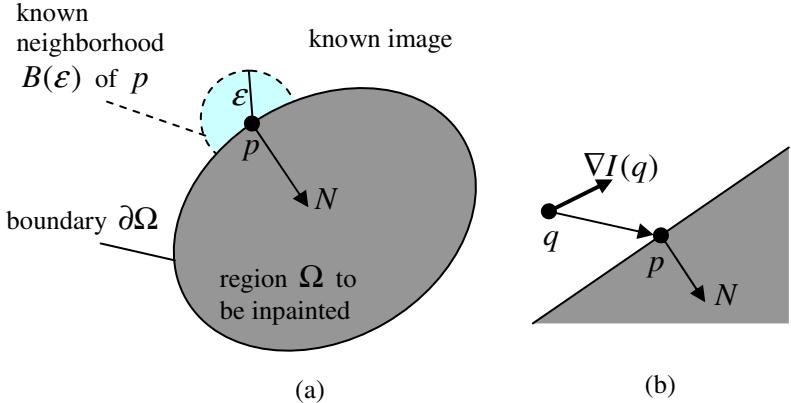


**Fig. 1.** Kokaram's method to detect line scratch (a) damaged image, (b) Gaussian filtered image from (a), (c) median filtered image from (b), (d) difference image between (b) and (c), (e) detected line scratch

$$L_n^{(p)}(i) = b_p \cos\left(\frac{3\pi|i - c_p|}{2w_p}\right) \quad (2)$$

Digital inpainting is the technique of restoring small damaged portions of an old image when the original image is unknown. Figure 2(a) shows Bertalmio's inpainting scheme of approximating the point  $P$  value from neighboring pixel values in  $\mathcal{E}$  areas. The point  $P$  is on the boundary  $\partial\Omega$  of the region  $\Omega$ . From a small neighborhood  $B_\epsilon(p)$  of size  $\epsilon$  of the known image around  $P$ , the inpainting of  $P$  is determined by the values of the known image points  $q$  in  $B_\epsilon(p)$  [7]. This scheme prolongs the isophotes lines arriving at  $\partial\Omega$ , while maintaining the angle of arrival, and the region  $\Omega$  are filled with the structure of the area  $\mathcal{E}$  preserving isophote lines.

Figure 2(b) shows Telea's simplified inpainting technique [8]. To inpaint point  $P$ , they use Equation (3), which is a normalized weighted sum of all estimates from all  $q$  in  $B_\epsilon(p)$ . For  $\epsilon$  small enough, a first order approximation  $I_q(p)$  of the image in point  $P$  can be computed by equation (5), where  $I(q)$  is the known image pixel and  $\nabla I(q)$  is gradient value of point  $q$  [8]. The weighting function  $w(p,q)$  consists of the directional component  $dir(p,q)$  which ensures that the contribution of the pixels close to the normal direction  $N$ , and the geometric distance component  $dst(p,q)$  which decreases the contribution of the pixels geometrically farther from  $P$  [8].



**Fig. 2.** The basic principle of Inpainting: (a) Bertalmio's inpainting method, (b) Telea's inpainting method [8]

$$I(p) = \frac{\sum_{q \in B_\epsilon(p)} w(p, q) \cdot I_q(p)}{\sum_{q \in B_\epsilon(p)} w(p, q)} \quad (3)$$

$$I_q(p) = I(q) + \nabla I(q)(p - q)$$

$$\nabla I(q) = (I(k+1, l) - I(k-1, l), I(k, l+1) - I(k, l-1))$$

$$w(p, q) = \text{dir}(p, q) \cdot \text{dst}(p, q), \text{dir}(p, q) = \frac{p - q}{\|p - q\|} \cdot N(p), \text{dst}(p, q) = \frac{1}{\|p - q\|^2}.$$

### 3 Joint Line Scratch Detection and Restoration Algorithm

In this paper, we improve Kokaram's method to detect line scratch. It provides not only detecting column position which has line scratch, but also detecting block position in selected column position which have true scratch. After the line scratch detection, we use MSE based inpainting method which compute gradient vector of inpainting point to improve directional weighting component  $\text{dir}(p, q)$ , and this algorithm improves video quality in restoration process. Figure 3 shows the overall scheme. We used YUV 4:2:0 video formats for our experimentation of line scratch restoration scheme. Only Y value is needed in line scratch detection process, but inpainting process uses color information to restore damaged video. As shown in Figure 3, we create Gaussian filtered image  $G(i, j)$  and Median filtered image  $M(i, j)$  from input YUV image, and obtain  $e(i, j)$  using the difference between  $G(i, j)$  and  $M(i, j)$ . Then we create scratch image from thresholds  $T_e$ ,  $T_b$ ,  $T_s$ . Now, we can inpaint damaged area from input YUV image and scratch image.

Figure 4 shows our block based line detection algorithm. We compute  $e(i, j) = G(i, j) - M(i, j)$ , where  $G(i, j)$  is a vertically oriented Gaussian filtered image and  $M(i, j)$  is the horizontally median filtered of  $G(i, j)$ . By setting

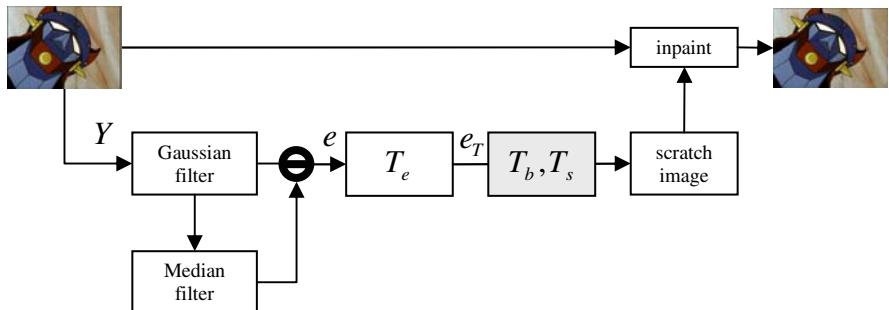
threshold  $e(i, j)$  using  $T_e$ , we get  $e_T(i, j)$ . After that, we use Equation (4) to inspect each block whether it has enough line features.  $s_{c_t, b}$  is the summation of  $e_T(i, j)$  in  $3 \times 16$  block  $b$  which include column  $c_t$  and neighborhood columns. If  $s_{c_t, b}$  is greater than  $T_s$ , we determine that the block is line-featured.

$$e_T(i, j) = \begin{cases} 0, & \text{if } e(i, j) < T_e \\ 1, & \text{if } e(i, j) \geq T_e \end{cases} \quad (4)$$

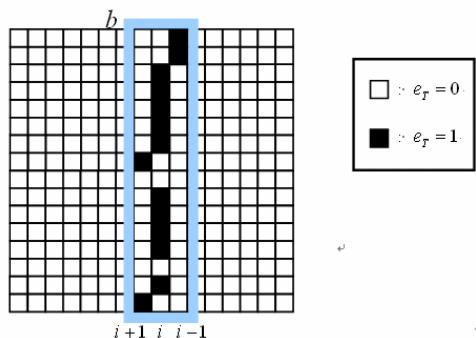
$$s_{c_t, b} = \frac{\sum_{i=c_t, j \in b} e_T(i, j) + 0.5e_T(i-1, j) + 0.5e_T(i+1, j)}{\text{block size } (=16)} > T_s$$

Finally, we can determine that the column  $c_t$  is true line if it includes enough line featured blocks as defined in Equation (5).

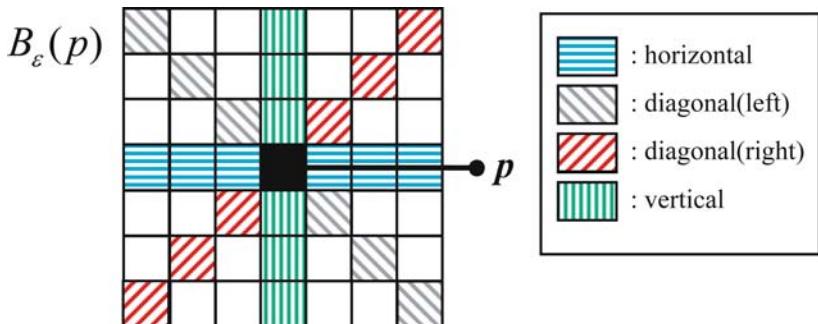
$$\frac{\text{number of line featured blocks in } c_t}{\text{number of blocks in } c_t} > T_b \quad (5)$$



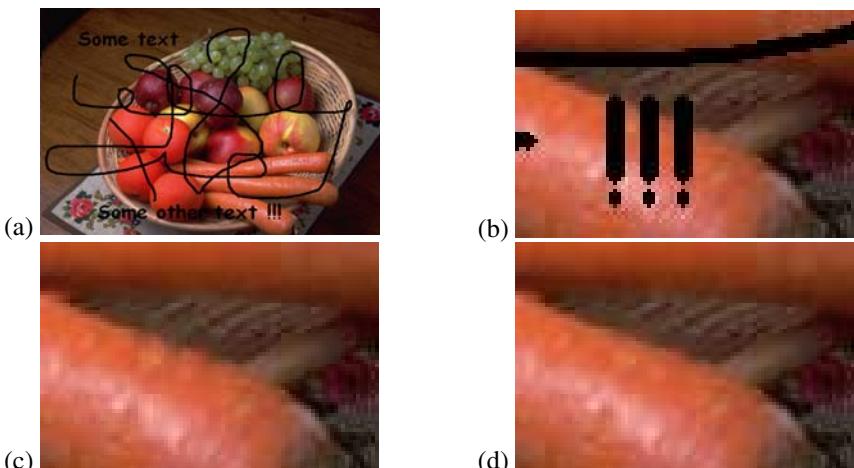
**Fig. 3.** Proposed block based line scratch detection and MSE based inpainting scheme



**Fig. 4.** Block based line feature detection



**Fig. 5.** Efficient gradient vector computation using MSE for four directional candidates



**Fig. 6.** Inpainting results for fruit test image (a) degraded image, (b) zoomed image, (c) Bertalmio's method, (d) MSE based method

Like other inpainting algorithm, we should find the isophote to compute directional weighting component  $dir(p, q)$ , but the isophote  $N(p)$  is difficult to compute and time consuming. As shown in Figure 5, we compute an approximation of isophote using 4 directional (horizontal, vertical, and diagonal) matching in  $B_\epsilon$ . The matching is based on MSE computation between each pixel values in the direction and the mean of  $B_\epsilon$ . Once the minimum MSE direction is chosen, it is the isophote  $N(p)$  (perpendicular line of gradient vector).

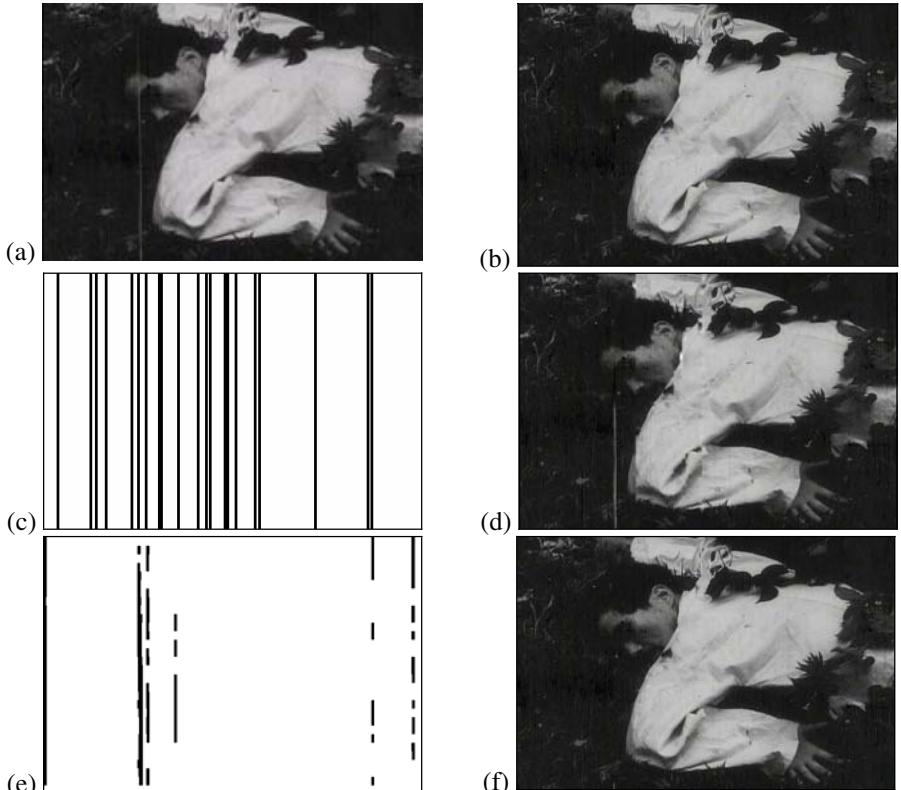
We compute MSE value by Equation (6), where  $D$  is known area in each line,  $n_{\mathfrak{R}}$  is the number of pixels in  $D$ ,  $I_x$  is intensity of coordinates  $\mathcal{X}$ , and  $\mu_{\mathfrak{R}}$  is mean value of  $D$ . If MSE value of entire  $B_\epsilon$  is minimal, we set  $dir(p, q)$  to 1, which means that we does not consider directional component in weight function.

$$MSE(D) = \frac{1}{n_D - 1} \sum_{x \in D} (I_x - \mu_D)^2 \quad (6)$$

For computational efficiency, we define only 4 lines, but the result of inpainting is better than previous method as shown in Figure 6(d).

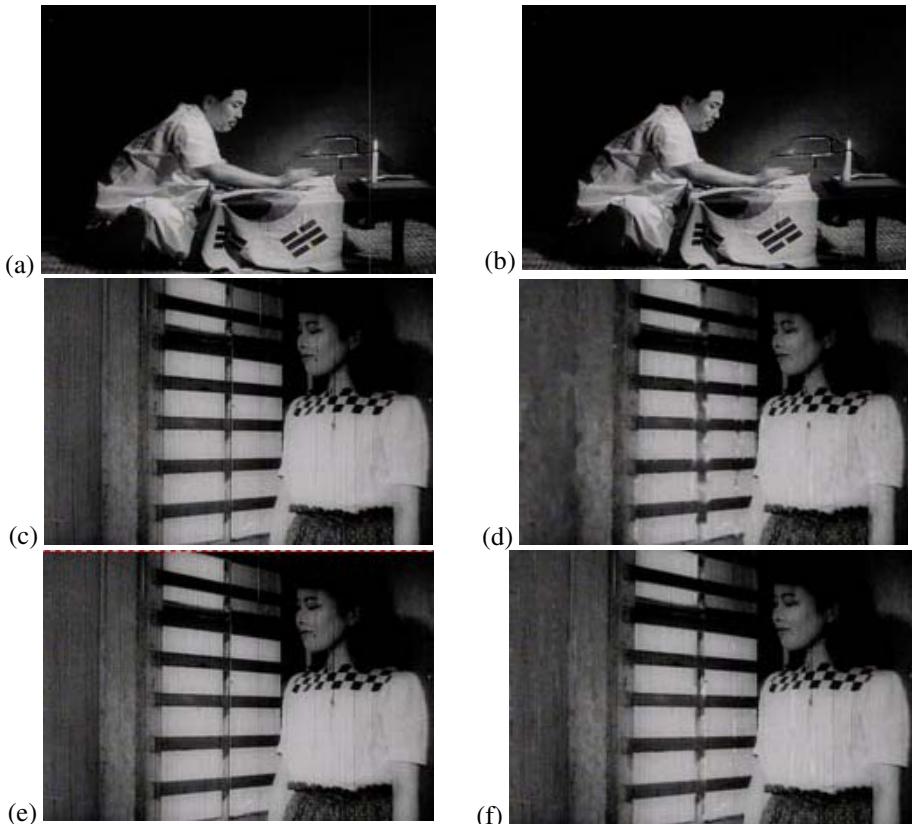
## 4 Experimental Results for Old Korean Films

In our experiment, we used a grey scale film sequences: “Viva Freedom!” (1946). We converted to 720x480 YUV video to restore, and experimentally set the threshold values as;  $T_e = 3$ ,  $T_b = 0.4$ ,  $T_s = 0.5$ . Figure 7(a) shows original degraded image and Figure 7(c)(e) show a detected line scratch by energy based model and proposed method. As shown in Figure 7(c), energy based model detects many useless line scratches, so that it makes bad effect on restoration. Our proposed method detects only important line scratches and it makes good quality after inpainting.



**Fig. 7.** Comparison of line scratch detection: (a) degraded image, (b) restored image (manually), (c, d) energy based method and the restored image, (e, f) proposed method ( $T_e = 3$ ,  $T_b = 0.4$ ,  $T_s = 0.5$ ) and the restored image

For “Viva Freedom!” most line scratches are detected and restored successfully. Figure 8(c) shows false line detection case, where the wooden door seems to have line scratch. Consequently restored result is much blurred and distorted as Figure 8(d) shows. This kind of false alarm problem can be corrected by controlling threshold values. As shown in Figure 8(e), we can choose  $T_e = 6, T_b = 0.4, T_s = 0.5$  for optimum threshold values.



**Fig. 8.** Optimum control of threshold values for restoration of “Viva Freedom!” (a, c, e) degraded frames, (b, d) restored frame using  $T_e = 3, T_b = 0.4, T_s = 0.5$ , (f) restored frame using  $T_e = 6, T_b = 0.4, T_s = 0.5$

## 5 Conclusions

This paper proposes an efficient line scratch detection technique and MSE (mean square error) based inpainting method to fill the identified line scratch areas. Previous research works seeks only the full line scratch; however, our approach can find partial line scratch, which is more important for better film restoration. We find the parital

line scratch using block by block line feature summarization approach. If each block in a line has enough line feature, the line can be a true line after summing up the line feature in each block. This partial line scratch detection capability improves the restored film quality. After identifying the line scratches, we use a efficient inpainting scheme, which uses MSE measure to compute isophote. We experiment old Korean film “Viva Freedom!” (1941), and the result showed better restoration quality than traditional line scratch detection and restoration schemes.

**Acknowledgment.** We acknowledge Korea Film Archive (KOFA) for their providing old Korean film for our research.

## References

1. Ko, S.-J., Lee, S.-H., Jeon, S.-W. and Kang, E.-S.: Fast Digital Image Stabilizer based on Gray-coded Bit-plane Matching: IEEE Trans. on Consumer Electronics, Vol. 45, No. 3 (1999) 598-603
2. Vlachos, T.: Flicker Correction for Archived Film Sequences Using a Nonlinear Model: IEEE Trans. On Circuits and Systems for Video Technology, vol. 14, no. 4 (2004) 508-516
3. Kokaram, Anil C.: On missing data treatment for degraded video and film archives: a survey and a new Bayesian approach: IEEE Trans. Image Proc., vol. 13, no. 3 (2004)
4. Kokaram, Anil C.: Detection and removal of line scratches in degraded motion picture sequences: Signal Processing VIII, vol. 1 (1996) 5-8
5. Vitulano, D., Bruni, V. and Ciarlini, P.: Line scratch detection on digital images: an energy based model: IAC-CNR, Rome, Italy (2001)
6. Bruni, V. and Vitulano, D.: A generalized model for scratch detection: IEEE Trans. Image Proc., vol. 13, no. 1 (2004) 44-50
7. Bertalmio, M., Sapiro, G., Caselles, V. and Ballester, C.: Image Inpainting: In Proceedings SIGGRAPH 2000, Computer Graphics Proceedings, Annual Conference Series, edited by Kurt Akeley, Addison-Wesley (2000) 417-424
8. Telea, A.: An Image Inpainting Technique Based on the Fast Marching Method: journal of graphics tools, vol. 9, no. 1 (2003) 25-36

# Partial Encryption of Digital Contents Using Face Detection Algorithm

Kwangjin Hong and Keechul Jung

HCI Lab., Department of Media, Graduate school of Soongsil University,  
Seoul, South Korea  
[{hongmsz\\_kcjung}@ssu.ac.kr](mailto:{hongmsz_kcjung}@ssu.ac.kr)  
<http://hci.ssu.ac.kr>

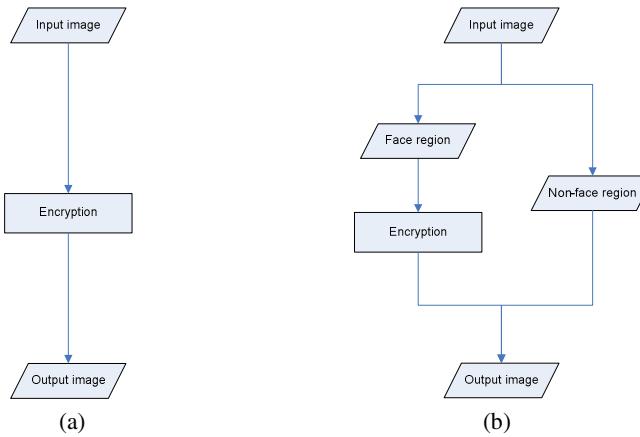
**Abstract.** Recently, a great number of people can share the same digital contents, because it is possible to copy and to transmit of the digital contents easy and fast. These properties of the digital contents are causes of that reduce the will to creation of makers and that hamper industrial development. Therefore recent studies focus on the protection of digital contents. However it is not efficient that traditional encryption algorithms apply to the digital image/video contents, because of the long encryption time. To solve this problem, recent studies use the partial encryption algorithm that encrypts some parts of the image or the video frame. However there are still problems which features do not have the semantic information, because previous studies extract the features for reducing the encryption time. In this paper, we proposed the partial encryption method using the face region as the feature because the face has the semantic information and is the most important part in the digital content, especially the video contents. As shown by experimental results, the proposed method can reduce the encryption time and can improve the protection strength using the traditional encryption algorithms for the digital contents.

## 1 Introduction

Recently, it is very easy that a great number of people share huge amounts of digital contents due to the improvement of the computer performance, the spread of high-speed communication, and the rapid increase of the Internet use. In accordance with this conditions, concerns about the security of digital contents has been increased, especially the digital video contents. And recent studies focus on the protection of digital contents. However it is not efficient that traditional encryption algorithms apply to the digital image/video contents, because the content has an amount of information<sup>1</sup> and the contents encryption spends long time. Therefore the major recent trend is to minimize the encryption time of the digital contents by the partial encryption that only parts of the contents are encrypted.

---

<sup>1</sup> The Video CD format compressed by the MPEG-I consists of 30 frames per second, and the DVD format compressed by the MPEG-II consists of 60 frames per second; the resolution of the Video CD is  $352 \times 240$ , and that of the DVD is  $720 \times 480$ . And the running time of most films is about 2 hours. Therefore the number of frames in the Video CD is about  $2.16 \times 10^5$ , and that in the DVD is about  $4.32 \times 10^5$ .



**Fig. 1.** Comparison between the traditional method and the proposed method: (a) traditional method, (b) proposed partial encryption method

**Table 1.** Classification of the previous partial encryption methods

Level of the feature complexity	What is encrypted?	Author [Paper]	Year
High ↑  ↓ Low	Quadtree structure	Cheng and Li [1]	2000
	Wavelet-Packet tree structure	Pommer and Uhl [13]	2003
	JBIG structure	Pfarrhofer and Uhl[12]	2005
	Bits that indicate the sign and magnitude of the non-zero DCT coefficients	Droogenbroeck and Benedett [2, 3]	2002
	Low frequency component	Suchindran and Nikolaos [4]	1999
	Filter permutation	Pommer and Uhl [5]	2001

Table 1 shows previous studies about the partial encryption method. We classify the methods according to the level of the feature complexity; the level depends on whether the features focus on the human or the computer.

For the partial encryption, Cheng and Li [1] find important parts using the Quadtree structure that is used to compress images. In the Quadtree structure, most of important parts are found in near the edge. Therefore this method is easy to find feature region and to control the range of the region. Pommer and Uhl [13] find the feature region using the wavelet-packet tree structure. The wavelet packet tree structure is also used to compressed image. In the wavelet packet decomposition the recursive procedure may be applied to all, the coarse scale approximations and the detail signals; it is the modified wavelet. In their paper, the encryption method is similar to the method using the Quadtree structure. This method provides confidentiality for retrieval-based visual communications. Pfarrhofer and Uhl [12] find the feature region

using the JBIG structure. The JBIG structure is for compressing binary images. In their paper, Pfarrhofer and Uhl select the higher layer and encrypt the selected layer. Therefore this method use small amount of the data, and simplify key management issues. Droogenbroeck and Benedett [2, 3] find bits that indicate the sign and magnitude of the non-zero DCT coefficients. And Suchindran and Nikolaos [4] find low-frequency components on the frequency domain. These two methods can improve the protection strength by encryption for some parts of the low-frequency components/the feature bit. Pommer and Uhl [5] use the filter permutation method. To encrypt, this method does not use the components in the image, but use the image processing filters. Therefore this method is easy to encrypt and the encryption time is very fast.

However, these features are still low-levels from the human point of view and do not have the semantic information, because previous studies extract the features for reducing the encryption time. In this paper, we proposed the partial encryption method using the face region as the feature because the face has the semantic information and is the most important part in the digital content, especially the video contents.

In this paper, we proposed the partial encryption method that uses the face region as the feature so that we encrypt only part of the video frame. The face region is the most important region for the human in the digital video. Therefore our proposed method can reduce the encryption time and can increase the protection strength by adding the location of the face region as the second key. To apply the proposed method to the digital video contents, we develop the DCS (Digital Contents Security) system, and we explain how to use this system on the Chapter 2.

## 2 DCS System

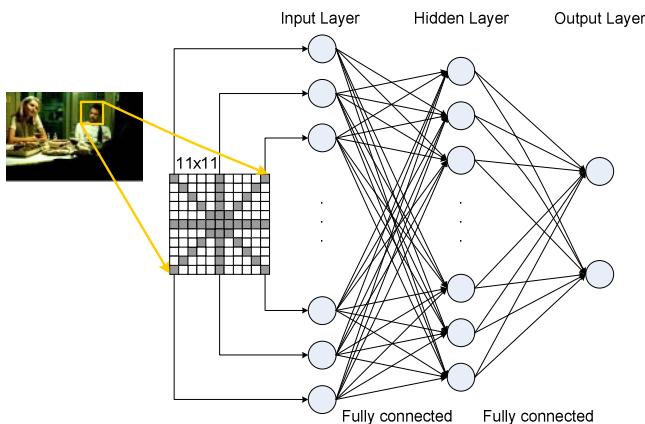
In the previous chapter, we classify features used for the partial encryption according to the complexity of the features. The previous studies consider the encryption time when they select the features. Therefore, the more the feature size is reduced, the less the encryption strength becomes weak, because the features do not have the semantic information from the point of view of human. Our proposed method applied to the DCS system increases the efficiency of the encryption algorithm and reduces the encryption time due to the partial encryption that encrypts only part of the digital video contents. And the parts are detected by the face detection algorithm in a frame of the digital video. The reason that we use the face region to encrypt the digital contents is that the face region is the most important part for the digital video contents; actors show their feelings and atmospheres of the contents via their faces.

### 2.1 Detection of the Face Region

To encrypt the video contents, the proposed system uses the face region, which is one of the digital video contents' features. For the more exact face detection, we use the MLP (Multi-Layer Perceptrons) to make a texture classifier, which discriminates between face pixels and non-face ones[6, 7], and the Gaussian skin-color model, discriminates between skin regions and non-skin regions[8]. Because we use two methods to detect face region, our proposed method detects more exact, but spends more computation time. However, our method reduces the feature size with exact

results, and reduces the encryption time. Therefore, there is very little difference in the whole time of detection and encryption.

For detecting the face region, we assume that the colors of faces are not different in the video frame. Under this assumption, the face detection process of our proposed system consists of two methods as follows. A received image is scanned by the MLP, which receives a given pixel and its neighbors within a small window; in this paper, we use the  $11 \times 11$  window<sup>2</sup>. The outputs of the MLP are combined into a face probability image, where each pixel's value is in the range  $[0, 1]$  and represents the probability that the corresponding input pixel is a part of face; one node is for face regions and the other is for non-face regions. If a pixel has a larger value than the given threshold value, we deem it to be a face pixel(Fig. 2).



**Fig. 2.** Face detection using the MLP

For more exact face detection, we use the Gaussian skin-color model. This model consists of two steps as follows:

1. a color-space selection used for face detection.
2. a skin color modeling.

For the first step, we use the normalized RGB which is easily obtained from the RGB values, reduce the space dimensionality, and are invariant to the brightness relatively to the light source as the color-space, and can be computed as follows:

$$r = \frac{R}{R + G + B}, g = \frac{G}{R + G + B}, b = \frac{B}{R + G + B}. \quad (1)$$

Here, the normalized RGB uses only  $r$  and  $g$ . And for the second step, we use Bayesian classifier to construct the skin color modeling, and can be expressed as follows:

---

<sup>2</sup> In this paper, we use horizontal, vertical and diagonal features(41 features), instead of rectangular features(121 features) for neural network, due to this method using 41 features is efficient for the texture recognition[11].

$$P(\text{skin} \mid \mathbf{c}) = \frac{P(\mathbf{c} \mid \text{skin})P(\text{skin})}{P(\mathbf{c})}. \quad (2)$$

Where  $\mathbf{c}$  is input vector( $r, g$ ),  $P(\text{skin} \mid \mathbf{c})$  is the posterior probability and  $P(\mathbf{c} \mid \text{skin})$  is the likelihood. In this system, we use the maximum likelihood estimation to classify the face region, because we do not know the posterior probability. When constructing the skin color modeling using Bayesian classifier, there are two assumptions. First, when the probability that a pixel is skin of the color-space is assumed to be the same as the probability that a pixel is not skin ( $P(\text{skin})=P(\sim \text{skin})$ ), we construct the modeling using only  $P(\mathbf{c} \mid \text{skin})$ . Second, when the probability that a pixel is skin and the probability that a pixel is not skin are not a same, we classify a pixel between a skin and non-skin using the ratio found from Eq. (3).

$$\frac{P(\mathbf{c} \mid \text{skin})}{P(\mathbf{c} \mid \sim \text{skin})} > \frac{1 - P(\text{skin})}{P(\text{skin})} \quad (3)$$

We assume that the probability that a pixel is skin and the probability that a pixel is not skin are a same, model the skin-color distribution using only  $P(\mathbf{c} \mid \text{skin})$ , and use a Gaussian probability density function as follows:

$$P(\mathbf{c} \mid \text{skin}) = \frac{1}{2\pi|\Sigma|^{1/2}} e^{-\frac{1}{2}(\mathbf{c}-\mu)^T \Sigma^{-1} (\mathbf{c}-\mu)} \quad (4)$$

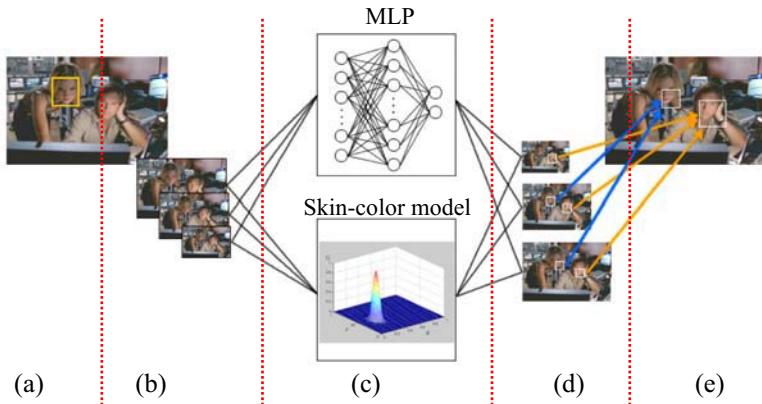
Where  $\mu$  and  $\Sigma$  are a mean vector and a covariance matrix as parameters of the skin-color distribution respectively. Table 2 shows the parameters used in our implementation.

**Table 2.** The parameters value of our method

	$\mu_r$	$\mu_g$	$\sigma_{rr}$	$\sigma_{rg}$	$\sigma_{gr}$	$\sigma_{gg}$
Value	0.3844	0.3193	0.00025	-0.00013	-0.00013	0.0002

When a frame of the digital video contents is inputted, the DCS system outputs the result through the five steps as follows:

1. determining the basis size of the face region by the users(Fig. 3(a)).
2. resizing the frame size based on the basis face size specified on step 1 and sending three outputs to step 3(Fig. 3(b)).
3. detecting the face regions using the MLP and the skin color-model in the 3 images respectively. If the locations of results by the MLP and that of results by the skin-color model overlap, this system determines that the location is the face region(Fig. 3(c, d)).
4. Finally, determining the face region using the 3 results from step 3(Fig. 3(e)).



**Fig. 3.** The face detection using the MLP and the skin-color model: (a) determining the basis face size on the input image, (b) resized images, (c) face detection using the MLP and skin-color model, (d) results of (b), (e) the final result

In the first step, the reason for the determination of the basis size is to detect meaningful face region; if the face size is too small or too large, most of the face will be the meaningless region. And to detect various sizes of face regions, we make three images in the second step; one is the resized frame, another is the 30% enlarged image of the first image, and another is the 30% reduced image of the first image.

## 2.2 Encryption and Decryption

After the face detection, the proposed system encrypts the detected face regions of the frame and hides the location information in the frame using a watermarking method. The reason of location information hiding is that improve the protection strength for the contents.

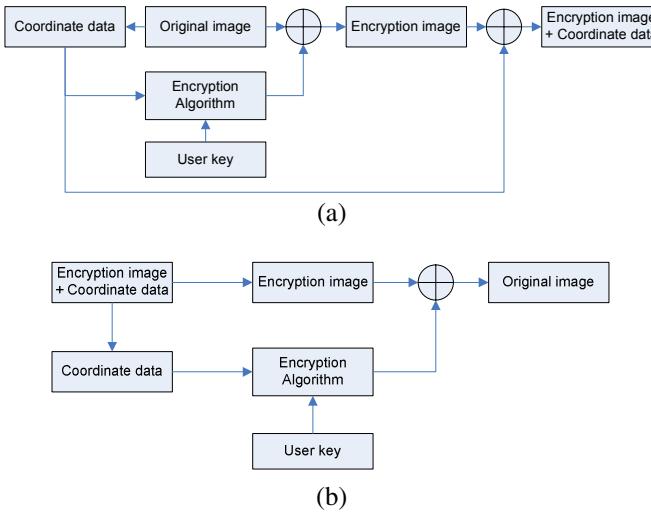
In this paper, because we encrypt the contents using the block encryption algorithms such as the DES (Data Encryption Standard) [9] and the AES (Advanced Encryption Standard) [10], we make a rectangle of the face region and use the x and y location of the top-left and the bottom-right position of the detected face region. The DES and the AES are the archetypal block encryption algorithm that takes a fixed-length string of plaintext bits and transforms it through a series of complicated operations into another cipher text bit string of the same length.

In the proposed system, the encryption process is different from the decryption process. The encryption process consists of three steps as follows(Fig. 4(a)):

1. the face regions detection.
2. the detected region encryption in the content with the key.
3. the location information hiding of the encrypted region using a watermarking algorithm.

On the other hand, the decryption process consists of two steps(Fig. 4(b)):

1. extracting the location information of the encrypted face region.
2. the face region decryption.



**Fig. 4.** The flowchart of the DCS system: (a) encryption, (b) decryption

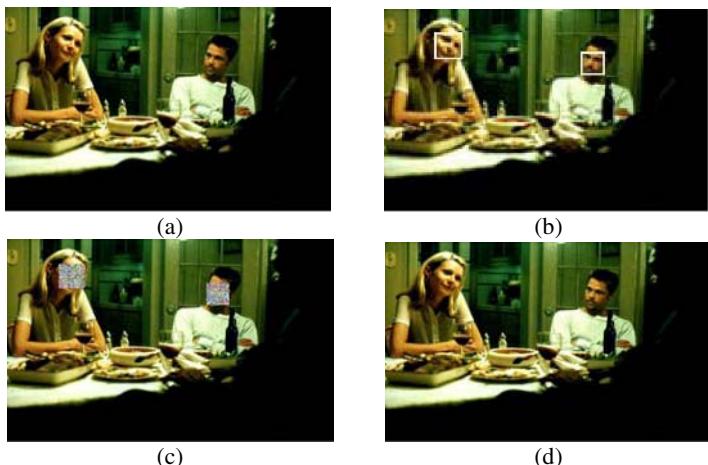
### 3 Experimental Results

The experiment system consists of the CPU Intel® Pentium4 3.0GHz, and the graphic card ATI® X1600. The resolution of the input video frame is 720×480 pixels. The run-time of the input video is 3 minutes and the number of frames per second is 30. The experimental system is implemented using Microsoft® Visual C++ 6.0, DirectX 9.0 SDK.

In this system, we use the MLP to detect face regions. The MLP consists of three layers: the input layer, the output layer, and the hidden layer; the number of nodes in the input layer, the output layer, and the hidden layer are 41, 2, and 33 respectively. And we use the  $11 \times 11$  texture model, which obtained by the R and G value in the normalized RGB model of the input image, as the input value for the input layer.

Fig. 5 shows the results of the encryption and the decryption using the DCS system. When an image is inputted this system, we detect the feature region from the input image (Fig. 5(a, b)). The feature region is the important part such as the actor's face, text, objects, etc. And we use the face region as the feature region in the proposed system. For the face detection, we use the MLP that this system learn for detecting face regions, and apply the skin-color model to the image. Through the AND operation between the results of the MLP and that of the skin-color model, we can detect face regions more exactly. After the face detection, we encrypt the feature region using the location of detected face regions (Fig. 5(c)). For the contents decryption, we extract the location information and decrypt the contents through the inserting the location information and the user key (Fig. 5(d)).

In the proposed system, because the encryption time is different from the decryption time, we measured the encryption time and the decryption time using the DES



**Fig. 5.** The digital video contents protection using the DCS system: (a) input image, (b) the result of the face detection step, (c) partial encryption with the face region, (d) decryption of (c)

**Table 3.** The comparison the encryption times per frame between the proposed method and the traditional method

Encryption algorithm	DES		AES	
	Find features	Encryption	Find features	Encryption
Fully encryption method	-	421ms	-	622ms
Proposed method	217ms	27ms	217ms	40ms

**Table 4.** The comparison the decryption times per frame between the proposed method and the traditional method

Encryption algorithm	DES	AES
Fully encryption method	421ms	622ms
Proposed method	27ms	40ms

and AES algorithm as shown in the table 3 and table 4. Because the encryption time includes the face detection time but the decryption time does not include another process time, the decryption time is faster than the encryption time.

In this experiment, there are two face regions and the ratio of the feature regions to the input image is 0.06<sup>3</sup>. If the ratio becomes smaller, the feature find time and the total encryption time will be increased. However the encryption time does not affect the contents users. As shown in the table 4, fully encryption methods provide 2 or 3 frames in a second, but the proposed method provides 25 or 30 frames in a second. Therefore it is applicable to the video contents encryption.

<sup>3</sup> The size of a digital video contents frame is 720×480 pixels, and the size of face regions are 85×85 pixels and 123×123 pixels respectively. Therefore the ratio is 0.06.

## 4 Conclusion

In this paper, we proposed the partial encryption method that use face region and developed the DCS system that applied the proposed method. To detect the face regions more exactly, we use two methods: the MLP and the skin-color model.

Our proposed method offers some advantages: reducing the encryption time and increasing the protection strength by using the meaningful face regions. Therefore our method is applicable to the digital video contents such as the DVD, the HDTV, films, and the satellite image transport.

Our system reduces the encryption time, but this is not able to provide real-time encryption yet. These problems are caused by the face detection and the block encryption algorithms such as the DES and AES. To solve these problems, we will study about improvement of the face detection algorithm and use of the stream encryption algorithm.

**Acknowledgement.** This work was supported by the Soongsil University Research Fund.

## References

1. Cheng, L., Li, X.: Partial Encryption of Compressed Images and Video. IEEE Transactions on Signal Processing, Vol. 48, No. 8. (2000) 2439-2451
2. Van Droogenbroeck, M., Benedett, R.: Techniques for a Selective Encryption of Uncompressed and Compressed Images. ACIVS, Ghent, Belgium (2002) 90-97
3. Van Droogenbroeck, M.: Partial encryption of images for real-time applications. IEEE Benelux Signal Processing, Hilvarenbeek, Netherlands (2004) 11-15
4. Suchindran S. M. and Nikolaos G. B.: SCAN based lossless image compression and encryption. IEEE Int. Conference on Information Intelligence and Systems (1999) 490-499
5. Pommer, A., Uhl, A.: Wavelet packet methods for multimedia compression and encryption. IEEE Pacific Rim Conference on Communications, Computers and Signal Processing, Victoria, Canada (2001) 1-4
6. Keechul, J.: Neural Network-based Text Localization in Color Images. Pattern Recognition Letter, Vol. 22, No. 14. (2001) 1503-1515
7. Keechul, J., JungHyun, H.: Hybrid Approach to Efficient Text Extraction in Complex Color Images. Pattern Recognition Letter, Vol. 25, Issue 6. (2004) 679-699
8. Zarit, B. D., Super, B. J., Quek, F. K. H.: Comparison of Five Color Models in Skin Pixel Classification. International Workshop on Recognition, Analysis and Tracking of Face and Gestures in Real-Time System (1999) 58-63
9. Ehksam et al.: Product Block Cipher System for Data Security. U.S. Patent 3 (1975) 962-539
10. Daemen, J., Rijmen, V.: The Design of Rijndael: AES - the advanced encryption standard. Springer Verlag (2002)
11. Jain, A.K., Karu, K.: Learning Texture Discrimination Masks. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 18, Issue 2. (1996) 195-205
12. Pfarrhofer P., Uhl A.: Selective Image Encryption Using JBIG. LNCS Vol. 3677 (2005) 98-107
13. Pommer A., Uhl A.: Selective encryption of wavelet-packet encoded image data: efficiency and security. Multimedia Systems, Vol. 9, No. 3 (2003) 279-287

# Relevance Feedback Using Adaptive Clustering for Region Based Image Similarity Retrieval

Deok-Hwan Kim<sup>1,\*</sup> and Seok-Lyong Lee<sup>2,\*\*</sup>

<sup>1</sup> School of Electronics and Electrical Engineering,  
Inha University

253 Yonghyun-dong, Nam-gu, Incheon 402-751, Korea  
[deokhwan@inha.ac.kr](mailto:deokhwan@inha.ac.kr)

<sup>2</sup> School of Industrial and Information Systems Engineering,  
Hankuk University of Foreign Studies  
89, Wangsan-ri Mohyun, Yongin-si, Kyounggi-do, Korea  
[s1lee@hufs.ac.kr](mailto:s1lee@hufs.ac.kr)

**Abstract.** In this paper, we propose a novel relevance feedback approach using adaptive clustering based on region representation. Performance of content based image retrieval system is usually very low because of the semantic gap between the low level feature representation and the user's high level concept in a query image. Semantically relevant images may exhibit very different visual characteristics, and may be scattered in several clusters. Our main goal is finding semantically related clusters to reduce this semantic gap. Our method consists of region based clustering process and cluster-merging process. All segmented regions of relevant images are grouped into semantically related clusters, and clusters are merged by estimating the number of the clusters. We form representatives of clusters as the optimal query. A region based image similarity measure is used to calculate the distance between the multipoint optimal query and an image in the database. Experiments have demonstrated that the proposed approach is effective in improving the performance of image similarity retrieval system.

**Keywords:** Region Based Image Retrieval, Image Segmentation, Relevance Feedback, Clustering.

## 1 Introduction

With the advances in computer vision, information retrieval, and database management, there have been extensive studies on relevance feedback based approach to content-based image retrieval (CBIR). Content-based image retrieval systems use the visual contents of images, such as color histogram, color layout, and texture features to represent and index images as vectors in the feature space. Each image can be viewed as a data point in the feature space. Similarly, a query

---

\* This work was supported by INHA UNIVERSITY Research Grant (INHA-34436).

\*\* Corresponding author.

image is mapped to a query point in the same feature space. Then, k-nearest neighbor query is accomplished by retrieving the images corresponding to the k data points nearest to the query point. That is, the closer two vectors are, the more similar the corresponding images are.

However, performance of retrieval system is usually very low because the low level feature representation cannot capture the user's high level concept in a query image. To solve this problem, recent studies have focused on two approaches separately: (1) region based image retrieval; (2) relevance feedback processing in CBIR.

However, little research has been made to combine region-based image retrieval and relevance feedback methods together. First, region based image retrieval systems [10] attempt to overcome the deficiencies of low level feature representation such as color histogram and color layout by representing images at the object level. A region based retrieval system applies image segmentation to decompose an image into regions.

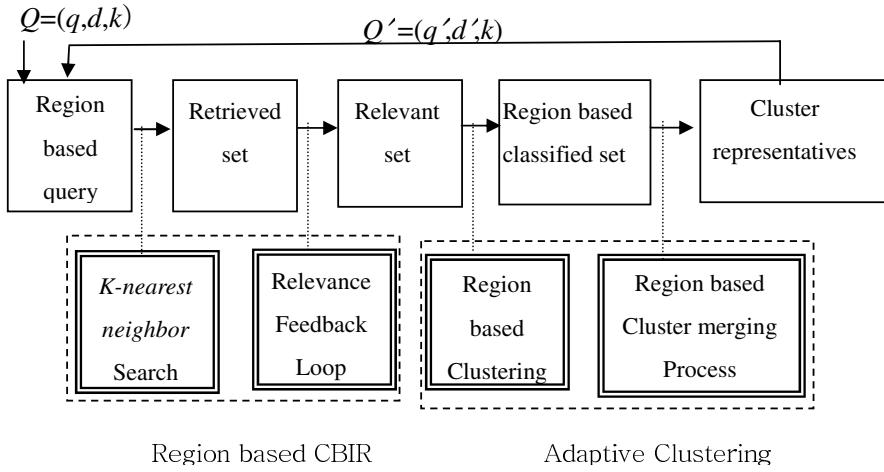
Second, a relevance feedback can reduce the semantic gap between low level visual features and high level concepts conveyed by the query images. In each round, the user evaluates the result by picking up relevant images within the set of images retrieved in the last round. The system then utilizes this feedback to refine the current query and to improve its retrieval results in the next round. This process is repeated until the user is satisfied with the results. Early studies in relevance feedback [1][2] represent a new query as a single point in a feature space and change the weights of feature components to find an optimal query point and an optimal distance function. In each round of feedback, the weighted average of the relevant images is used as the new query point in the next iteration. Meanwhile, a study [3] to the multipoint query groups the relevant points according to user feedback into clusters and treats representatives of clusters as a new query. Recently, Jing et al. [6] [7] try to combine the query-point movement and multipoint query with region based image retrieval. However, their region clustering may not find semantically related clusters since they don't consider cluster merging among nearest clusters.

The similarity between images perceived by humans does not necessarily correlate with the distance between them in the feature space. That is, semantically relevant images may exhibit very different visual characteristics, and may be scattered in several clusters rather than one. From this point of view, the problem of reducing the semantic gap becomes that of finding semantically related clusters. In this paper, we propose a new relevance feedback approach using adaptive clustering, which groups all segmented regions of relevant images and merges certain clusters by determining the number of clusters, for region based image retrieval.

The remainder of the paper is organized as follows: In Section 2, we describe the overall process of region based image retrieval. In Section 3, the region based similarity measure and the general algorithm is explained. In Section 4, we describe proposed adaptive clustering method. Section 5 contains the result of our experiments. Finally, Section 6 summarizes our work.

## 2 Region Based Image Retrieval with Relevance Feedback Approach

Fig. 1 shows the proposed region based CBIR with relevance feedback mechanism. In the preprocessing step, an image is decomposed into several regions and feature vectors extracted from regions are stored in the database. At the first stage, an example image submitted by the user is parsed to generate an initial query  $Q = (q, d, k)$ , where  $q$  consists of several query points in the feature space because an example image is decomposed into multiple regions,  $k$  is the number of images in the query result returned by the system, and  $d$  is the distance function. The Earth Mover's Distance(EMD) is used to measure the distance between two images[5]. The query point  $q$  is compared with images in the database using the distance function  $d$ . According to  $d$ , the result set consisting of  $k$  images close to  $q$ ,  $Result(Q) = \{p_{1_1}, \dots, p_{1_n}, \dots, p_{k_1}, \dots, p_{k_n}\}$ , is returned to the user.  $p_{k_n}$  is the  $n$ th region of  $k$ th image since each image consists of multiple regions.



**Fig. 1.** Region-Based CBIR with Relevance Feedback Process

At the next stage, the user evaluates the relevance of images in  $Result(Q)$  by assigning a relevance score,  $v$ , to each of them. Based on those scores, the relevant set consisting of relevant images,  $Relevant(Q) = \{p'_{1_1}, p'_{1_2}, \dots, p'_{m_n}\}$ , is obtained.  $p'_{m_n}$  is the  $n$ th region of  $m$ th relevant image. That is,  $Relevant(Q)$  includes newly added relevant images and relevant images at previous iterations. Note that newly added relevant images reflect the user's query concept more precisely. To consider this, we put more emphasis on these images than prior relevant images by assigning them larger weights. In this paper, we present a new adaptive clustering method consisting of two processes: the region based adaptive clustering process and the cluster-merging process. As more relevant images are available, the number of regions in the query increases rapidly. The

time required calculating EMD distance between the query and an image is proportional to the number of regions in the query. To reduce retrieval speed of the system, the region based clustering process merges similar regions in the relevant set,  $Relevant(Q)$ . The hierarchical clustering algorithm is adopted to group the regions in  $Relevant(Q)$  into a few clusters, each of which corresponds to a new region of the next query. Then, the proposed cluster-merging process determines the number of clusters and merge certain clusters at the same level to reduce the number of query points at the next iteration. Finally, representatives of clusters generated from regions of relevant images in the classified set make up the set of new query points. A new query,  $Q' = (q', d', k)$  with a set of new query points  $q'$  and a new distance function  $d'$  to reflect new adjusted weights, is computed and then used as an input for the second round.

After some iterations, the loop ends up with the final result set when the user is satisfied with the result.

In this paper, normalized cut segmentation method [9] is used to decompose an image into regions, which identify objects in an image. That is, an image is represented as multiple regions including object. Our approach to the relevance feedback allows multiple objects (regions) to be a query. We refer to them as a multipoint query. When the user marks several points as relevant, we cluster sets of relevant points and choose the centroids of the clusters as their representatives. Then, we construct the multipoint query using a small number of good representative points. At the region-based adaptive clustering process, hierarchical tree structure is organized and an adaptive clustering technique is used to improve the retrieval efficiency. Since prior relevant images can be kept unchanged as new ones are added, statistics such as mean and covariance of each cluster, which were computed from the previous iteration, can be utilized to further accelerate the current clustering process. At the cluster-merging process, Hotelling's  $T^2$  [4] is used to merge any pair of clusters in arbitrary shapes and find the number of clusters for a given query.

### 3 Similarity Measure and General Algorithm

#### 3.1 Region Based Image Similarity Measure

To retrieve similar images for a query image, Earth Mover's Distance(EMD)[5] is used to measure distance between two images based on region representation. It is suitable for the region based image similarity measure since it can operate on variable length feature representations of distributions.

When a user marks several images as relevant ones at each iteration of the relevance feedback, we cluster a set of relevant points and choose the centroid of the cluster as its representative. Similar regions are merged together via clustering and we form composite images as the multipoint optimal query. That is, a signature in EMD is an composite image with all regions coresponding to clusters. The Euclidean distance is used to measure the ground distance between two regions and total weight of each signature is 1. We use the region size as the weight of the signature.

### 3.2 A Region Based Image Retrieval Algorithm

We propose a novel relevance feedback approach for multipoint queries using the region based adaptive clustering and cluster-merging method. The algorithm of our relevance feedback for multipoint queries is as follows:

**Algorithm 1.**  $k$ -nearest neighbor search for region based image retrieval

**input:** a query example

**output:**  $k$  retrieved images

**begin**

**Step1:** Initialization

(1) Set  $Q = (q, d, k)$  using an initial query image.

**Step2:** Retrieval of images

(1) Retrieve  $k$  images  $Result(Q) = \{p_1, p_2, \dots, p_k\}$  such that  $emd(p_1, Q)$  is the lowest,  $emd(p_2, Q)$  is the next lowest, and so on.

**Step3:** User Interaction

For each image  $p_j$  in  $Result(Q)$ ,

(1) recommend image  $p_j$  to the user.

(2) If image  $p_j$  is marked as relevant, then  $Relevant(Q) = Relevant_{previous}(Q) \cup \{p_j\}$ .

**EndFor**

**Step4:** Hierarchical Clustering or adaptive region clustering

**If initial iteration**

(1) perform hierarchical clustering using all relevant regions in  $Relevant(Q)$

**Else**

(2) perform adaptive clustering by using statistics in previous iteration

**Step5:** Region based Cluster-merging

(1) For each cluster  $C_i$

calculate a centroid, its covariance, and its weight.

(2) determine the number of clusters  $C_1, \dots, C_g$  by using Hotelling's  $T^2$ .

(3) Representatives of clusters make up the set of new query points  $q'$ .

**Step6:** Query refinement and distance function update

(1) Refine  $Q' = (q', d', k)$  with a set of new query points  $q'$  and a new distance function  $d'$  to reflect new adjusted weights [7].

(2) Goto Step2 for the next iteration

**end**

## 4 Adaptive Clustering Method

In this section, we propose basic ideas to find semantically related clusters, whose representatives are used as query points at the next iteration of region based image retrieval.

### 4.1 Region Based Adaptive Clustering Stage

A hierarchical clustering technique is used to organize the relevant images into a hierarchical tree structure. At first iteration, initial clusters of all regions of the relevant images form the basis of the hierarchy. The level  $g$  at the hierarchy of clusters corresponds to  $g$  clusters.

The clustering algorithm groups data into hyperspherical regions. Once initial clusters are obtained, we calculate a mean vector  $\bar{x}$ , a weighted covariance matrix  $S$ . The mean vector determines the location of the hyperellipsoid, while the covariance matrix characterizes its shape and orientation. The weight of each cluster compared with the others is determined by the sum of relevance score values of points in each cluster, that is,  $m = \sum_{k=1}^n v_k$ .

## 4.2 Cluster-Merging Stage

The clusters after the region based clustering stage can be further merged into bigger clusters. Given  $g$  clusters, our cluster-merging algorithm finds candidate pairs of clusters to be merged. Two clusters most likely to be merged should be “close” enough. The algorithm selects the next pair of clusters to be merged until the number of the latent clusters is found.

For this purpose, we compare their mean vectors. We infer the merge of the two clusters statistically from the closeness of two mean vectors  $\bar{x}_i$  and  $\bar{x}_j$ . For the statistical test, let us define:

- the points of  $i$ th cluster,  $x_{i1}, x_{i2}, \dots, x_{in_i}$ , to be a random sample of size  $n_i$  from a population with a mean vector  $\mu_i$  and a covariance matrix  $\Sigma_i$ .
- the points of  $j$ th cluster,  $x_{j1}, x_{j2}, \dots, x_{jn_j}$ , to be a random sample of size  $n_j$  from a population with the mean vector  $\mu_j$  and a covariance matrix  $\Sigma_j$ .
- $x_{i1}, x_{i2}, \dots, x_{in_i}$  to be independent of  $x_{j1}, x_{j2}, \dots, x_{jn_j}$ .

We use a pooled covariance to estimate the common covariance since we assume that the population covariances for the two clusters are nearly equal.

When two clusters are characterized by the mean vector,  $\bar{x}_i, \bar{x}_j \in \mathbb{R}^p$ , covariance matrix,  $S_i, S_j$ , the number of elements in the cluster,  $n_i, n_j$ , and the weight of the cluster,  $m_i, m_j$ , respectively, Hotelling’s  $T^2$  is used to decide whether the locations of two clusters are equal or not. For  $C_i$  and  $C_j$  clusters with  $i \neq j$ , it is defined by

$$T^2 = \frac{m_i m_j}{m_i + m_j} (\bar{x}_i - \bar{x}_j)' S_{pooled}^{-1} (\bar{x}_i - \bar{x}_j), \quad (1)$$

where

$$S_{pooled} = \frac{1}{m_i + m_j} \left( \sum_{k=1}^{n_i} v_{ik} (x_{ik} - \bar{x}_i) (x_{ik} - \bar{x}_i)' + \sum_{k=1}^{n_j} v_{jk} (x_{jk} - \bar{x}_j) (x_{jk} - \bar{x}_j)' \right). \quad (2)$$

The usual hypothesis to test the location difference is as follows:

$$H_0 : \mu_i = \mu_j \text{ and } H_1 : \mu_i \neq \mu_j,$$

where  $\mu_i$  is the unknown true center of  $C_i$  for  $i = 1, \dots, g$ . If  $T^2$  is too big which happens when  $\bar{x}_i$  is “too far” from  $\bar{x}_j$ , then the null hypothesis  $H_0$  is rejected. Note that  $T^2 \approx \frac{p(m_i + m_j - 2)}{m_i + m_j - p - 1} F_{p, m_i + m_j - p - 1}(\alpha)$  if  $H_0$  is true. Here  $F_{p, m_i + m_j - p - 1}(\alpha)$  is the upper  $100(1 - \alpha)$ th percentile of F-distribution with  $p$  and  $m_i + m_j - p - 1$  degrees of freedom. Therefore

$$\text{Reject } H_0 \text{ if } T^2 > c^2 \quad (3)$$

where  $c^2 = \frac{(m_i+m_j-2)p}{m_i+m_j-p-1} F_{p,m_i+m_j-p-1}(\alpha)$ . In other words, if  $T^2$  is larger than  $c^2$ , we conclude that the two clusters are separated.

When the data follow a multivariate normal distribution,  $(m_i + m_j - p - 1)/p(m_i + m_j + p - 2)T^2$  follows an  $F(p, m_i + m_j - p - 1)$ . This statistic can be interpreted as the Mahalanobis distance between the centers of two clusters. The proposed measure makes use of the sample covariance matrix and its inverse.

To start finding the number of clusters, let us consider two clustering levels with  $(g - 1)$  and  $g$  clusters. It is necessary to decide which level is more optimal than the other. At the  $g$ th clustering level, there are  $\binom{g}{2}$  of Hotelling's  $T^2$ 's used to decide which pair of clusters to be merged. If no significant merging occurs, then  $g$  number of clusters is closer than  $(g - 1)$  number of clusters. Otherwise  $(g - 1)$  number of clusters is closer to the optimal.

For efficient cluster-merging, we determine the parameters of the merged clusters from those of existing clusters instead of those of points in existing clusters. We characterize a new cluster created by combining clusters  $C_i$  and  $C_j$  with the following statistics [4]:

$$m_{new} = m_i + m_j \quad (4)$$

$$\bar{x}_{new} = \frac{m_i}{m_{new}}\bar{x}_i + \frac{m_j}{m_{new}}\bar{x}_j \quad (5)$$

$$S_{new} = \frac{m_i - 1}{m_{new-1}}S_i + \frac{m_j - 1}{m_{new-1}}S_j + \frac{m_im_j}{m_{new}(m_{new} - 1)}[(\bar{x}_i - \bar{x}_j)(\bar{x}_i - \bar{x}_j)'] \quad (6)$$

#### **Algorithm 2.** Cluster Merging

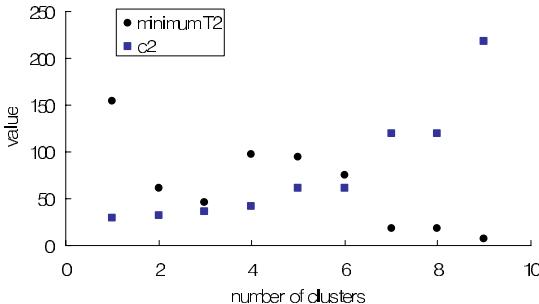
**begin**

1. Compute  $T^2$  values and their  $c^2$  values for all pairs of clusters
2. Select candidate pair of clusters having minimum value of  $T^2$  values
3. **While** minimum  $T^2$  value  $\leq$  their  $c^2$
4. merge candidate pair of clusters
5. calculate  $m_{new}, \bar{x}_{new}, S_{new}$  using Equation (4), (5), (6)
6. select next candidate pair of clusters
7. **EndWhile**

**end**

### 4.3 Example of Cluster-Merging Procedure

Fig. 2 shows the cluster-merging procedure for 9 clusters. In order to test closeness of any pair of clusters, given a proper  $\alpha(0.01)$ , the minimum value of  $T^2$  values for all pairs of clusters and  $c^2$  value for corresponding clusters are calculated. If minimum  $T^2$  value is less than  $c^2$  value, then two clusters are close enough and merge them. When 9 clusters are given at certain iteration, among all possible pairs of clusters, we find a pair of clusters with minimum  $T^2$ . The pair of clusters with minimum  $T^2$  is the candidate to be merged and their  $c^2$  is calculated for the given  $\alpha$ . When  $T^2$  is less than  $c^2$ , they are merged into one. Then the number of clusters is reduced by one. This process will be applied until minimum  $T^2$  and  $c^2$  are reversed. Then the number of adjusted clusters becomes 6. That is, we can adjust the number of clusters to be merged by selecting a proper significance level  $\alpha$ .

**Fig. 2.** scatter plot

## 5 Experiment and Results

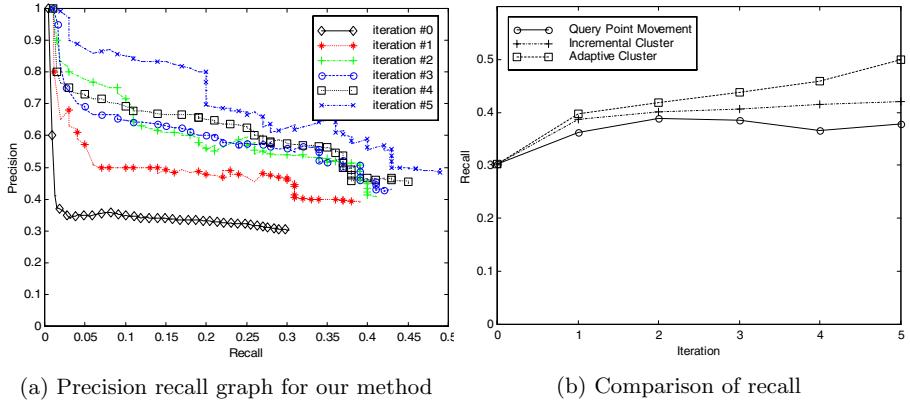
Our experiments have been conducted for evaluating relevance feedback using adaptive clustering approach for multi-point  $k$ -NN query in region based image database. To evaluate the performance of our system, we compare our method with those used in [6] [7], which use region based relevance feedback and single query point movement and incremental clustering, respectively. For incremental clustering approach, we initialize the number of clusters 2 and increases by one at each iteration. The algorithm has been implemented on a Pentium IV 3.20Ghz PC using Windows 2003 operating system.

For experimental studies, the Corel image collection is used as the test set of data. The image collection includes 10,000 color images. Its images have been classified into distinct categories (e.g., airplane, flower, sunset, mountain, pyramid etc) by domain professionals and there are about 100 images in each category. In the experiments, we use high level category information as the ground truth to obtain the relevance feedback since the user wants to retrieve the images based on high level concepts, not low level feature representations. That is, images from the same category are considered relevant.

In this experiment, the segmentation method we use is normalized cuts algorithm [9]. Segmentation of an image renders a partition of the image into distinct regions. We use 5 major regions per each image. Features are extracted from each of the individual regions to characterize the objects implied by those regions. We use two features to describe a region: color moment and region size. The first three color moment is extracted from each channel of  $L^*a^*b$  color space. Region size is the area of the region normalized by the size of the image and it describes the importance of the region.

We generate 100 random initial queries from the test data set and evaluate the retrieval quality for a sequence of iterations starting with these initial queries. We perform five feedback iterations in addition to the initial query. All the measurements are averaged over 100 queries. The  $k$ -NN query is used to accomplish the similarity-based match and we set  $k$  to 100.

Fig. 3(a) shows the precision-recall graph for our method. In this graph, one line is plotted per iteration. Each line is drawn with 100 points, each of which



(a) Precision recall graph for our method

(b) Comparison of recall

**Fig. 3.** Performance evaluation**Fig. 4.** 10 result images after 1st iteration**Fig. 5.** 10 result images after 5th iteration

shows precision and recall as the number of retrieved images increases from 1 to 100. This figure shows that the retrieval quality improves at each iteration.

Fig. 3(b) compares the recall for adaptive clustering, incremental clustering and query point movement [6] at each iteration, respectively. This figure shows that the recall of our method increase at each iteration and outperform those of the incremental clustering and the query point movement approach since our method can find semantically related clusters better than those methods. Fig. 4 and 5 show the result of a sample query at the airplane category at 1st and 5th iteration, respectively. The query image appears in the upper leftmost side. Due to the limitation of space, we show only two rows of images with k-nearest neighbor search( $k=100$ ). The result at 5th iteration shows better performance.

## 6 Conclusion

We have presented a novel relevance feedback method using adaptive clustering based on region representation.

Experimental results on a database of 10,000 general-purpose images show the better effectiveness of our method over others. It demonstrates that the clustering efficiency can be improved by using a new cluster-merging algorithm. For future work, we will integrate kernel-based learning algorithms into our method.

## References

1. Y. Ishikawa, R. Subramanya, C. Faloutsos. MindReader: Querying databases through multiple examples. Proceedings of the 24th VLDB Conference, 218-227, New York, USA, 1998.
2. Y. Rui, T. Huang, S. Mehrotra. Content-based image retrieval with relevance feedback in MARS. Proceedings of IEEE International Conference on Image Processing '97, Santa Barbara, CA, October 1997.
3. K. Pornaew, K. Chakrabarti. Query Refinement for Multimedia Similarity Retrieval in MARS. Proceedings of the 7th ACM Multimedia Conference, 235-238, Orlando, Florida, 1999.
4. R.A. Johnson, D.W. Wichern. Applied Multivariate Statistical Analysis. Prentice-Hall, N.J., 1998.
5. Y. Rubner, L.J. Guibas, C. Tomasi. The earth mover's distance, Shimutil-dimensional scaling, and color-based image retrieval. Proc. ARPA Image Understanding Workshop, 661-668, New Orelans, LA, May 1997.
6. F. Jing, M. Li, H. J. Zhang, B. Zhang. Region-Based Relevance Feedback In Image Retrieval. Proc. IEEE Int. Symp. Circuits and Systems, vol. 4, 145-148, 2002.
7. F. Jing, M. Li, H. J. Zhang, B. Zhang. Relevance feedback in region-based image retrieval. IEEE Transactions on Circuits and Systems for Video Technology, 14(5), 672-681, May 2004.
8. R. D. Duda, P. E. Hart, D. G. Stork. Pattern Classification. John Wiley & Sons, Inc., New York, 2001.
9. J. Shi and J. Malik. Normalized Cuts and Image Segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(8), 888-905, August 2000.
10. Z. H. Zhou, M. L. Zhang, K. J. Chen. A Novel Bag Generator for Image Database Retrieval with Multi-Instance Learning. Proceedings of the 15th IEEE International Conference on Tools with Artificial Intelligence, 565-569, 2003.

# Learning and Evolution Affected by Spatial Structure

Masahiro Ono and Mitsuru Ishizuka

<sup>1</sup>Graduate School of Information Science and Technology,  
The University of Tokyo,  
7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan  
[{mono, ishizuka}@mi.ci.i.u-tokyo.ac.jp](mailto:{mono, ishizuka}@mi.ci.i.u-tokyo.ac.jp)

**Abstract.** In this study, we explore the roles of learning and evolution in a non-cooperative autonomous system through a spatial IPD (Iterated Prisoner’s Dilemma) game. First, we propose a new agent model playing the IPD game; the game has a gene of the coded parameters of reinforcement learning. The agents evolve and learn during the course of the game. Second, we report an empirical study. In our simulation, we observe that the spatial structure affects learning and evolution. Learning is not effective for achieving mutual cooperation except under certain special conditions. The learning process depends on the spatial structure.

**Keywords:** game theory, prisoner’s dilemma, small world network.

## 1 Introduction

From the scientific viewpoint, learning, evolution, and the interaction between them have been studied. One of the famous earlier works is the Baldwin effect [1], which hypothesized that the characteristics of how an individual learns affect the evolution of a species.

Moreover, these two concepts are worth studying from an engineering viewpoint. Recently, the autonomous agent has been studied, and there are systems that consist of many autonomous individuals, such as the multi-agent system. In general, agents adapt to the environment owing to learning in addition to their hard-coded program. These two functions correspond to learning and evolution (a part of the hard-coded program), as described earlier. In this area, the focus has mainly been on the manner in which the agents cooperate. However, the agents are not always cooperative, for example, in the case of agents that belong to different organizations and whose payoffs conflict. We forecast that there would be two types of systems—the cooperative and non-cooperative. Studies based on the abstract model can provide useful insights for many possible application systems both in the cooperative and non-cooperative cases.

One of the non-cooperative relations is PD (Prisoner’s Dilemma) from game theory. This game has been studied thus far because the game itself arouses our curiosity and is suitable for studying learning and evolution. Further, the evolutionary game theory pertains to evolution. In general, the equilibrium point

of the game has been studied in the field of game theory. The evolutionary game theory assumes an alternative equilibrium point ESS (Evolutionary Stable Strategy). In addition, numerical as well as the analytical approaches exist. One of the numerical approaches is that given by Lindgren [2]. His model expresses a meta-strategy that determines the subsequent moves based on the game history. The learning game has been studied in accordance with the development in the area of reinforcement learning. For example, it is shown that the reinforcement learning agents in the multi-agent system cause instability due to mutual learning [3]. There are a few studies that deal with both learning and evolution. Thus far, the combination of learning and evolution were adapted to the stochastic game [4] and Baldwin effect [5] under the simple learning condition.

With regard to the original concern of IPD (Iterated Prisoner's Dilemma), the basic issue pertains to the contradiction between the mathematical solution where the non-cooperative strategy is stable and the phenomenon in the real world, i.e., we often cooperate with each other. This issue has been studied for a long time, and a spatial structure was introduced in the evolutionary game theory. Players are located on the spatial structure, for example, a two-dimensional regular lattice [6][7] or a small-world network model [8][9][10]. They evolve in every generation after they play games across neighborhoods. In this case, the emergence of a cooperative strategy is observed after a period of dominance of the non-cooperative strategy. Thus far, investigation on the learning and evolution of the spatial structure through studies has been sufficient.

In this paper, we investigate the roles of learning and evolution in relation to the emergence of cooperation. In particular, we focus on the spatial structure that limits the communication among agents. This assumption is reasonable in the case of agents on the Internet. Our work may help design an agent to improve the system performance from an engineering viewpoint. In addition, it also contributes to science: the human characteristic to make a decision is affected by the spatial structure. In the subsequent sections, we first provide brief backgrounds of the Prisoner's Dilemma, mechanism of reinforcement learning, and small-world network model. We then describe a model that we have designed for combining the functions of learning and evolution. Experimental results are then presented and discussed. Finally, we present our conclusions and directions for future work.

## 2 Backgrounds

Here, we describe IPD, reinforcement learning, and the small-world network model. We consider learning and evolution in the IPD game. Reinforcement learning is the method of learning the selection of a move in the game. The small-world network is a model that expresses the spatial structure.

### 2.1 Iterated Prisoner's Dilemma

The Prisoner's Dilemma is the most popular game in game theory because it is an elegant model to express many social phenomena. The name and the typical payoff matrix of this game were given by Albert Tucker in the 1950s.

In a symmetric two-player game, the payoff matrix of Prisoner's Dilemma is expressed as Table 1, where R, T, S, and P represent the reward, temptation, sucker, and punishment payoffs, respectively. The payoff relations ( $T > R > P > S$ ,  $2R > T + S$ ) that exist among them raise a dilemma.

Both players in the game would select the defect strategy if they are assumed to be rational. Player 1 considers that he should defect and earn a higher payoff whenever player 2 cooperates or defects. Player 2 would also defect after the same consideration. In the end, each player defects, and (D, D) is the only Nash equilibrium in this game. However, this state is a pareto inferior and therefore it is not an optimal strategy for either player. Hence, this game poses a dilemma.

**Table 1.** The payoff matrix of Prisoner's Dilemma

		Player2	
		C (cooperate)	D (defect)
		R	T
Player 1	C	R	S
	D	T	P

$$(T > R > P > S, 2R > T + S)$$

## 2.2 Reinforcement Learning

Reinforcement learning is a type of machine learning. Assuming that an agent takes an action and receives a reward in return from the environment, the reinforcement learning algorithm attempts to find a policy for maximizing the agent's cumulative reward.

Essentially, an agent has inner states. The learning process is as follows: the agent selects an action in a state according to each value function, then the action in the state is evaluated and the value function is updated. The agent performs a state transition and selects the next action in the state. This process is repeated and the agent continues to refine the action-value function. Although there are several algorithms for reinforcement learning, in this study, we consider the one given by Sarsa [11].

## 2.3 Small-World Network

In 1998, Duncan Watts defined the small-world network using two characteristic parameters—characteristic *path length* and *clustering coefficient* [12].

The *characteristic path length L* is the average of the shortest path length between any two vertices on the network. The *clustering coefficient C* indicates the extent to which vertices adjacent to any vertex are adjacent to each other as an average.

If it is assumed that the number of vertices and edges on a graph are fixed, the structure changes as the randomness parameter  $p$  of the network varies. Each

vertex is connected to its neighborhoods mutually in the case of  $p = 0$ . The edges change stochastically as  $p$  increases. In case of  $p = 0$ , the network is referred to as a regular network and both  $L$  and  $C$  are large. On the other hand, a random network appears at  $p = 1$ , where  $L$  and  $C$  are small. In the middle, between these two extremes, the network exhibits the property of a small-world network, where  $L$  is small and  $C$  is large.

### 3 Proposed Model

We propose a new agent model that learns and evolves for the IPD game. The process by which the agent selects the next move is as follows:

We assume that the agent remembers the moves from the previous game in iterated games and he selects the next move in the manner of the first-order meta-strategy. The agent has four inner states (own previous move and opponent's previous move) = CC, CD, DC, and DD derived from the possible combinations of the moves from the previous game. In other words, the agent selects the next move in a state based on the previous game.

The agent has a gene in which two components are coded. One is the part that expresses the initial action-values for each of the possible combinations of actions and states. The other is the part for parameters of reinforcement learning. The created agent has initial action-values and a reinforcement learning function with parameters derived from the coded gene. He can play the IPD games and update the action-values. In other words, he can learn the game.

Primarily, when the agent selects a move in the game, he compares the action-values of each action in the states and picks up the greater one; as shown in Eqn. (1):

$$\text{next move} = \begin{cases} C, & Q(s, C) \geq Q(s, D) \\ D, & Q(s, C) < Q(s, D) \end{cases} \quad (1)$$

where  $Q(s, a)$  evaluates the action  $a$  in the state  $s$ , and  $s$  is a possible state. For example, in the state CC, the next move is C in the condition  $Q(CC, C) \geq Q(CC, D)$ . Since the agent has information on all the combinations of actions and states, all possible first-order meta-strategies can be expressed. The examples are listed in Table 2.

**Table 2.** First-order meta-strategy examples

PM	OPM	Strategy examples			
		All C	All D	TFT	Pavlov
C	C	C	D	C	C
C	D	C	D	D	D
D	C	C	D	C	D
D	D	C	D	D	C

PM: Previous Move, OPM: Opponent's Previous Move

The agent selects a move regardless of the previous moves in the case of all C and D. Tit-for-Tat (TFT) created by Anatol Rapoport is a famous strategy that repeats the opponent's previous move. This strategy won the famous iterated prisoner's dilemma tournament organized by Robert Axelrod in the year 1981 [13]. The Pavlov strategy [14]-also known as the "Win-Stay, Lose-Shift" strategy-selects the move opposite to the previous move when the agent cannot earn a high reward.

In return for the action, the agent receives a reward (the payoff of the game) and updates the action-values. The rule for updating the action-values is denoted by Eqn.(2).  $Q(s_t, a_t)$  evaluates the action  $a_t$  in the state  $s_t$  including the next action-value  $Q(s_{t+1}, a_{t+1})$ :

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha [r_{t+1} + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)] \quad (2)$$

where  $\alpha$  is the learning rate;  $r_{t+1}$ , a reward in return for the action  $a_t$ ; and  $\gamma$ , the discount rate. Here,  $a_{t+1}$  is determined by the  $\epsilon$ -greedy method in which the next move primarily depends on the action-values but is selected randomly according to a possibility  $\epsilon$ . The agent obeys the action-value completely; it selects a move in a deterministic manner in the case of  $\epsilon = 0$ . Otherwise, the agent selects randomly and the selection depends on  $\epsilon$  to some extent.

This process of updating is repeated during each selection. This implies that the agent learns through the iterated games. Since the strategy depends on the action-values as described before, the updating of the action-values indicates a change in the strategy, for example, from TFT to Pavlov. The characteristics of the learning process are the speed of learning, dependence on the reinforcement learning parameters, etc.

In this model, the eligibility trace parameter  $\lambda$  is also taken into account. The agent has a gene for these four parameters  $\alpha, \gamma, \lambda$ , and  $\epsilon$ .

## 4 Experimental Studies

The purpose of this experiment is to investigate the influence of spatial structure on learning and evolution. We carry out the experiment in two different spatial structure cases—a pool case and network case. Our simulation is performed as follows:

### 4.1 The Network Case

This is the case with a spatial structure that limits the communication among agents to some extent. The small-world network, in which randomness parameter  $p$  is given as an initial value and fixed during the simulation, is structured at first. The network consists of  $n$  agents, each of which has  $m$  links. These parameters are fixed at The agents act as the players of the IPD game. They play games with other agents having a direct connection on the network. A unit of a game is an IPD game that is iterated 100 times. This does not pose a problem as the agents do not employ backward induction because they cannot determine the end of

the IPD game in this simulation. It is assumed that a random noise involved in move selection by the agent is reversed according to a noise probability. The order in which the agents play games is determined randomly because the fixed order in each generation would influence the learning.

A generation change occurs after all the games in every generation end. A total of 20% of the agents are to die stochastically according to their gain earned in the generation. Then a new agent is to be located there in lieu of the dead agent; the new agent is generated as a copy of the agent that earned the most in the first neighborhoods of the dead agent. The mutation of the gene is generated at a copying probability of 0.02%. The payoff parameters in the simulation are set as  $(T, R, P, S) = (5, 3, 1, 0)$  in Table 1.

## 4.2 The Pool Case

Here, we assume a pool of agents without a spatial structure. In this case, the agents are not linked and they have the opportunity to meet any agent. Although the general procedure is similar to the network case as described before, there are two differences:

The first one determines the opponents they play games with. Since the spatial property is without a structure, there is a possibility that they might play a game with only one agent amongst all agents. However, an agent plays games  $m$  times with randomly selected agents in order to provide a chance for learning with an equivalent frequency for the network case.

The second one is related to the generation change. In this case, we adopt tournament selection (tournament size: 2), two-point crossover, and mutation. Essentially, the parameters in relation to the generation change are the same as that in the network case.

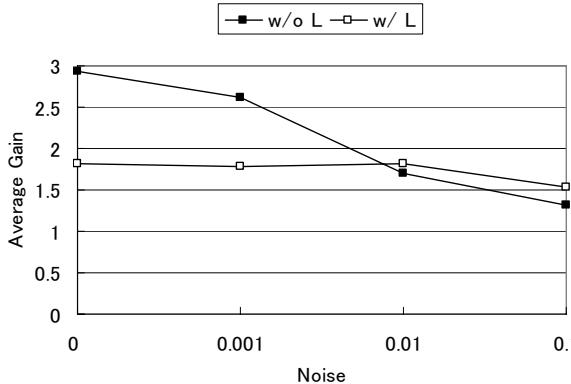
In this study, we fix two parameters,  $\epsilon = 0.1$  and  $\lambda = 0$  as the first step. A total of 2,000 generations are repeated in the simulation. The average of a trial is the value that is obtained by averaging the values from 1,000 to 2,000 generations. The results shown below are the average of over 10 tries.

## 4.3 Results

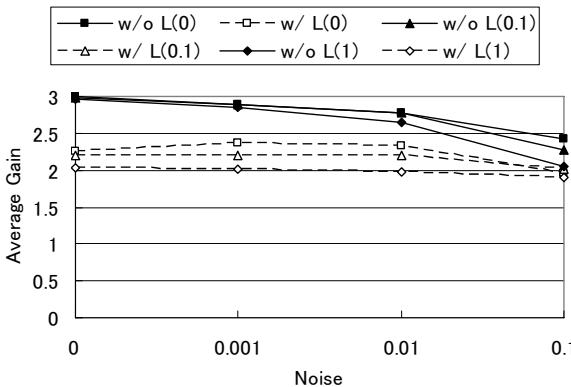
We conducted simulations by varying the parameters of noise probability.

At first, we focus on the average gain, which is the index indicating the number of cooperators existing in the population. An approximate index value of 3 indicates the existence of many cooperators.

The average gain in the pool case is shown in Fig. 1. “L” in the legend represents “learning.” The difference between the cases with and without learning is clear in this figure. In the case with learning, the value remains almost constant regardless of the noise probability. On the other hand, in the case without learning, the average gain is about 3 when the noise does not exist. This implies that almost all agents cooperate with each other. The line decreases monotonically from the value 3, as the noise increases. It is observed that the lines intersect



**Fig. 1.** Average gain in the pool case

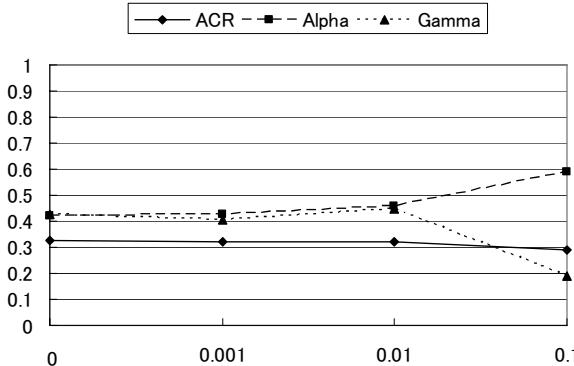


**Fig. 2.** Average gain in the network case ( $p = 0.1$ )

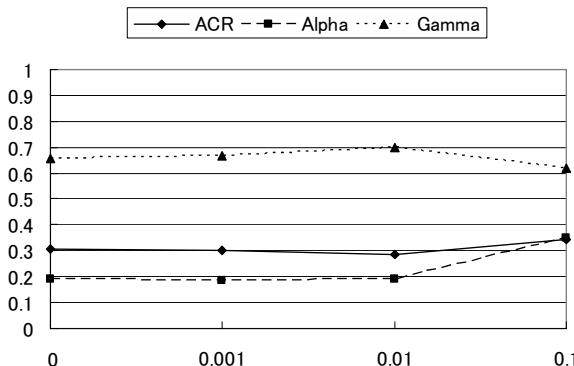
when the noise is approximately  $1.0 \times 10^{-2}$ . This implies that the agents cooperate well based on a condition. In this case, the condition is the noise probability.

The average gain in the network case is shown in Fig. 2. The number in the legend represents the randomness parameter of the small-world network. It is evident that all points in the no-learning case are higher than the ones in the learning case. The lines of the no-learning case decrease monotonically and vary when noise = 0.1. However, the lines of the learning case remain almost constant in the range 2.0~2.4, irrespective of the change in the noise or the randomness parameter of the network structure. With regard to the randomness of the network, the higher value lines are lower than the lower value lines in both learning and no-learning cases.

The fact that no-learning case is better than learning case, except under a few special conditions, implies that learning does not always make agents happy. In our experiments, the exception is in the limited condition that is in the range of higher noise in the pool case. The agents with learning ability tend to be



**Fig. 3.** Reinforcement learning parameters in the pool case



**Fig. 4.** Reinforcement learning parameters in the network case

extremely greedy; they tend to defect to get higher rewards. In the end, they select Nash equilibrium in the traditional PD game. Learning is a useful ability, but learning without additional wisdom, such as a general social rule or anticipation of the opponent's action, makes agents non-cooperative.

Next, the focus is on the learning process. In addition to the reinforcement parameters  $\alpha$  and  $\gamma$ , we define an index "ACR (action change rate)" to investigate the effect of learning. ACR is the rate of the action difference in each possible state between the strategy changed by learning and the initial strategy decided by the gene.

Figs. 3 and 4 depict the case of pool and the small-world network of  $p = 0.1$  respectively.

The ACR line is maintained at about 0.3 in both cases, as shown in Figs. 3 and 4. This implies that there are no stable strategies by nature. Learning always plays a constant role to adapt to the environment.

With regard to  $\alpha, \gamma$  of the variable reinforcement parameters of agents, their variations are shown in Figs. 3 and 4.  $\alpha$  could be considered as the speed at which the value function changes.  $\gamma$  expresses the evaluation of the gain that

would be received in the future. In the pool case,  $\alpha$  and  $\gamma$  are approximately 0.4 when the range of noise rate is less than 0.01. At noise = 0.1,  $\alpha$  increases and  $\gamma$  decreases suddenly.

On the other hand in the network case, until noise = 0.01,  $\alpha$  is 0.2 and  $\gamma$  is about 0.7. Then, similar to the pool case,  $\alpha$  increases and  $\gamma$  decreases.

Comparing the two cases, the values of  $\alpha$  and  $\gamma$  are clearly different. These data imply that, in the network case, agents evaluate the gain that would be received in the future and they do not change their strategy rapidly; this is because they tend to maintain their relationships with other agents and they can expect that there is a high possibility that the opponent uses the same strategy. On the other hand, in the pool case, the agents evaluate the gain received at the moment, and they change their strategy rapidly because they face unknown opponents in almost all games. It is noteworthy that the learning process differs considerably depending on the spatial structure, regardless of the fact that each ACR is almost identical.

The characteristics of the lines represent the behavior of the noise. At a lower noise, the values are almost constant; however,  $\alpha$  increases and  $\gamma$  decreases at the highest noise point. These results are explained by the same reason, that is, uncertainty of the opponent's strategy.

## 5 Conclusion

We studied the Prisoner's Dilemma game on the spatial structure played by agents who learn and evolve.

We proposed an agent model incorporating learning and evolution of strategy. These experiments revealed that the spatial structure of the agents influences cooperation and learning and evolution. In comparison with the learning case, agents in the no-learning case perform better, except under certain special conditions against our intuition. There are no stable strategies by nature and learning always plays a role in adaptation to the environment. The learning process is quite different depending on the spatial structure regardless of the fact that each ACR is almost identical.

For further study, the condition for effective learning is required to be clarified. In addition, the dynamic network case has to be considered. The next step is to bridge the gap between the pool and the network cases.

Although our study is still in a preliminary stage, it will contribute to the agent or system design policy in the future.

## References

1. Baldwin, J.M.: A new factor in evolution. *American Naturalist* **30** (1896) 441–451
2. Lindgren, K.: Evolutionary phenomena in simple dynamics. *Artificial Life II: Proceedings of the Workshop on Artificial Life held February 1990 in Santa Fe, New Mexico* (1992) 295–312
3. Sandholm, T.W., Crites, R.H.: Multiagent reinforcement learning in the iterated prisoner's dilemma. *Biosystems* **37**(147–166) (1996)

4. Hingston, P., Kendall, G.: Learning versus evolution in iterated prisoner's dilemma. In: Proceedings of the Congress on Evolutionary Computation, CEC2004. Volume 1., Portland, Oregon (2004) 364–372
5. Suzuki, R., Arita, T.: Interactions between learning and evolution: Outstanding strategy generated by the baldwin effect. *Biosystems* **77**(1-3) (2004) 57–71
6. Nowak, M.A., May, R.M.: Evolutionary games and spatial chaos. *Nature* **359** (1992) 826–829
7. Lindgren, K., Nordahl, M.G.: Evolutionary dynamics of spatial games. *Physica D* **75** (1994) 292–309
8. Watts, D.J.: Small Worlds. Princeton University Press (1999)
9. Masuda, N., Aihara, K.: Spatial prisoner's dilemma optimally played in small-world networks. *Physics Letters A* **313** (2003) 55–61
10. Ono, M., Ishizuka, M.: Prisoner's dilemma game on network. In: Proceedings of the Eighth Pacific-Rim International Workshop on Multi-Agents, PRIMA2005, Kuala Lumpur, Malaysia (2005) 9–22
11. Sutton, R.S., Barto, A.G.: Reinforcement Learning. MIT Press (1998)
12. Watts, D.J., Strogatz, S.H.: Collective dynamics of 'small-world' networks. *Nature* **393** (1998) 440–442
13. Axelrod, R.: The Evolution of Cooperation. Basic Books (1985)
14. Nowak, M.A., Sigmund, K.: A strategy of win-stay, lose-shift that outperforms tit-for-tat in the prisoner's dilemma game. *Nature* **364** (1993) 56–58

# Immune Clonal Selection Evolutionary Strategy for Constrained Optimization

Wenping Ma, Licheng Jiao, Maoguo Gong, and Ronghua Shang

Institute of Intelligent Information Processing, P.O. Box 224, Xidian University,  
Xi'an, 710071, P.R. China  
wpma@mail.xidian.edu.cn, maoguo\_gong@hotmail.com

**Abstract.** Based on the clonal selection theory, a novel artificial immune systems algorithm, immune clonal selection evolutionary strategy for constrained optimization (ICSCES), is put forward. The new algorithm uses the stochastic ranking constraint-handling technique, realizes local search using clonal proliferation and clonal selection, and global search using clonal deletion. The experimental results on ten benchmark problems show, compared with the  $(\mu, \lambda)$  evolutionary strategies adopting stochastic ranking technique and dynamic penalty function method, ICSCES has the ability of significantly improving the search performance both in convergence speed and precision.

## 1 Introduction

Just like evolutionary algorithms (EAs), artificial immune systems make use of the mechanism of vertebrate immune system, the model of processing information, and construct new computational intelligence methods with immunology terms and fundamental, for providing some novel methods to solve problems<sup>[1]</sup>. For example, it is available for any function, especially the functions that have no expressions, or have expressions but cannot be computed accurately. Artificial immune systems provide the evolutionary learning mechanism like noise enduring, non-teacher learning, self-organization, and memory, and combine with some advantages of other systems, like classifier, neural network and machine reasoning. Thus it has the potential for providing novel method for solving problems, and its research production refers to many fields, such as control, data processing, optimization learning and trouble diagnosing. And it has been a research focus following the neural network, fuzzy logic and evolutionary computation<sup>[2,3]</sup>.

Constrained optimizations, i.e. nonlinear programming, is a very important technology almost in all engineering, operational research and mathematics. The main problem using evolutionary algorithms to constrained optimizations is how to deal with constraints. In resent years, some methods were proposed to solve this problem, such as rejecting method, repairing method, and penalty function method<sup>[4]</sup> which is commonly used. But the traditional penalty functions depend much on penalty coefficients, and need to balance the relationship between penalty functions and objective function based on particular problems. Stochastic ranking for constrained evolutionary optimization is put forward in [5], which provides novel ideas about EAs to solve correlative problems, and obtains nicer effect.

## 2 Basic Definitions

In order to describe the algorithms well, we define the terms in our algorithms as follows.

### Definition 1. Antigen

In artificial immune system, antigens refer to problems and its constraints, especially to constrained problems.

$$(P) \begin{cases} \text{minimize } f(\mathbf{x}) = f(x_1, x_2, \dots, x_n) \\ \text{subject to } g_i(\mathbf{x}) < 0 \quad i = 1, 2, \dots, p \\ \qquad \qquad h_j(\mathbf{x}) = 0 \quad j = p+1, p+2, \dots, q \end{cases} \quad (1)$$

Where  $\mathbf{x} = (x_1, x_2, \dots, x_n)$ , antigen is the function of objective function  $f(\mathbf{x})$ , namely,  $G(\mathbf{x}) = g(f(\mathbf{x}))$ , similar to the effect of antigen in immunology. It is the initial factor in artificial immune system.

### Definition 2. Antibody

Set  $I = \{\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3, \dots, \mathbf{b}_n\}$ , which is called antibody space, it is the set of antibodies available during artificial immune response, where  $n$  is a infinite integer. Antibody  $b$  varies with antigen  $G$ , such as binary string, real number sequence.

Make  $\mathbf{b} = b_1 b_2 \dots b_l$  for example, depending on biological term,  $b_i$  is regarded as genetic gene, called allele, whose probable value is correlative to the coding method. In practice, binary coding and decimal coding are used frequently. i.e., an antibody is represented as 8 bits binary digits, '0-1-1-1-0-1-0-0'. The antibody population  $B = \{\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3, \dots, \mathbf{b}_m\}$ , is an  $m$ -dimensional group of antibody  $b$ , and it is a point in the antibody group space  $I$ , where positive integer  $m$  is called the size of antibody population.

### Definition 3. Antibody-Antigen Affinity ( $A_g$ - $A_b$ Affinity)

The reflection of the total combination power locates between antigen and antibodies. In AIS, it generally indicates values of objective functions or fitness measurement of the problem. In this paper, consulting [11], we define the  $A_g$ - $A_b$  affinity as follows: introducing a stochastic numeral  $P_f$ , in feasible solution space of problem, making objective function as  $A_g$ - $A_b$  affinity with probability 1, in unfeasible solution space, making the normalized value of objective function as  $A_g$ - $A_b$  affinity with probability  $P_f$ , making the normalized value of penalty function as  $A_g$ - $A_b$  affinity with probability  $1 - P_f$ . Through this it will achieve the balance between objective function and penalty function probability pattern of calculating affinity, and the successful probability of the affinity between two antibodies is

$$P_w = P_{fw} P_f + P_{\phi w} (1 - P_f) \quad (2)$$

where  $P_{fw}$  is the successful probability of antibody for its objective function,  $P_{\phi w}$  is the successful probability of antibody for its penalty function. Specially, the successful probability of antibody is  $P_w = P_{fw}$  when both antibodies are in feasible solution space.

### 3 Presentation of the Algorithm

Inspired by the process of biologic immune response, the operations in Immune Clonal Selection Constraint Evolutionary Strategy mainly include clonal deletion  $r_1$ , clonal proliferation  $r_2$ , genic mutation  $r_3$  and clonal selection  $r_4$ .

#### 3.1 Clonal Deletion $r_1$

Clonal deletion  $r_1$  is a simple simulation of immune tolerance in immunology. In immunology, the negative adjusting of B cells during the antibody creating may be clonal deletion or clonal anergy. This paper intitutes this as clonal deletion without any difference. Clonal deletion  $r_1$  on the antibody population  $\mathbf{B} = \{\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3, \dots, \mathbf{b}_n\}$  is defined as:

$$\mathbf{b}_1 + \mathbf{b}_2 + \dots + \mathbf{b}_n \xrightarrow{r_1} \alpha_1 \mathbf{b}_1 + \alpha_2 \mathbf{b}_2 + \dots + \alpha_n \mathbf{b}_n \quad (3)$$

where  $\alpha_i \mathbf{b}_i$  represents that there exists  $\mathbf{b}_i$  with the number of  $\alpha_i$  in antibody population after clonal deletion on antibody  $\mathbf{b}_i$ , in which  $\alpha_i$  is a integer with  $\alpha_i \geq 0$ .  $\alpha_i = 0$  means that antibody  $\mathbf{b}_i$  presents immune tolerance, or clonal deletion, the reason of which may be the low  $A_g$ - $A_b$  Affinity of  $\mathbf{b}_i$ , or may be the decline of the whole diversity of antibody population result from the common genotype between antibody  $\mathbf{b}_i$  and other antibodies.  $\alpha_i > 1$  means that the antibodies of high affinity (i.e. target cells) have a local increase with the positive feedback immune modulation of target cells result from immune tolerance. In artificial immune system, for holding the durative of algorithms, there is  $\alpha_1 + \alpha_2 + \dots + \alpha_n = n$  in common.

#### 3.2 Clonal Proliferation $r_2$

In immunology, Clone means propagating asexually, so that a group of genetically identical cells can be descended from a single common ancestor. In artificial immune system, the clonal proliferation  $r_2$  on antibody population  $\mathbf{B} = \{\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3, \dots, \mathbf{b}_n\}$  is defined as:

$$\mathbf{b}_1 + \mathbf{b}_2 + \dots + \mathbf{b}_n \xrightarrow{r_2} \{\mathbf{b}_1^1 + \mathbf{b}_1^2 + \dots + \mathbf{b}_1^{q_1}\} + \{\mathbf{b}_2^1 + \mathbf{b}_2^2 + \dots + \mathbf{b}_2^{q_2}\} + \dots + \{\mathbf{b}_n^1 + \mathbf{b}_n^2 + \dots + \mathbf{b}_n^{q_n}\} \quad (4)$$

where  $\mathbf{b}_i^j = \mathbf{b}_i$ ,  $i = 1, 2, \dots, n$ ;  $j = 1, 2, \dots, q_i$ ,  $q_i \in [1, n_c]$  is an self-adaptive parameter, or set as an constant, where  $n_c$  is related to the upper limit of clone scale, and  $q_i = 1$  represents that there is no clonal proliferation on antibodies. It is obvious that clonal proliferation above is similar to that of immunology, which is a simple process of propagating asexually. All the antibodies in sub-population  $\mathbf{B}_i(\mathbf{b}_i^1, \mathbf{b}_i^2, \dots, \mathbf{b}_i^{q_i})$  are result from the clonal proliferation on the same antibody  $\mathbf{b}_i$  and have the same property as antibody  $\mathbf{b}_i$ .

### 3.3 Genic Mutation $r_3$

Genic mutation  $r_3$  is a simulation on recognizing external pattern, antibody gene mutation, and compilation in immune system. Genic mutation  $r_3$  on antibody population  $\mathbf{B} = \{\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3, \dots, \mathbf{b}_n\}$  is defined as:

$$\mathbf{b}_1 + \mathbf{b}_2 + \dots + \mathbf{b}_n \xrightarrow{r_3} \mathbf{b}'_1 + \mathbf{b}'_2 + \dots + \mathbf{b}'_n \quad (5)$$

The essence of genic mutation is that change genic values in some genic positions of antibody. As far as the binary encoding cluster of character set {0,1} is concerned, mutation means to make some genic positions inverse with a certain probability (i.e. 1→0 or 0→1). For real number encoding cluster, uniform mutation, Gaussian mutation, Cauchy mutation and some stochastic mutations can be used. This paper we adopt Gaussian mutation as follows:

Antibody  $\mathbf{b}_i$  contains two vectors  $(\mathbf{x}, \sigma)$ , where the first vector  $\mathbf{x}$  represents a point in search space, and the second vector  $\sigma$  represents standard deviation. Similarly, offspring  $\mathbf{b}'_i$  also contains two vectors  $(\mathbf{x}', \sigma')$ , where  $\mathbf{x}'$  and  $\sigma'$  result from formula below:

$$\sigma' = \sigma e^{N(0, \Delta\sigma)} \quad (6)$$

$$\mathbf{x}' = \mathbf{x} + N(0, \Delta\sigma') \quad (7)$$

where  $N(0, \Delta\sigma)$  is a vector of absolute Gaussian stochastic number with average value 0 and standard deviation  $\sigma$ .

### 3.4 Clonal Selection $r_4$

Different from the selection handling in evolutionary algorithms, clonal selection  $r_4$  is a asexual selection process, we selects excellent individuals in sub-generation in which each of antibodies has clone proliferated, thereby forming new population.

For antibody population  $\mathbf{B}(\mathbf{b}_1^1, \mathbf{b}_1^2, \dots, \mathbf{b}_1^{q_1}, \mathbf{b}_2^1, \mathbf{b}_2^2, \dots, \mathbf{b}_2^{q_2}, \dots, \mathbf{b}_n^1, \mathbf{b}_n^2, \dots, \mathbf{b}_n^{q_n})$ , clonal selection  $r_4$  is defined as:

$$\{\mathbf{b}_1^1 + \mathbf{b}_1^2 + \dots + \mathbf{b}_1^{q_1}\} + \{\mathbf{b}_2^1 + \mathbf{b}_2^2 + \dots + \mathbf{b}_2^{q_2}\} + \dots + \{\mathbf{b}_n^1 + \mathbf{b}_n^2 + \dots + \mathbf{b}_n^{q_n}\} \xrightarrow{r_4} \mathbf{b}'_1 + \mathbf{b}'_2 + \dots + \mathbf{b}'_n \quad (8)$$

It is obvious that clonal selection is the contrary handling of clonal proliferation. The sub-population formed by the clonal proliferation on the same antibody  $\mathbf{b}_i$  heightens local affinity through clonal selection after the compilation of artificial immune system. Concretely,  $\forall i = 1, 2, \dots, n, \exists j \in \{1, 2, \dots, q_i\}$ , making the affinity of antibody  $\mathbf{b}_i^j$  highest in sub-population  $\mathbf{B}_i(\mathbf{b}_i^1, \mathbf{b}_i^2, \dots, \mathbf{b}_i^{q_i})$ , as a result the selection stress of  $\mathbf{b}_i^j$ , namely the probability of  $\mathbf{b}'_i = \mathbf{b}_i^j$ , is the largest. In this paper, to be robust, we set  $p(\mathbf{b}'_i = \mathbf{b}_i^j) = 1$ .

### 3.5 Immune Clonal Selection Constraint Evolutionary Strategy

Aim at constrained optimizations, making antigen  $G$  as formula (1),  $A_g$ - $A_b$  Affinity is defined as Definition 3, the Immune Clonal Selection Constraint Evolutionary Strategy is designed as follow:

#### Algorithm 1. Immune Clonal Selection Constraint Evolutionary Strategy

Step 1: Set the initial parameters: stochastic number  $P_f$ , the size of antibody population  $N$ , clonal scale  $q$ , clonal deletion scale  $d$ , and genic mutation probability  $P_m$ . Randomly generate the original antibody population  $\mathbf{B}(0)$ . Give the conditions of affinity maturity. Set the current generation  $k=0$ ;

Step 2: If  $\mathbf{B}(k)$  satisfies the conditions of affinity maturity, export the antibody with the highest  $A_g$ - $A_b$  Affinity in  $\mathbf{B}(k)$ , Stop. Otherwise, go to Step 3;

Step 3: Clonal deletion  $r_1$  on  $\mathbf{B}(k)$ , get new antibody population  $\mathbf{B}^{(1)}(k)$ ;

Step 4: Clonal proliferation  $r_2$  on  $\mathbf{B}^{(1)}(k)$ , get new antibody population  $\mathbf{B}^{(2)}(k)$ ;

Step 5: Genic mutation  $r_3$  on  $\mathbf{B}^{(2)}(k)$ , get new antibody population  $\mathbf{B}^{(3)}(k)$ ;

Step 6: Clonal selection  $r_4$  on  $\mathbf{B}^{(3)}(k)$ , get new antibody population  $\mathbf{B}^{(4)}(k)$ ;

Step 7: Set  $\mathbf{B}(k+1)=\mathbf{B}^{(4)}(k)$ ,  $k=k+1$ , go to Step 2.

## 4 Validation of ICSCES's Effectiveness

### 4.1 Description of Test Problems

In order to validate our algorithm, ICSCES is executed to solve the following ten benchmarks, which has been described in [5].

#### G01:

$$\text{Minimize } f(x) = 5 \sum_{i=1}^4 x_i - 5 \sum_{i=1}^4 x_i^2 - \sum_{i=5}^{13} x_i$$

Subject to

$$\begin{aligned}
g_1(x) &= 2x_1 + 2x_2 + x_{10} + x_{11} - 10 \leq 0; & g_2(x) &= 2x_1 + 2x_3 + x_{10} + x_{12} - 10 \leq 0; \\
g_3(x) &= 2x_1 + 2x_3 + x_{11} + x_{12} - 10 \leq 0; & g_4(x) &= -8x_1 + x_{10} \leq 0; \\
g_5(x) &= -8x_2 + x_{11} \leq 0; & g_6(x) &= -8x_3 + x_{12} \leq 0; & g_7(x) &= -2x_4 - x_5 + x_{10} \leq 0; \\
g_8(x) &= -2x_6 - x_7 + x_{11} \leq 0; & g_9(x) &= -2x_8 - x_9 + x_{12} \leq 0.
\end{aligned}$$

where  $0 \leq x_i \leq 1(i=1,2,\dots,9); 0 \leq x_i \leq 100(i=10,11,12); 0 \leq x_{13} \leq 1$ .

the global value locals at  $x^* = (1, 1, 1, 1, 1, 1, 1, 1, 3, 3, 1)$ ,  $f(x^*) = -15$ .

### G02:

$$\text{Maximize } f(x) = \left| \frac{\sum_{i=1}^n \cos^4(x_i) - 2 \prod_{i=1}^n \cos^2(x_i)}{\sqrt{\sum_{i=1}^n i x_i^2}} \right|$$

$$\text{Subject to } g_1(x) = 0.75 - \prod_{i=1}^n x_i \leq 0; \quad g_2(x) = \sum_{i=1}^n x_i - 7.5n \leq 0.$$

Where  $n=20$  and  $0 \leq x_i \leq 10(i=1,2,\dots,n)$ . The global maximum is unknown, and the best we found is  $f(x^*) = 0.803619$ , where  $g_1 = -10^{-8}$ .

### G03:

$$\text{Maximize } f(x) = (\sqrt{n})^n \prod_{i=1}^n x_i$$

$$\text{Subject to } h(x) = \sum_{i=1}^n x_i^2 - 1 = 0.$$

$n=10$  and  $0 \leq x_i \leq 1(i=1,2,\dots,n)$ , the global value locals at  $x_i^* = 1/\sqrt{n}(i=1,2,\dots,n)$ ,  $f(x^*) = 1$ .

### G04:

$$\text{Minimize } f(x) = 5.3578547x_3^2 + 0.8356891x_1x_5 + 37.293239x_1 - 40792.141$$

Subject to/

$$g_1(x) = 85.334407 + 0.0056858x_2x_5 + 0.0006262x_1x_4 - 0.0022053x_3x_5 - 92 \leq 0;$$

$$g_2(x) = 85.334407 - 0.0056858x_2x_5 - 0.0006262x_1x_4 + 0.0022053x_3x_5 \leq 0;$$

$$g_3(x) = 80.51249 + 0.0071317x_2x_5 + 0.0029955x_1x_2 + 0.0021813x_3^2 - 110 \leq 0;$$

$$g_4(x) = -80.51249 - 0.0071317x_2x_5 - 0.0029955x_1x_2 - 0.0021813x_3^2 + 90 \leq 0;$$

$$g_5(x) = 9.300961 + 0.0047026x_3x_5 + 0.0012547x_1x_3 + 0.0019085x_3x_4 - 25 \leq 0;$$

$$g_6(x) = -9.300961 - 0.0047026x_3x_5 - 0.0012547x_1x_3 - 0.0019085x_3x_4 + 20 \leq 0;$$

where  $78 \leq x_1 \leq 102; 33 \leq x_2 \leq 45; 27 \leq x_i \leq 45(i=3,4,5)$ .

The optimal value locals at  $x^* = (78, 33, 29.995256025682, 45, 36.775812905788)$ ,  $f(x^*) = -30665.539$ .

### G05:

$$\text{Minimize } f(x) = (x_1 - 10)^3 + (x_2 - 20)^3$$

Subject to

$$g_1(x) = -(x_1 - 5)^2 - (x_2 - 5)^2 + 100 \leq 0; \quad g_2(x) = (x_1 - 6)^2 + (x_2 - 5)^2 - 82.81 \leq 0.$$

Where,  $13 \leq x_1 \leq 100$ ;  $0 \leq x_2 \leq 100$ .

The optimal value locals at  $x^* = (14.095, 0.84296)$ ,  $f(x^*) = -6961.81388$ .

### G06:

$$\text{Maximize } f(x) = \frac{\sin^3(2\pi x_1) \sin(2\pi x_2)}{x_1^3(x_1 + x_2)}$$

Subject to

$$g_1(x) = x_1^2 - x_2 + 1 \leq 0; \quad g_2(x) = 1 - x_1 + (x_2 - 4)^2 \leq 0.$$

where  $0 \leq x_1 \leq 10$ ;  $0 \leq x_2 \leq 10$ .

The optimal value locals at  $x^* = (1.2279713, 4.2453733)$ , the optimal value is  $f(x^*) = 0.095825$ .

### G07:

Minimize

$$f(x) = (x_1 - 10)^2 + 5(x_2 - 12)^2 + x_3^4 + 3(x_4 - 11)^2 + 10x_5^6 + 7x_6^2 + x_7^4 - 4x_6x_7 - 10x_6 - 8x_7$$

Subject to

$$g_1(x) = -127 + 2x_1^2 + 3x_2^4 + x_3 + 4x_4^2 + 5x_5 \leq 0; \quad g_2(x) = -282 + 7x_1 + 3x_2 + 10x_3^2 + x_4 - x_5 \leq 0;$$

$$g_3(x) = -196 + 23x_1 + x_2^2 + 6x_6^2 - 8x_7 \leq 0; \quad g_4(x) = 4x_1^2 + x_2^2 - 3x_1x_2 + 2x_3^2 + 5x_6 - 11x_7 \leq 0.$$

where  $-10 \leq x_i \leq 10 (i = 1, 2, \dots, 7)$ .

The optimal value locals at

$$x^* = (2.330499, 1.951372, -0.4775414, 4.365726, -0.6244870, 1.038131, 1.594227),$$

the optimal value is  $f(x^*) = 680.6300573$

### G08:

$$\text{Minimize } f(x) = x_1^2 + (x_2 - 1)^2$$

$$\text{Subject to } h(x) = x_2 - x_1^2 = 0.$$

where  $-1 \leq x_1 \leq 1, -1 \leq x_2 \leq 1$ .

The optimal value locals at  $x^* = (\pm 1/\sqrt{2}, 1/2)$ , the optimal value is  $f(x^*) = 0.75$ .

### G09:

$$\text{Maximize } f(x) = (100 - (x_1 - 5)^2 - (x_2 - 5)^2 - (x_3 - 5)^2)/100$$

$$\text{Subject to } g(x) = (x_1 - p)^2 + (x_2 - q)^2 + (x_3 - r)^2 - 0.0625 \leq 0.$$

where  $0 \leq x_i \leq 10 (i = 1, 2, 3), p, q, r = 1, 2, \dots, 9$ .

The optimal value locals at  $x^* = (5, 5, 5)$ , the optimal value is  $f(x^*) = 1$ .

### G10:

$$\text{Minimize } f(x) = e^{x_1 x_2 x_3 x_4 x_5}$$

Subject to

$$h_1(x) = x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_5^2 - 10 = 0; \quad h_2(x) = x_2 x_3 - 5x_4 x_5 = 0; \quad h_3(x) = x_1^3 + x_2^3 + 1 = 0.$$

where  $-2.3 \leq x_i \leq 2.3 (i=1,2)$ ;  $-3.2 \leq x_i \leq 3.2 (i=3,4,5)$ .

The optimal value locals at

$x^* = (-1.717143, 1.595709, 1.827247, -0.763643, -0.763645)$  the optimal value is  $f(x^*) = 0.0539498$ .

## 4.2 Test Results and Performance Analysis

The performance comparisons among ICSCES, RY<sup>[5]</sup> and KM<sup>[6]</sup> to solve the above constraint problems are shown in Table 1. Set the parameters of ICSCES:  $P_f = 0.45$ , the size of antibody population  $N=100$ , the clone scale  $q=2$ , individual mutation probability  $P_m = 1$ , and clonal deletion probability  $d = 1 - \mu/\lambda = 0.875$ . Set the parameters of RY and KM with the optimal parameters in [5]. With the parameters above, for every test, KM needs to evaluate function values  $70 \times 20\,000$  times, and RY and ICSCES only need  $200 \times 1\,750$  times. The data is the statistical results obtained from 30 times of random running.

**Table 1.** Performance comparison of ICSCES, RY, and KM

test	Optimal		ICSCES	RY	KM
G01	-15.000	The optimal value	-15.000	-15.000	-14.7864
		Mean value	-15.000	-15.000	-14.7082
		The worst value	-15.000	-15.000	-14.6154
G02	0.803619	The optimal value	-0.803575	-0.803515	-0.79953
		Mean value	-0.779465	-0.781975	-0.79671
		The worst value	-0.716312	-0.726688	-0.79119
G03	-1.000	The optimal value	-1.000	-1.000	-0.9997
		Mean value	-1.000	-1.000	-0.9989
		The worst value	-0.999	-1.000	-0.9978
G04	30665.539	The optimal value	-30665.539	-30665.539	-30664.5
		Mean value	-30665.539	-30665.539	-30655.3
		The worst value	-30665.539	-30665.539	-30645.9
G05	6961.814	The optimal value	-6961.814	-6961.814	-6952.1
		Mean value	-6695.987	-6875.940	-6342.6
		The worst value	-6030.333	-6350.262	-5473.9
G06	0.095825	The optimal value	-0.095825	-0.095825	-0.095825
		Mean value	-0.095825	-0.095825	-0.0891568
		The worst value	-0.095825	-0.095825	-0.0291438
G07	680.630	The optimal value	680.630	680.630	680.91
		Mean value	680.652	680.656	681.16
		The worst value	680.801	680.763	683.18
G08	0.750	The optimal value	0.750	0.750	0.75
		Mean value	0.750	0.750	0.75
		The worst value	0.750	0.750	0.75
G09	1.000000	The optimal value	-1.000000	-1.000000	-0.999999857
		Mean value	-1.000000	-1.000000	-0.999134613
		The worst value	-1.000000	-1.000000	-0.991950498
G10	0.053950	The optimal value	0.053950	0.053957	0.054
		Mean value	0.054716	0.057006	0.064
		The worst value	0.440825	0.216915	0.557

**Table 2.** The mean evolutionary generation for ICSCES and RY to obtain the data shown in table 1

	G01	G02	G03	G04	G05	G06	G07	G08	G09	G10
ICSCES	621	1316	1094	348	16	124	592	66	62	340
RY	741	1086	1146	441	590	381	557	57	82	349

From Table 1, the optimal results obtained from ICSCES and RY are better than those from KM. While for G02 and G10, ICSCES can also find better results than RY. For other eight tests, ICSCES and RY can find results having the same precision. For tests G01, G06 and G10, the mean results of ICSCES and RY are better than these of KM clearly. While for G10, ICSCES is also better than RY. But for G02, ICSCES and RY are worse than KM, for G05, ICSCES and KM are worse than RY. For tests G01, G03, G04, G05, G06, G07, G09 and G10, the worst results of KM are bad clearly. From Table 2, for tests G01, G03, G04, G05, G06, G09 and G10, the generations obtaining the optimal results of ICSCES are less than RY. Specially for G05 and G06, the global convergence speed of ICSCES is far higher than RY. But for G02, G07 and G08, the convergence speed of ICSCES is little worse than RY.

Estimating an algorithm needs two important metrics, solving precision and convergence speed, from above results, we can get following conclusions:

- (1) Take solving precision into account, for tests G02 and G10, ICSCES is better than RY. While for other eight tests, ICSCES and RY have the same solving precision; For the ten test problems, ICSCES is better than KM.
- (2) Take convergence speed into account, for tests G01, G03, G04, G05, G06, G09 and G10, ICSCES is better than RY. For G02, G07, and G08, ICSCES is little worse than RY; The test results are that ICSCES and RY only have  $\frac{1}{4}$  the calculation amount than that of KM.

## 5 Concluding Remarks

This paper describes a new artificial immune system algorithm, i.e. the immune clonal selection evolutionary strategy for constrained optimizations. Inspired by the process of biologic immune response, ICSCES has used four operations including clonal deletion, clonal proliferation, genic mutation and clonal selection. The experimental results on ten benchmark problems show that when compared with the  $(\mu, \lambda)$ -ES adopting stochastic ranking method and dynamic penalty function method, ICSCES is capable of improving the search performance significantly in solving precision. In convergence speed, although the new algorithm is little worse than RY for tests G02, G07 and G08, for other seven tests, the new algorithm has clear dominance.

## References

1. Jiao Licheng, Du Haifeng. Artificial immune system: progress and prospect (in Chinese). ACTA ELECTRONICA SINICA, 2003, Vol.31, No.10: 1540-1548
2. Dasgupta, D.: Artificial Immune Systems and Their Applications. Springer-Verlag, Berlin Heidelberg New York (1999)

3. de Castro, L. N., Timmis, J.: Artificial Immune Systems: A New Computational Intelligence Approach. Springer-Verlag, Berlin Heidelberg New York (2002)
4. Mitsuo G., Runwei C.: Genetic Algorithm and Engineering Optimization. John Wiley & Sons Inc, New York (1999)
5. Thomas P. R., Yao X.: Stochastic Ranking for Constrained Evolutionary Optimization. *IEEE Transactions on Evolutionary Computation*, Vol. 4, No. 3 (2000) 284-294
6. Kozie S., Michalewicz Z.: Evolutionary algorithms, homomorphous mapping, and constrained parameter optimization. *Evolutionary Computation*, Vol. 7, No. 1 (1999) 19-44
7. Abbas, A.K., Lichtman, A.H., Pober, J.S.: Cellular and Molecular Immunology. 4th edn. W B Saunders Co., New York (2000)
8. Du H.F., Jiao L.C., Gong M.G., Liu R.C.: Adaptive Dynamic Clone Selection Algorithms. In: Zdzislaw P., Lotfi Z. (eds): Proceedings of the Fourth International Conference on Rough Sets and Current Trends in Computing (RSCTC'2004). Uppsala, Sweden. June 1-5 (2004)
9. T. Bäck: Evolutionary Algorithms in Theory and Practice. New York: Oxford University Press (1996)
10. Kim, J., Bentley, P.J.: Towards an Artificial Immune System for Network Intrusion Detection: an Investigation of Clonal Selection with a Negative Selection Operator. In: IEEE Neural Networks Council. Proceedings of the 2001 Congress on Evolutionary Computation. Vol. 2. Seoul Korea, IEEE (2001) 1244-1252
11. Hofmeyr, S., Forrest, S.: Immunity by Design: An Artificial Immune System. In: Proceedings of the Genetic and Evolutionary Computation Conference (GECCO), Morgan-Kaufmann, San Francisco, CA (1999) 1289-1296
12. Gasper, A., Collard, P.: From GAs to artificial immune systems: improving adaptation in time dependent optimization. In: Proceedings of the Congress on Evolutionary Computation (CEC 99). IEEE press (1999) 1859-1866

# An Intelligent System for Supporting Personal Creativity Based on Genetic Algorithm

Heng-Li Yang and Cheng-Hwa Lee

64, Sec. 2, Chihnan Rd., Mucha Dist, Taipei 116, Taiwan  
{yanh, 92356512}@nccu.edu.tw

**Abstract.** Creativity has long been recognized as vital to organizational success. IT (Information Technology) may play a supporting role to help organizations and groups in support of creativity. However idea generation inside the personal creative process is still a black box. This motivates us to propose an intelligent system for supporting personal creativity. This study proposes a human-machine interactive mechanism based on IGA (interactive genetic algorithm) for evolution process to help individuals' creativity. Chance discovery has been applied as the strategy in evolution process. It provides more efficient evolution. The intelligent system could bring individuals stimulus to improve personal creativity.

## 1 Introduction

Creativity is the implementation of a new or significantly improved idea, good, service, process or practice that is intended to be useful. Creativity has long been recognized as vital to organizational success [1,9,18]. The successful implementation of creativity is innovation. Product innovation is important to maintain a sufficient market share, similarly process innovation to produce low-cost product, and social innovation to maintain a flexible and durable organization. Amabile [1,2] provided a popular stage model of personal creative process based on previous works [21]. It includes four stages: problem identification, preparation, idea generation, and validation. At the third stage, idea generation, an individual would generate idea possibilities with a particular cognitive pathway. At this stage the creativity-relevant processes plays an important role [1,2]. IT (information technology) may play multiple supporting roles in the stage of the creative process. Proctor [17] suggested the use of a computer program in the production of creative thought. Machrone [11] indicated computer programs can provide "mind maps" or graphical representations for idea generation. Boden [3] explained computer software would facilitate the creativity of individuals by forcing the search for new ideas. Partridge and Rowe [16] indicated that computers could be enlisted to limit constraints on the work of creators to force the development of new ideas. Edwards [6] examined the impact of IT on creativity. The question arises as to how organizations can institutionalize its usage in support of creativity [5,12,19].

Expert systems and decision support systems have been applied commercially. Since we know the creativity is a most important factor for organization development,

how can we support? Data searching or accessing is not enough. This motivates us to propose an intelligent system for supporting personal creativity.

## 2 Related Works

For supporting personal creativity, we should check the creative process to explore how to apply IT. In addition, probably some chance discovery concepts could help us.

### 2.1 Creative Process Model

Wallas [21] formalized the four-stage model of the creative process, including preparation, incubation, illumination, and verification. In the past, the four-stage model or its variant has served and continues to serve as the basis for understanding the creative process. Amabile[1,2] incorporated a version of the basic stage model into her componential model of creativity. The creative process is described as consisting of several phases: problem or task identification, preparation (gathering and reactivating relevant information and resources), idea generation (seeking and producing potential responses), and idea validation and communication (testing the possible response against criteria). The most important phase is idea generation, which seems to be a crucial part for personal creativity.

### 2.2 IT Supporting

IT may play a supporting role to help individuals and organizations in support of creativity. IT becomes a conduit through which creativity-supportive goals and messages are distributed throughout the organization [5].

One IT-related problem is how to represent the knowledge, especially in describing the relations between the concepts. One of methods is ontology, which is the product of an attempt to formulate an exhaustive and rigorous conceptual schema about a domain. It is typically a hierarchical data structure containing all the relevant entities and their relationships and rules within that domain.

The other IT-related problem is how to assist the creativity-producing. A genetic algorithm (GA) may be applied. GA is defined as search procedures based on the mechanics of natural selection and genetics. GA applies three basic principles of Darwin's theory (inheritance, variation, and selection) with two main results from genetics (genetic coding and mechanisms of recombination) in one computational procedure. GA combines the exploitation of past results with the exploration of new areas of the search space. By using survival of the fittest techniques combined with a structured yet randomized information exchange, GA can mimic some of the innovative flair of human search. Inside a GA, a generation is a collection of artificial creatures (strings). In every new generation, a set of strings is created using information from the previous ones. They efficiently exploit historical information to speculate on new search points with expected improvement [7,8]. Furthermore, interactive genetic algorithms (IGA) introduce the idea of human evaluation that extended the area of application of GAs drastically [4]. IGAs represent humanized technology. They can fit human needs, instead of fitting some computational fitness

functions, which is a very approximate model of human needs. In this study, new ideas originating from the output of IGA would provide stimulus to people.

### 2.3 Chance Discovery

Chance discovery means discovering chances. It involves determining the significance of some piece of information about an event and then using this new knowledge in decision making [20]. It might combine with data mining methods for finding rare but important events with knowledge management, groupware, and social psychology. The applied perspectives are from social network, small world, association, etc. A social network is a set of persons with some pattern of contacts or interactions between them. Newman [14] divides into four loose categories of networks: social networks, information networks, technological networks, and biological networks. Over the past few years, researchers have come to recognize that the social network including people and information plays a crucial role in knowledge management [10]. Some indicators (e.g., distance, centrality, bridge) of “item” network might be applied to find some fresh ideas. The idea of small world asserts that that everyone in the world can be reached through a short chain of social acquaintances. Similarly, each “item” might have a shortcut to link others, which might imply fresh ideas. Association of items, particularly long-chain association, is useful to illuminate new ideas [15]. This study would apply these perspectives in the proposed system to find out some fresh ideas as stimulus through the process of human-machine interactions.

## 3 System Architecture

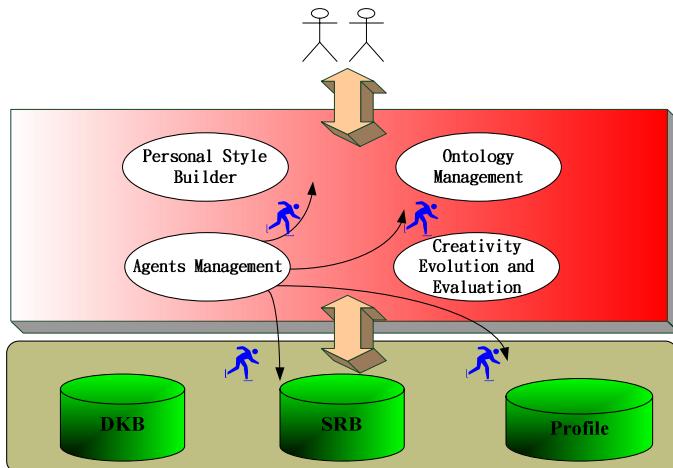
The proposed system architecture for supporting personal creativity is shown in Figure 1, which is divided into two layers: the database layer and the system layer. When a user logins first time, the system needs to learn user’s cognitive style. Later through interface, users could search related domain knowledge and ask for assistances during the creative process. According to the Stimulus-Response Model of behaviorism, a stimulus encourages response. Therefore, in our architecture, some stimuli are given by intelligent system to help people thinking during the creative process.

In the following, a scenario would be used to illustrate the proposed system function. Assume that there is a MIS (Management Information Systems) research center. The researchers, who obtained related PhD degrees before, are requested to generate research ideas periodically. The proposed intelligent supporting system would provide researchers to find out some new and fresh thinking for brainstorming. The research center has abundant domain knowledge in MIS area and already built well-organized document databases.

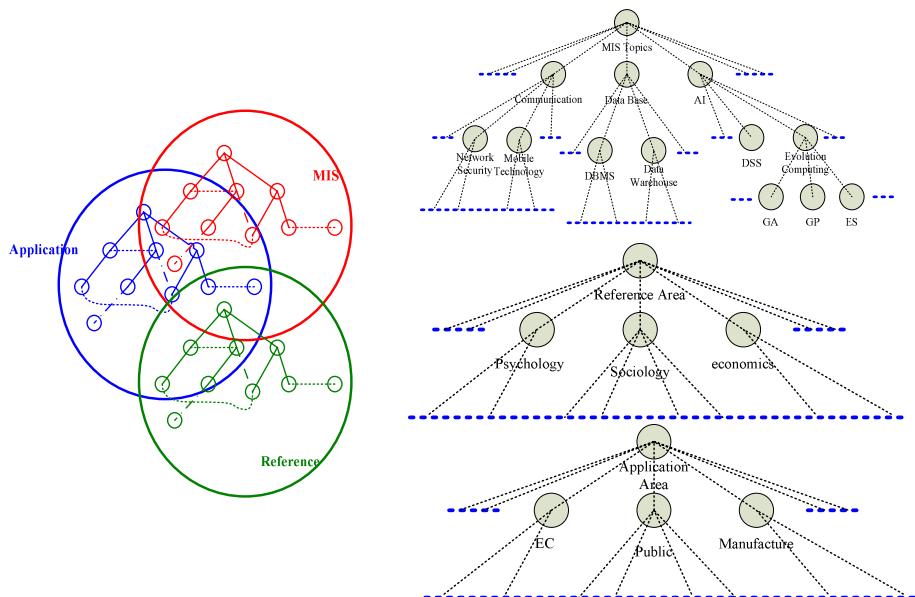
### 3.1 Database Layer

The database layer includes (1) domain knowledge base (DKB) that stores domain knowledge, (2) specific rule base (SRB) that has some pre-defined rules for new concept discovery, and (3) profile that records the styles of personal decision and

thinking so that the system could choose the appropriate stimulus to users. DKB and SRB could grow and be enhanced continuously. The MBTI (Myers-Briggs Type Indicator) catalog could be adopted to classify users into four styles: *Sensing-Thinking*, *Sensing-Feeling*, *Intuition-Thinking* and *Intuition-Feeling* [13].



**Fig. 1.** Proposed system architecture



**Fig. 2.** Ontology for DKB

Before the system could support user creation, mass and rich databases are essential and necessary to be built. In DKB, the ontology structure includes three elements: concepts, attributes and relations. The relations between concepts can be classified into three types: independent, intersection, and inheritance. In our scenario, the research center must first define catalogues for MIS. The classified catalogues provide meta-data of documents, including the classification trees of MIS research topics, their reference areas, and application areas (as shown in Figure 2). The primary knowledge would be built from many type documents, such as research reports, papers, and project documents, etc. Many techniques could be applied to build the database, e.g., data mining skills, text mining, document clustering, etc. Besides the catalogues, DKB would also include a graph of paper keywords, which idea would be adapted from social network. Reference links among papers would become relations of keywords.

SRB stores some pre-defined rules for new concept discovery. These rules are based on different perspectives. For example, *co-keywords rule*, and *second-order reference rule* are from chance discovery; *association by similarity*, *contiguity* and *contrast* are from association relationship. Based on these rules, the system would then provide him (her) other information related to the keywords to complement the chromosomes. Here a chromosome is set of the ideas. The available rules are illustrated here.

Assume that from historical documents, the system keeps the past records of frequencies of all keywords, and frequencies of sets of keywords, which simultaneously appeared. **Co-keywords rule** would concern about the cases (1) those keywords that appeared frequently at the same time; (2) those keywords that have higher frequency individually, but have not appeared so often at the same time before. The second case might imply some possible chances for research that have been ignored before. For example, if a user first considers “Knowledge Management”, after mining DKB, the system would find (1) that “Knowledge Management” ( $K_2$ ), “Data Mining” ( $K_1$ ) and “Organization Behavior” ( $K_3$ ) co-appeared quite often before; (2) “Security” ( $K_A$ ) and “Privacy” ( $K_B$ ) whose individual frequencies of appearance are higher, but have lower frequencies of appearing simultaneously with  $K_1$  and  $K_3$ , respectively. Then, the system would suggest add  $K_1$ ,  $K_3$ ,  $K_A$ , and  $K_B$  (as shown in Figure 3).

**Second-order reference rule** considers that some research chances could emerge from the indirect relationships or linkages. For example, “EC (Electronic Commerce)” ( $K_1$ ) is related to “Finance” ( $K_2$ ), and  $K_2$  is related to “Risk Management” ( $K_3$ ). Thus the rule suggests consider the possibility that  $K_1$  could be related to  $K_3$  (as shown in Figure 4).

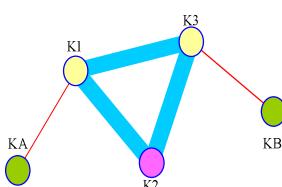


Fig. 3. Co-keywords rule

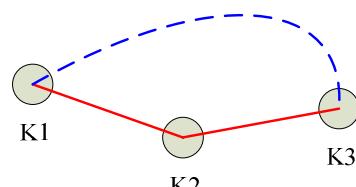


Fig. 4. Second-order reference rule

**Association by similarity** means that some keywords have the same parent node in a generalization (*is-a*) tree, thus these keywords might have the relationship of similarity. For example, the parent of both “GA” and “GP” is “Evolution Computing”. While “GA” appears, the system could give “GP” for reference. **Association by contiguity** means that some keywords have the same parent node in an aggregation (*part-of*) tree, thus these keywords might have the relationship of contiguity. For example, “Customer Life Cycle” includes “Fetch”, “Improve”, and “Maintain” phases. While “Fetch” phase appears, the system could give “Improve” or “Maintain” phase to user. **Association by contrast** means that some keywords have the same parent node in a generalization (*is-a*) tree, and furthermore these keywords have specific orders, thus they might have the relationship of contrast. For example, the parent of “Association Rule”, “Decision Tree”, “Neural Network”, and “Clustering” is “Data Mining”. From the perspective of importance, the highest is “Association Rule”, and the lowest is “Clustering”. While “Clustering” appears, the system could give “Association Rule” to user.

**Analog rule** would suggest user some research hints through analogy. Take an example, suppose that a research is classified as “EC” (A), its reference area as “Marketing” ( $\alpha$ ), research method as “Case Study” ( $RM-1$ ), and application area as “Internet Shopping” ( $AP-1$ ). Another research Advertisement’s (B) reference area is also  $\alpha$ , research method is  $RM-1$ , but application area is “Broadcast” ( $AP-2$ ). Then, it probably has some research chances in “EC” (A), which reference area could be  $\alpha$ , research method could be  $RM-1$ , but application area could become  $AP-2$  (shown as Figure 5).

...	A	...	$\alpha$	...	$RM-1$	...	$AP-1$	...
...	B	...	$\alpha$	...	$RM-1$	...	$AP-2$	...
...	A	...	$\alpha$	...	$RM-1$	...	$AP-2$	...
...	B	...	$\alpha$	...	$RM-1$	...	$AP-1$	...

**Fig. 5.** Analog rule

**Order-importance rule** considers the difference in keywords’ sequence. It implies that if the order is different, the issue is different. For example, a researcher concerns “Software Engineering” before “Software Market”. It might be explained that he is interested in the exchange market of software components. On the other side, if he considers “Software Market” before “Software Engineering”. It might be explained that he is interested in the methodology of software engineering for building software market. Hence the system should give different advices to user.

### 3.2 System Layer

The system layer includes four modules: (1) *Ontology Management* module, which allows to read, insert, delete, update domain knowledge in DKB, (2) *Personal Style Builder* module, which classify users into appropriate cognitive styles, (3) *Creativity Evolution and Evaluation* module, which is the core module to help create some new

ideas for users, and (4) *Agents Management*, which would initiate, coordinate, and trace execution of agents (*Personal Agent*, *Stimulus-Given Agent*, and *Learning Agent*). The *Personal Agent* would automatically provide personal profile to *Stimulus-Given Agent*, which would sequentially give appropriate stimulus. The *Learning Agent* would detect the possible necessity to change user personal profile or domain ontology, and notify the *Personal Style Builder* and *Ontology Management*, respectively.

An algorithm, IGA, is proposed to assist creativity evolution. Another algorithm, named PIEA (Personal Idea Evaluation Algorithm), is proposed to evaluate the ideas. These two algorithms are shown as Figures 6 and 7. Here the terms of genetic algorithm are applied. The chromosome is a basic building block of IGA. Each gene in the chromosome is keyword. Thus, one chromosome is set of keywords. One chromosome represents an idea (shown as Figure 5).

```
{Step 1: User inputs some keywords about research ideas
through interface.

Step 2: Stimulus-Given Agent gives first stimulus
according to user profile.

Step 3: System fetches related data from DKB.

Step 4: Generate chromosomes (ideas) based on rules
from SRB.

Step 5: Evaluate chromosomes by PIEA and sort the
evaluation values.

Step 6: Displays several higher sets of chromosomes to
user.

Step 7: Stimulus-Given Agent gives further stimulus
according to user profile and interacts with user.

Step 8: User chooses several chromosomes that he (her)
is interested, and then gives evaluative flags ( $W_i$ ).

Step 9: Generate chromosomes by crossover and mutate
based on the rules from SRB.

Step 10: Repeat Step 5 to Step 9 until user stops.

Step 11: End }
```

**Fig. 6.** The IGA algorithm

The above procedures could be explained as follows. First, a user needs input some initial research keywords. For example, if a user is interested in research of information security under EC, thus he gives “EC” and “Security” as keywords. If the user wishes to have more domain knowledge, he (she) could search DKB through interface. Upon receiving inputs from user, the stimulus-given agent would try to give some stimulus to user according to his (her) profile. Next, based on rules in SRB, the

```

{Step 1: Input gene (keyword) or chromosomes (ideas).
Step 2: Input evaluative flags ( $W_i$ ).
Step 3: While (genes (concepts) is still inputed)
{Step 3.1: In each run, compute distances ( $D_{ij}$ ) of any
two genes  $I, j$  by referencing Figure 2.
Step 3.2:  $R = R + (1/\max(\{D_{ij}\}))$ 
Step 4:  $V_i = (1/R) * W_i$ 
Step 5: End }

```

**Fig. 7.** The PIEA algorithm

system fetches data from DKB and generates some chromosomes (ideas) of the first generation, such as  $C1$  (EC-Security-SSL-Interception),  $C2$  (EC-Security-Payment-Marketing),  $C3$  (EC-Security-Neural Network), etc. Then such chromosomes would be passed to PIEA for evaluation. The basic idea for PIEA is that the larger distance between genes is, the more creativity the chromosome would imply. The distance is based on ontology and graphs in DKB. Take the above  $C3$  as the example. Its gene would be passed to PIEA one by one: EC, Security, and Neural Network. First input  $EC$ . Then input  $Security$  and compute the distance. The initial value of  $R$  is 0. After entering two genes, the  $D_{12}$  is 5, and  $R$  becomes  $1/5$ . After entering three genes,  $D_{12}$  is 5,  $D_{13}$  is 10, and  $D_{23}$  is 6, maximal  $D_{ij}$  is 10, and  $R$  is accumulated to  $3/10$  ( $1/5 + 1/10$ ). Thus  $V_i$  is  $10/3$ . In addition, user's personal justification also plays some roles to give weights ( $W_i$ ) (with default value 1).

After PIEA providing evaluation values, the system displays several sets of higher chromosomes to user and waits for feedbacks. While receiving feedbacks from user, the system generates new population of chromosomes to the next generation by crossover and mutation. Crossover is a genetic operator used to vary the programming of chromosomes from one generation to the next. Many techniques could be used, such as one point crossover, two point crossovers, cut and splice, etc. It implies inheritance and selection. It would take chromosomes to recombine children chromosomes. Mutation, which implies variation, is used to maintain genetic diversity. It may generate new children chromosomes. Traditionally, a common method of implementing the mutation operator involves generating a random variable for each bit in a chromosome. However, there are two differences between our proposed IGA and simple GA. First, our building block is not just a bit. Each gene has specific meaning. Second, we adopt some specific rules, such as chance discovery or association relationships, to replace random mutation. The system would repeat steps 5 to 9 again and again until user satisfies. Each run implies to give newborn generation, which inherits, recombines, and varies from last generation. The whole evolution procedure would need continuous interactions between user and system. The system would give user stimulus continuously through agents for brainstorming and hopes to drive him (her) to inspire some different creative thinking.

## 4 Discussions and Conclusions

Researches in personal creative process are not novel. Nevertheless, IT supporting have been limited. IT can play a supporting role to help organizations and groups in support of creativity [5]. But idea generation inside the personal creative process is still a black box. Outsiders cannot figure out what is going on inside. In a group, he (she) can get some fresh thoughts as stimulus by brainstorming. However, while he (she) is alone, he (she) lacks chances of interaction with others. He (she) must face an unknown status that is full of chaos himself. Thus motivates us to propose an intelligent supporting system to help the personal creative process. According to behaviorism theory in psychology, a stimulus could bring responses. The system could play an advisor to give people some ideas by human-machine interaction.

Applying our system, a person would be still involved in the evolutionary process and provide his (her) feedbacks to system to influence the evolutionary process and possible solutions. Because the volume of chromosome generated by system might be huge, PIEA is adopted as an evaluation and filtering mechanism to decrease user's burden of selection. In addition, which strategy for chromosomes mutation and recombination is important. Random selection might be an ad-hoc strategy used in the past. But it is not efficient and useful. In this study, we apply the perspective of some specific rules, such as chance discovery and association. People easily ignore unobvious information and were accustomed to past frequent opportunities. However, chance might emerge from the insignificant points. Chance discovery could allow us to find out some valuable information based on existing data for chromosomes mutation and recombination. It may also allow the evolution process more efficient.

However, some warnings are given. On one hand, user should not expect to just push a button to obtain the ready-made creation can. Even if the system could do so in the future, it would destroy human creativity. On the other hand, a creative idea is not just the combination of keywords as described in the above algorithms. Producing a new idea could be seemed as accomplishing a jigsaw puzzle. The first step might be obtaining enough pieces of polygons. Then we need more domain knowledge and human-machine interactions to construct the whole puzzle. Several future works are under exploration, such as theoretical basis, better human-machine interaction, and strategy used in evolution process, etc.

## References

1. Amabile, T. M.: *The Social Psychology of Creativity*, Westview, Boulder, CO (1983).
2. Amabile, T. M.: *Creativity in Context*, Westview, Boulder, CO (1996).
3. Boden, M. A.: Agents and Creativity, *Communications of the ACM*. 37(7) (1994) 117-121.
4. Caldwell, C. and Johnston, V. S.: Tracking a Criminal Suspect through 'Face-Space' with a Genetic Algorithm. *Proceedings of the Fourth International Conference on Genetic Algorithm*. (1991) 416-421.
5. Dewett, T.: Understanding the Relationship between Information Technology and Creativity in Organizations, *Creativity Research Journal*. 15(2 & 3) (2003) 167-182.
6. Edwards, S. M.: The Technology Paradox: Efficiency Versus Creativity, *Creativity Research Journal*. 13(2) (2001) 221-228.

7. Goldberg, D. E.: *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, Mass. (1989)
8. Holland, J. H.: *Adaptation in Natural and Artificial Systems*, Univ. of Michigan Press, Michigan (1975).
9. Katz, D.: The Motivational basis of Organizational Behavior. *Behavioral Science*. 9 (1964) 131-133.
10. Kautz, H., Selman, B. and Shah, M.: Referral Web: Combining Social Networks and Collaborative Filtering, *Communication of ACM*. 40(3) (1997) 27-36.
11. Machrone, B.: Computers and Creativity, *PC Magazine*. 13 (1994) 87-89.
12. Misue, K. and Tanaka, J.: A Handwriting Tool to Support Creative Activities, *Proceedings of the 9th International Conference on Knowledge-Based Intelligent Information and Engineering Systems*. (2005) 423-429.
13. Myers, I.B. and McCaulley, M.H.: Manual: A Guide to The Development and Use of The Myers-Briggs Type Indicator. Consulting Psychologists Press, Palo Alto, CA. (1985)
14. Newman, M. E. J.: The Structure and Function of Complex Networks, *SIAM REVIEW*. 45(2) (2003) 167-256.
15. Osborn, A.P.: *Applied Imagination*, Charles Scribner, NY. (1953)
16. Partridge, T. and Rowe, J.: Computers and Creativity, Wiltshire, England. (1994)
17. Proctor, T.: Brain, a Computer Program to Aid Creative Thinking, *Journal of Creative Behavior*. 25 (1991) 61-68.
18. Scott, S. G. and Bruce, R. A.: Determinants of Innovative Behavior: A Path Model of Individual Innovation in the Workplace. *Academy of Management Journal*. 37(3) (1994) 580-607.
19. Tang, X., Liu, Y. and Zhang, W.: Computerized Support for Idea Generation During Knowledge Creating Process, *Proceedings of the 9th International Conference on Knowledge-Based Intelligent Information and Engineering Systems*. (2005) 437-443.
20. Ohsawa, Y. and McBurney, P.: *Chance Discovery*. Springer-Verlag N.Y. (2003).
21. Wallas, G.: *Art of Thought*. Harcourt, Brace, N.Y. (1926).

# Generating Creative Ideas Through Patents

Guihua Wen<sup>1,3</sup>, Lijun Jiang<sup>1</sup>, Jun Wen<sup>2</sup>, and Nigel R. Shadbolt<sup>3</sup>

<sup>1</sup> South China University of Technology, Guangzhou 510641, China

[crghwen@scut.edu.cn](mailto:crghwen@scut.edu.cn), [cssturat@sohu.com](mailto:cssturat@sohu.com)

<sup>2</sup> Hubei Institute for Nationalities, HuBei Ensi 445000, China

[wenj\\_64@163.com](mailto:wenj_64@163.com)

<sup>3</sup> University of Southampton, Southampton SO17 1BJ, United Kingdom

[nrs@ecs.soton.ac.uk](mailto:nrs@ecs.soton.ac.uk)

**Abstract.** This paper describes a creativity support system for assisting human creative activities through patents. It integrates problem definition, reminding, creating, evaluating, and visualizing modules to automatically generate creative ideas. The system utilizes natural language processing techniques to map the patents to vectors by which related patent objects are automatically reminded around the given problem. These reminded objects are divided into text fragments, from which new idea can be generated through an interactive genetic algorithm. The system differs from existing creativity support systems in that it automatically reminds patents and generates creative ideas. This has been empirically validated by the conducted experiments.

## 1 Introduction

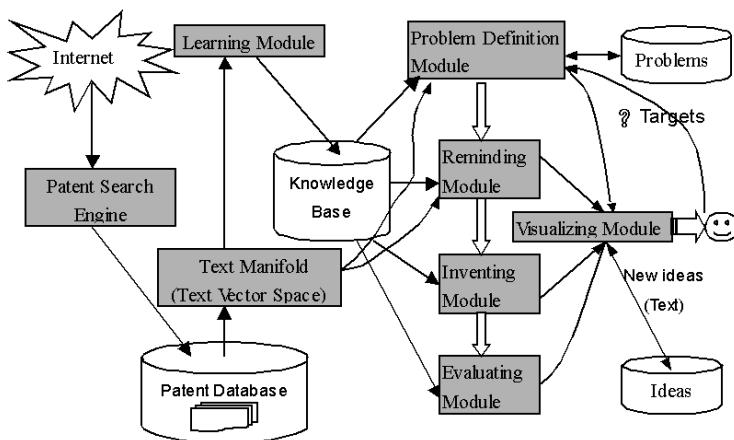
Most people facing ill-structured problems tend to think within a bounded, familiar, and narrow subset of the potential solution space, thus they routinely overlook up to 80% of the potential solution space[1,2]. Thus lots of creativity support systems are proposed to help people solve these problems [3,4,5,6]. Two primary approaches to study creativity support systems are used: intuitive and logical[8]. Intuitive methods use mechanisms to break what are believed to be mental blocks, such as the cognitive network model (CNM) of creativity [1] and the four-phase framework for creativity[13]. Logical methods involve systematic decomposition and analysis of the problem, relying heavily on technical databases and direct use of science and engineering principles. To integrate with aforementioned two approaches, a new theory called Computational Creativity Dynamics (CCD) is proposed[20,9,11,12]. Based on this theory, we develop a creativity support system named PatentProducist that emphasizes the usage and access of patent information. The reason is that patent gazettes reveal over 90% of research results for the patents, while more than 80% of information is not enclosed in academic theses and publications[11]. PatentProducist focuses on the very beginning of the creative design process, going from “no ideas” to “some ideas”, which hopefully include “some potentially important design concepts”. These novel ideas are provided for user as creative stimuli for further thinking.

## 2 Patent-Based Creativity Support System

CCD provides a framework for mathematically integrating the various aspects of creativity that are commonly omitted or treated haphazardly by existing methodologies. We have applied it to develop an interactive prototype creativity support system called PatentProducist to generate the novel ideas using patents.

### 2.1 System Architecture

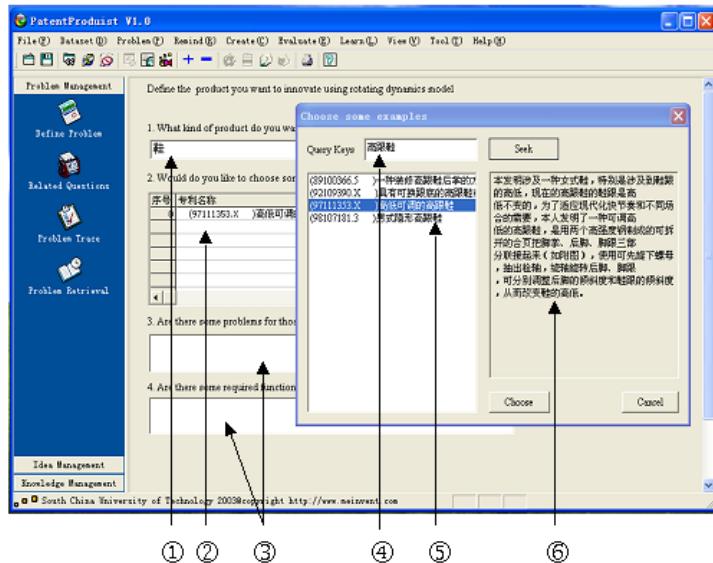
The system is structured into several parts, as shown as Fig.1, where problem definition, reminding, creating, evaluating, and visualizing modules serve as to generate creative ideas. Patent Search Engine searches and captures recent patent abstracts and full text into the local patent database. Learning module aims at acquiring inventive knowledge, construction of patent tree and domain knowledge either interactively or semi-automatically from patent databases using statistical approaches. The patent database consists of Chinese patents, which supplies a large amount of text data for the system to remind and create new ideas using natural language processing techniques and computational creativity techniques. The knowledge base involves knowledge such as Inventive principles for elimination of engineering conflicts, inventive standards, etc. These kinds of knowledge are summarized from TRIZ Theory.



**Fig. 1.** Architecture of System PatentProducist

### 2.2 Problem Definition

Problem definition module provides the interface for the user to define the problem they want to solve. It aims to define the problem clearly in a concise and simple manner. Users just input the text name of the object they want to design, and choose some existing exemplars similar to the problem from the patent database by searching on several keywords. If he does not become conscious of words



**Fig. 2.** Definition of the problem to be solved

that represent desired features or characteristics of problem, he can push the “search” button to choose some examples from patent database. From that user enters the system shown as Fig.2, system will generate new ideas about shoes based on the defined problem and chosen patent example. Define the product name user want to generate new ideas, for example, user enter the product name: “鞋” (Shoes), and then system know the user is willing to create novel shoes.k-Existing patent examples list chosen from patent database. For example, existing patent ”高度可调整的高跟鞋” (high heel shoe with adjustable height) is chosen here. This is very useful when users do not know how to describe the product they want to create.!The keywords user applies to seek for patent examples from patent database. m Existing patent examples satisfying the keywords. The abstract corresponding to the focused patent item of the left list. For advanced users, they can describe the disadvantages of existing products and the ideal functions they want in these two blanks.

### 2.3 Reminding Process

Innovative solutions often result from the selective combination of elements previously seen as unrelated, but experience suggests that when a person views the world from one paradigm, it can be very difficult to imagine it from another. Reminding module aims at providing some remote but logically related objects to the defined problem. It emphasizes the simulation of creative association of human being to choose patents . Thus reminding differs from retrieval in that it is possibly recursive (transitive). It also differs from discovery in that for different

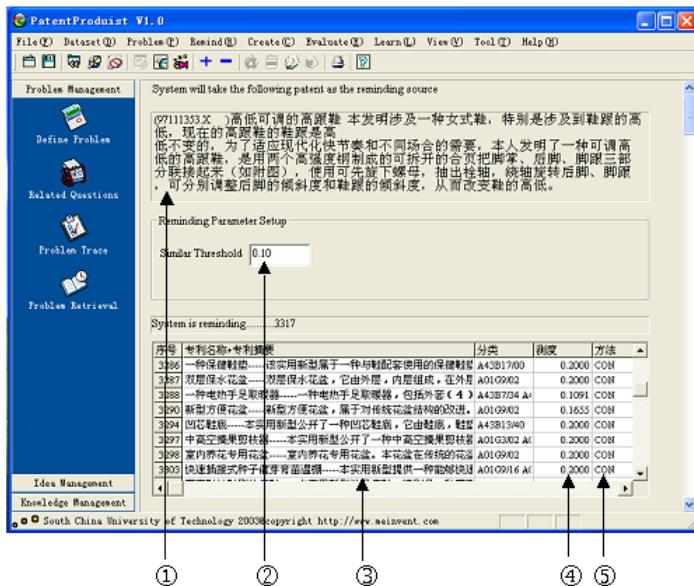
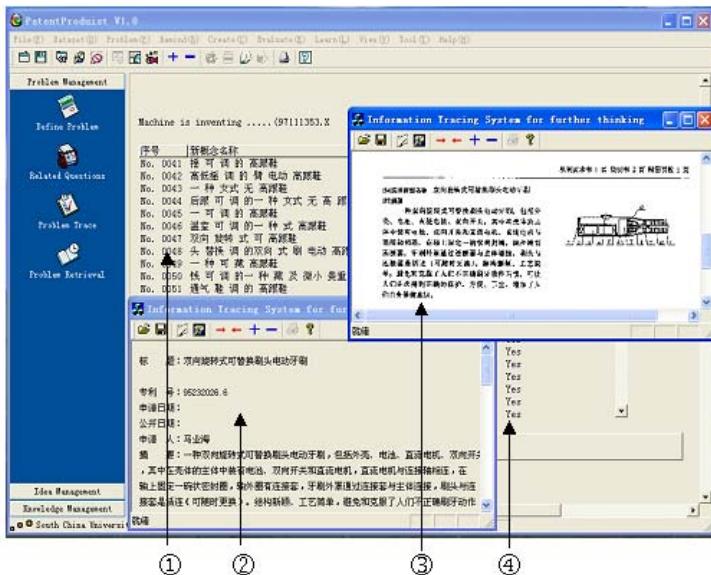


Fig. 3. Reminding process

problems the same patent object may be reminded. Currently several different creative reminding algorithms have been implemented and integrated [15,16,17,18,19,20]. These reminding approaches automatically remind patents that are formed among disparate elements from patent database, which can be utilized to generate new ideas, shown as Fig.3 The system reminds some interesting patents according to "高度可调整的高跟鞋" (high heel shoe with adjustable height) such as (only patent name listed) 1) 一种电热手足取暖器 (An electronic device providing heat for hand and foot) 2) 双向旋转式可替换刷头电动牙刷 (An electronic toothbrush with replaceable brush head that can rotate along two directions).

## 2.4 Creating Process

Inventing module concentrates on how to formalize inventive techniques and creative thinking methods to produce inventive solution to a given problem. This is performed based on the reminded patents. Currently the system is designed according to interactive genetic algorithm. It integrates all kinds of creative methods such as combination and transformation to generate alternatives for the problem[20,21,23,22]. During the process, novelty is utilized to judge the generated ideas, which is automatically determined by a comparison of the new idea with existing ones. These newly generated ideas are simply ranked by increasing order of the values of the similarity measure function. But human can modify this rank and directly choose the good ideas for the next generation. The



**Fig. 4.** Creating process

system generates new ideas about shoe according to the reminded patents, shown as Fig.4. Some interesting ideas are listed as follows:1) 遥可调的高跟鞋 (Remote adjustable high-heel shoe)2) 双向旋转式可高跟鞋 (Two directions rotating style high-heel shoe) 3) 温室可调的一种式高跟鞋 (Temperature adjustable one kind of high). Obviously these new ideas are not complete sentence, but they are good creative stimuli for further thinking.

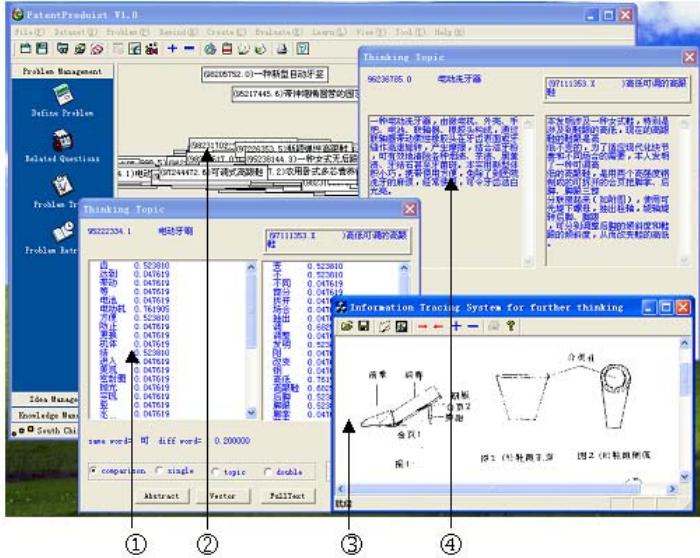
## 2.5 Evaluating Process

Currently to select the better ones from many created ideas cannot be easily finished automatically. This is because different ideas may have different degrees of interestingness to the user, and the same user may be interested in different things at different point in time. Accordingly it is desirable that a system only gives the user the set of interesting ideas and ranks them according to their degree of interestingness. PatentProducist can automatically perform the similarity-based comparison between new generated ideas and the existing ones. It also provides the tool for tracing the history of idea generation as well as for watching full text of patents when necessary. User can also modify the evaluation of each individual the system gives.

## 2.6 Visualization

Visual interfaces may significantly reduce the cognitive load of users when working with complex structures. Visualizing module aims to provide such a Visual

interface for the set of reminded patent texts and created new ideas. Because text vector space has a very high dimensionality and is also curved, it is visualized using local linear embedding approach[14]. The users can interact with this space through tools provided by the system to generate more new ideas.

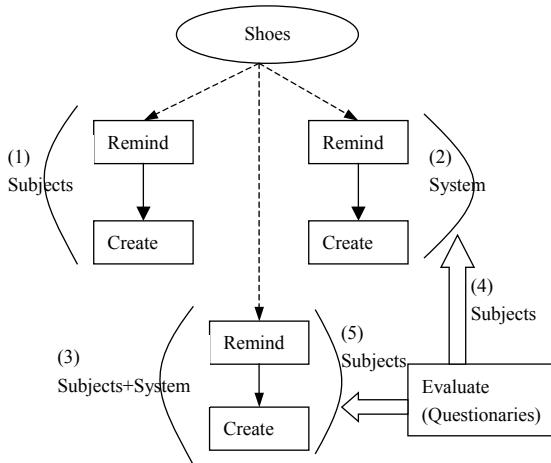


**Fig. 5.** Visualization interface of reminded patents space

Fig.5 is a visualized reminding space that consists of projected reminded patent vectors. In the space, every icon stands for a projected patent, which is composed of patent identifier and patent name that can be applied to help users understand the semantic cluster structure of reminded patents. User can reconfigure this space by moving the icon, but deleting and adding actions are not allowed. User can choose any icon in the space to watch its text vector, full text (in image format) as well as Figures, and patent abstract. In this way, user can understand why some patents are reminded around the given problem.

### 3 Experimental Evaluation

In order to investigate whether the system can generate creative ideas automatically and whether it can help subjects find new creative ideas, experiments were conducted to evaluate the system with regard to producing a set of novel ideas. 15 subjects took part in these experiments. They came from a pool of volunteer postgraduate students enrolled in our computer department. Some subjects did not take part in the whole test.



**Fig. 6.** Experiment steps for comparison

### 3.1 Experiment Design

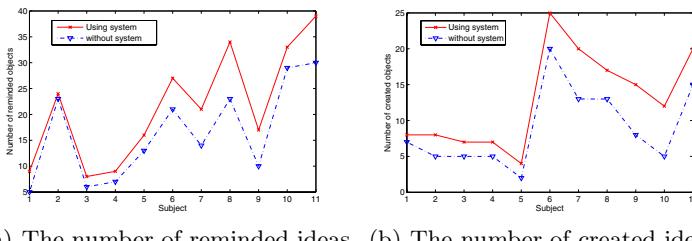
The experiments are performed on a small collection of 4000 Chinese-language patents. The problem in experiments to be solved is “Produce a list of as many new style of shoes as possible”. The subjects were specifically told to write down what did not already exist and they would personally like to wear. This problem was chosen for a number of reasons. First, most people have had some experience with it and requires no specialized knowledge in order to develop new ideas concerning them. Second, the task is open-ended. Third, the creativity of the ideas could be simply judged by a group of people who are not required to have specialized knowledge or training. Finally, while the basic problem is simple, it does emulate some real world problems. Experiments are performed in four steps, shown as Fig.6.

In step (1), subjects are required to remind and create as many as possible novel objects related to shoes independently, without any help of the system. Step (2) let the system automatically remind objects related to shoes and create novel ideas for shoes. Subjects are not required to interact with the system. In step (3), subjects are required to remind and create new objects through interaction with the system, where the system visualizes the space that consists of objects that the system automatically reminded and created. Finally subjects evaluate the results of step (2) and step (3) in step (4) and step (5) respectively. Evaluation is performed through questionnaires that are designed in terms of idea fluency and novelty. Idea fluency refers to the ability to generate a large number of ideas in response to a given creative task. Novelty was defined as the originality and uniqueness of an idea. Due to limited space, questionnaires and the corresponding answers are not provided here.

### 3.2 Experimental Results

We begin with the results achieved in step (1) and step (3) in experiments. It can be observed from Fig.7 that in all cases ideas that subjects reminded and created with help of the system is more than that they did independently, where only 11 subjects were selected as they took part in the whole experiments. It can be also discovered that the more a subject reminded and created objects independently, the more they did using the system and the more obvious the difference between the two cases is. This indicates that the system is beneficial to those who are positively engaged in doing creative activities, by assisting them to remind and create more novel ideas.

Now we turn to analyze the answers to the questionnaires that subjects finished in step (4) and step (5) to evaluate the system. It can be concluded from Table.1 that the system not only automatically reminds and generate many or some interesting ideas, it also assists a user to remind and create more useful ideas. Surprisingly, subjects give lower evaluation to the ability of the system to help a user but higher evaluation to the ability of the system to automatically remind and generate interesting ideas. This is possibly due to that the system provides only useful stimuli, a user is still required to creatively think to generate novel ideas. Simultaneously, the ideas that the system generates are excluded. In the process of the experiment, two subjects did not take part in step (1), who gave most “few” answers to the questions. Actually step (1) was designed to make evaluation as objective as possible. Another subject gave the answer-“few” to all questions. We discovered that this subject did not understand the questions clearly because what he reminded and created independently and what he did using the system are totally not related to topic: “shoes”. It can be concluded that the experiments present evidence to support our hypothesis



(a) The number of reminded ideas (b) The number of created ideas

**Fig. 7.** Comparison between two cases

**Table 1.** Statistical evaluation results to effects of the system (%)

No. Questions	Much	Some	few
1 Can the system remind interesting patents?	30	68	2
2 Can the system create interesting new ideas?	28	67	5
3 Can the system assist a user to remind more interesting objects?	11	68	21
4 Can the system assist a user to create more interesting new ideas?	18	68	14

that the system can automatically or assist a user generate some creative ideas through the patents. It is convincing that it is worthwhile to pursue research in this direction.

## 4 Conclusion

This paper describes a system for assisting human creative activities through patents. By comparison with existing systems, this system is characterized as follows. The system directly utilizes the patents in database using natural language techniques so that knowledge acquisition is not necessarily required. It can automatically reminds related objects in terms of multiple semantic relationships instead of single similarity. It automatically generates the novel ideas from those reminded objects, instead of the text fragments that a user input to describe his/her thinking process. Finally the system implements the advanced visualization techniques so that the highly curved text manifold can be visualized effectively. In the future, a specific application domain such as the shoe industry will be investigated deeply.

## Acknowledgement

This work is supported by Natural Science Foundation of China(60003019), Hubei Science and Technology Project (2005AA101C17),and UK Engineering and Physical Sciences Research Council under grant number GR/N15764/01.

## References

1. Eric L. Santanen, Robert O. Briggs, Gert-Jan de Vreede: The Impact of Stimulus Diversity on Creative Solution Generation: An Evaluation of the Cognitive Network Model of Creativity. Proceedings of the 36th Hawaii International Conference on System Sciences, 2003
2. Eric L. Santanen: Creative Approaches to Measuring Creativity: Comparing the Effectiveness, of Four Divergence thinkLets. Proceedings of the 37th Hawaii International Conference on System Sciences, 2004
3. Sugimoto, M., Hori, K., and Ohsuga, S.: A system for visualizing viewpoints and its application to intelligent activity support. IEEE Transactions on Systems, Man, and Cybernetics,**28**(1998)124 -136
4. Takashi Hayashi, Masafumi Hagiwara: Idea Divergent Editor Using Analogy: The IDEA System, International Journal of Intelligent Systems. International Journal of Intelligent Systems, **18**(2003)1155-1172
5. Yasuyuki Sumi, Kori K, and Setsuo Ohsuga: Computer-aided thinking by mapping text-objects into metric space. Artificial Intelligence, **91**(1997)119-155
6. Tomohiro Shigenobu, Takashi Yoshino, and Jun Munemori: Idea Generation Support System GUNGEN DX II beyond Papers. V. Palade, R.J. Howlett, and L.C. Jain (Eds.): KES 2003, LNAI,**2774**(2003)741-747
7. Elias Carayannis and John Coleman: Creative system design methodologies: the case of complex technical systems. Technovation, **25**(2005)831-840

8. Jami J. Shah, Noe Vargas-Hernandez, Steve M. Smith: Metrics for measuring ideation effectiveness. *Design Studies*, **24**(2003)111-134
9. Wen Guihua, Zheng Qilun, Ding Yuehua: Rotating dynamics framework for computational creativity. *Journal of system simulation*, **36**(2004)
10. Wen Guihua, Zheng Qilun, Ding Yuehua: Rotating dynamics framework for computational creativity. Proceedings of national Conference on Artificial Intelligence (Best Paper), Beijing, China(2001)
11. Wen Guihua: Rotating dynamics for computational creativity. National Defence Industry Press, BeiJing(2005)
12. Guihua WEN and Nigel Shadbolt: Fundamental Laws of Dynamics for Computational Creativity. Computational Creativity Workshop at 19th International Joint Conference on Artificial Intelligence, Edinburgh, United Kingdom(2005)
13. Ben Shneiderman: Creating creativity: user interfaces for supporting innovation. *ACM Transactions on Computer-Human Interaction*, **7**(2000) 114-138
14. S.T. Roweis and L.K. Saul: Nonlinear Dimensionality Reduction by Locally Linear Embedding. *Science*,**290**(2000)2323-2326
15. Wen Guihua et al: A Creative reminding algorithm CRA based on the information field relevance. *Pattern recognition and Artificial intelligence*,**4**(1999) 393-401
16. Wen Guihua et al.: A Reminding model based on the contrary concept. *Journal of computer research and development*,**36**(1999) 982-987
17. Wen Guihua et al.: A simulated annealing-based reminding algorithm. *Journal of computer research and development*,**36**(1999) 6-9
18. Wen Guihua et al.: An Integrated Creative Reminding Algorithm. Proceedings of 2000 IEEE international conference on system, man and cybernetics, Tennessee, USA(2000)
19. Wen Guihua et al.: Analyses of the star-decomposable reminding process. *High Technology Letters*,**12**(2002)12-16
20. Wen Guihua et al.: Theoretical Analysis of Inventive Methods. Proceedings of 2001 IEEE international conference on system, man and cybernetics,Tucson, AZ, USA(2001)
21. Oded Maimon, R.Horowitz: Sufficient Conditions for Inventive Solutions. *EEE Transactions on System Man and Cybernetics*,**29**(1999) 349-361
22. Sung-Bae Cho: Towards Creative Evolutionary Systems with Interactive Genetic Algorithm. *Applied Intelligence*,**16**(2002)129-138
23. Wen Guihua et al.: An Evolution Model for the Creative Design. Proceedings of 2000 IEEE international conference on system, man and cybernetics, Tennessee, USA(2000)

# An Improved Multiobjective Evolutionary Algorithm Based on Dominating Tree

Chuan Shi<sup>1,2</sup>, Qingyong Li<sup>1,2,3</sup>, Zhiyong Zhang<sup>1,2</sup>, and Zhongzhi Shi<sup>1</sup>

<sup>1</sup> Key Laboratory of Intelligent Information Process

Institute of Computing Technology Chinese Academy of Science

<sup>2</sup> Graduate University of the Chinese Academy of Sciences

<sup>3</sup> School of Computer and Information Technology, Beijing Jiaotong University  
{shic, liqy, zhangzy, shizz}@ics.ict.ac.cn

**Abstract.** There has emerged a surge of research activity on multiobjective optimization using evolutionary computation in recent years and a number of well performing algorithms have been published. The quick and highly efficient multiobjective evolutionary algorithm based on dominating tree has been criticized mainly for its restricted elite archive and absence of density estimation. This paper improves the algorithm in these two aspects. The nearest distance between the node and other nodes in the same sibling chain is used as its density estimation; the population growing and declining strategies are proposed to avoid the retreating and shrinking phenomenon caused by the restricted elite archive. The simulation results show that the improved algorithm is able to maintain a better spread of solutions and converge better in the obtained nondominated front compared with NSGA-II, SPEA2 and the original algorithm for most test functions.

## 1 Introduction

Multiobjective optimization problems (MOPs) are those problems that involve simultaneous optimization of more than two objectives (often competing) and usually there is no single optimal solution [1]. It is usually difficult or even impossible to assign priorities as in single objective optimization problems (SOPs). This makes an algorithm returning a set of promising solutions preferable to an algorithm returning only one solution based on some weighting of the objectives. For this reason, there has been an increasing interest in applying evolutionary algorithm (EA) to MOPs in the past ten years.

Over the past decade, a number of multiobjective evolutionary algorithms (MOEAs) have been suggested [2-5]. These MOEAs use Pareto dominance to guide the search, and return a set of nondominated solutions as result. Unlike in single objective optimization, there are two goals in a multiobjective optimization: 1) convergence to the Pareto optimal set and 2) maintenance of diversity in solutions of the Pareto optimal set [4]. Many strategies and methods are introduced to overcome these two difficulties existing in MOPs [7]. These algorithms perform well in some benchmark problems. However, there are some disadvantages existing in current MOEAs. Many MOEAs are

intricate. To achieve better solutions, many methods and strategies are used and many parameters need to be adjusted according to experiences and prior knowledge of the given problems. On the other hand, many MOEAs have high computation complexity. So designing a simple, yet efficient, MOEA is desirable.

We have proposed a simple and highly efficient multiobjective evolutionary algorithm [6]. This algorithm uses a novel fitness assignment: tree structure, which is a binary tree with the dominating information of individuals. The tree structure is called dominating tree in this paper. Moreover, the algorithm is called the multiobjective evolutionary algorithm based on dominating tree (DTEA). The experiment shows that DTEA can converge to the Pareto front, maintains the diversity of the population and cost less time. However, the algorithm has a restricted elite archive, which can cause oscillating/retreating phenomenon in the Pareto front; and it is also criticized that there are no density estimation. In this paper, we improve DTEA in these two aspects. The simulation experiment shows that the improved algorithm can obtain better solutions and the active elite archive can improve the performance of algorithms indeed.

The rest of the paper is organized as follows: in the following section, we review the related work. Section 3 explains the improvement of DTEA in detail; and then the simulation results are given in the section 4; at last, the section 5 makes a conclusion.

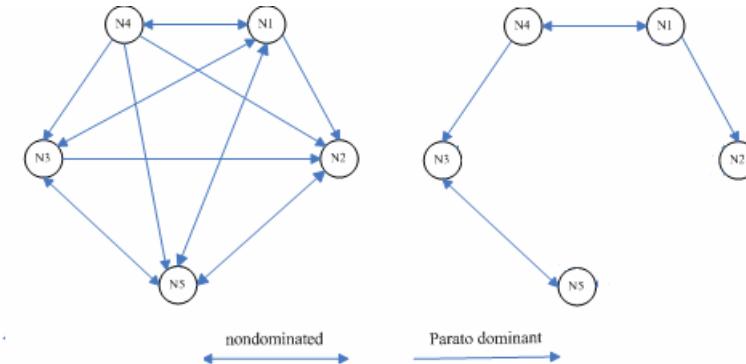
## 2 Related Work

There are some new developments of MOEAs in recent years. NSGA-II [4] is advanced from its origin, NSGA [2]. In NSGA-II, a nondominated sorting approach is used for each individual to create a Pareto rank, and a crowding distance assignment method is applied to implement density estimation. Similar to NSGA-II, SPEA2 [5] is an enhanced version of SPEA [3]. In SPEA2, each individual in both the main population and the elitist archive is assigned a strength value, which incorporates both dominance and density information. Meanwhile, a  $k$ th nearest neighbor density estimation method is applied to obtain the density value of each individual.

Recent research clearly shows that the elitism can speed up the performance of MOEAs significantly, and it helps to prevent the loss of good solutions once they have been found [8]. Fieldsend points out a consequence of restricting the number of solutions in the elite front can be shrinking and oscillating/retreating estimated Pareto front [9]. A remedy to this situation is simply to retain all the nondominated solutions found (as an active input to the continuing search process) [9].

Shi has introduced a simple and highly efficient multiobjective evolutionary algorithm [6]. Ref [6] has pointed out that there are many redundant comparisons in the former fitness assignments that is the main time-consuming part in MOEAs, and then proposed a novel tree structure (dominating tree) to preserves the necessary dominating relationships among individuals. As shown in Fig.1, the dominating tree maintains the necessary relationships among individuals; however, it uses fewer comparisons. The dominating tree is a binary tree; its child point links to its left subtree whose root is dominated by that node; and its lsibling and rsibling point link to its right subtree whose root is nondominated with that node (lsibling and rsibling point constitute a bidirection list actually). The tree's sibling chain refers to the chain constituted by the root and the root nodes of its right subtrees. Taking the right figure of Fig.1 for example, N1 and N4

is the sibling chain of the tree whose root is N4. Similarly, N3 and N5 also constitute a sibling chain. According to the construction algorithm of the tree, a tree's sibling chain contains all Pareto optimal nodes in the tree, and the nodes in the same sibling chain are sorted in their depth declining order. Without special density estimation, DTEA deletes the worst node, which maintain the diversity naturally and cost less time (the worst node is the leaf node of the node with the largest depth or the first node of the sibling chain of the tree when all nodes are nondominated). The experiment shows that the algorithm can converge to the Pareto optimal front, and maintains the diversity of population for the test problems [6].



**Fig. 1.** Two different fitness assignments. The left figure compare individuals with graph sufficiently. The right figure compare individuals with dominating tree.

### 3 The Improvement of DTEA

Although DTEA perform well for many test functions, there are some disadvantages in this algorithm. The main criticisms of DTEA approach have been as follows.

- 1) There is no density estimation in DTEA. Although DTEA does well in maintaining the diversity for many problems [6], the diversity of DTEA is worse than that of some algorithms with the density estimation in some problems. To improve its distribution, it is a good approach to add the density estimation to DTEA.
- 2) Recent work has showed that the restricting number of solutions in the elite front can result in shrinking [5] and oscillating/retreating estimated Pareto front [10], so some researches highlight the use of an active archive of elite to improve the optimization speed of these algorithms [9, 11]. In DTEA, the elite archive is active in the forepart of the evolutionary process, but it is restricted after all individuals are nondominated for the first time. So DTEA can't avoid shrinking and oscillating/retreating phenomenon.

To overcome these criticisms, we propose the improved version of DTEA, which is called EDTEA in this paper.

### 3.1 Density Estimation

The eliminating strategy used in DTEA maintains the population diversity naturally. A node with the large depth value means that there are many nodes in its neighborhood; deleting the leaf node of the node with the largest depth forces the nodes in its neighborhood to leave this district, which makes solutions distribute in the objective space evenly. Although DTEA maintains the diversity well for many test functions, the diversity estimation is necessary to enhance the distribution of the solutions in DTEA. The essential idea is that each node in the sibling chain are sorted by their depth values first and then by their density values when their depth values are same. In this way, the fitness of a node is decided by its dominance information (depth value) and density information (density value).

To achieve the goal, we change the tree structure first to add the density information. We add two new fields: nid and distance to express the diversity information of the node. nid records the identity of the node nearest to the given node in the sibling chain; and distance records their distance (the distance is Euclidean distance).

The changed tree structure is described as follows:

```
struct node{
    int id; //the ID of the given node
    int depth; // the number of the left children nodes of the given node (including
               // itself)
    int nid; //the ID of the node nearest to the given node
    int distance; // the nearest distance between the given node and the nearest nodes
    struct node *child; //a pointer that points to the node worse than the given node
    struct node *lsibling, *rsibling; // a pointer that points to the node nondominated
                                    // with the given node
}
```

Many density estimation methods have been proposed [7]. Deb has used the density-estimation metric and presented the crowded-comparison operator in NSGA-II [4]. Zitzler has used a  $k$ th nearest neighbor density estimation method in SPEA2 [5]. In our algorithm, we calculate the nearest Euclid distance in the objective space between two nodes in the same sibling chain as the their density estimation. This distance reflects the node's diversity information to some extend, and can be calculated easily.

Secondly, we need estimate density information and sort nodes in the sibling chain again when a new node is inserted into or deleted from the sibling chain. When a node is inserted into or deleted from the tree, the original operation in the construction algorithm of the tree don't change except that we must calculate the diversity information and sort nodes in the changed sibling chain according to their depth values and density information.

When a new node is inserted into a sibling chain; it will change density value of nodes in the sibling chain. It can be handled as the following step: 1) calculate the distance between the new node and other nodes, and select the minimum one as its density value; 2) for each node in the sibling chain (pnode), if the distance between the new node and pnode is smaller than its original distance, the new distance takes the

place of the old one; 3) sort the nodes in the sibling chain in their depth descending order firstly, and in the distance increasing order secondly when their depth values are same.

When a node is deleted from the sibling chain, it also changes the density information of nodes in the sibling chain. It can be handled as the following step: 1) calculate the nearest distance of the nodes whose nearest node is the deleted node again; 2) sort the nodes in the sibling chain in their depth descending order firstly, and in the distance increasing order secondly when their depth values are same.

In this way, the node dominated by many nodes have more possibility to be deleted, and the node in the crowded niching also have more possibility to be deleted when the nodes are nondominated.

### 3.2 Active Archive of Elite

When the number of Pareto optimal individuals is smaller than the population size that is restricted in DTEA, the archive of elite is active; all optimal individuals are stored in the sibling chain of the tree. However, when all individuals are nondominated and the new generated individual is nondominated with the former individuals, DTEA add the new generated individual into the sibling chain and delete the first one in the sibling chain. The deleted one may be a more promising one than the added one, so the oscillating/retreating phenomenon will happen.

To overcome this disadvantage, we make the elite archive active. The key idea is that the tree's sibling chain maintains all Pareto optimal individuals found. When all individuals are nondominated and the new individual is nondominated with all former individuals, the elite archive becomes larger and the population size also becomes larger. However, it will cost too much time when the population size is too large, so we must design the population growing and declining strategy. Yen also have designed the population growing and declining strategy to determine if an individual will survive or be eliminated [11].

To maintain all elite individuals, the population growing strategy is that we don't delete the worst node when the new generated individual is nondominated with former individuals and the worst individual is also in the elite archive. The tree's sibling chain maintains all elite individuals that have been found. When the number of elite individuals exceeds the initial population size, the population size also increases.

However, the large population size has some disadvantages: too much time consuming and storing memory [9], so we must design the declining strategy to control the population size. When the worst node is not in the elite archive, we add and delete a node at the same time, so the population size don't change. However, the population growing strategy increases the population size. We find that the optimal front may have no good distribution when the elite archive becomes larger: some front districts are crowded, but some front districts are sparse. To obtain the good distribution, we must delete the crowded individuals in the optimal front. The declining strategy is that the deleted nodes are the nodes whose density value is smaller than the  $\delta$  times of the mean density value of the nodes in the sibling chain when all individuals are nondominated; namely the distance of the deleted node satisfies the following

relationship ( $d$  is the deleted node's distance,  $\bar{d}$  is the mean distance of the nodes in the same sibling chain.).

$$d < \delta * \bar{d} \quad (\delta \in [0, 1]) \quad (1)$$

The parameter  $\delta$  need be settled according to the problem. The declining strategy decreases the population size and makes the population distribute evenly.

The population growing strategy makes the algorithm maintain all nondominated individuals found to avoid the oscillating/retreating phenomenon. The population declining strategy controls the population size to avoid too much cost of the running time and memory, and make the solutions distribute evenly.

## 4 Simulation Results

In order to validate EDTEA, we compare EDTEA with other two advanced MOEAs: SPEA2, NSGA-II and the original DTEA. The SPEA2 and NSGA-II are implemented in C according to their description in the literatures [4], [5] and PISA [12]. The experiments are carried out on a 3GHz and 524MRAM Pentium IV computer running Windows 2000.

In the experiments, we use six well-known test functions. In addition to the standard ZDT test functions that are two objectives [13], we also use three test functions with three objectives: DTLZ1-DTLZ3 [14]. The detailed information and comments can be found in the Ref [13,14] respectively.

The ZDT test functions involve two objectives and have the following structure:

Test function ZDT1-ZDT3

Minimize:  $T(\mathbf{x}) = (f_1(x_1), f_2(\mathbf{x}))$

Where  $f_2(\mathbf{x}) = g(x_2, \dots, x_p)h(f_1(x_1), g(x_2, \dots, x_p))$  (2)

And  $\mathbf{x} = (x_1, \dots, x_p)$

Test function DTLZ1

$$\begin{aligned} f_1(\mathbf{x}) &= \frac{1}{2}x_1x_2(1+g(\mathbf{x}_M)) \\ f_2(\mathbf{x}) &= \frac{1}{2}x_1(1-x_2)(1+g(\mathbf{x}_M)) \\ f_3(\mathbf{x}) &= \frac{1}{2}(1-x_1)(1+g(\mathbf{x}_M)) \\ g(\mathbf{x}_M) &= 100(|\mathbf{x}_M| + \sum_{x_i \in \mathbf{x}_M} ((x_i - 0.5)^2 - \cos(20\pi(x_i - 0.5)))) \end{aligned} \quad (3)$$

Test function DTLZ2

$$\begin{aligned} f_1(\mathbf{x}) &= (1+g(\mathbf{x}_M))\cos(x_1\pi/2)\cos(x_2\pi/2) \\ f_2(\mathbf{x}) &= (1+g(\mathbf{x}_M))\cos(x_1\pi/2)\sin(x_2\pi/2) \\ f_3(\mathbf{x}) &= (1+g(\mathbf{x}_M))(\sin(x_1\pi/2)) \\ g(\mathbf{x}_M) &= \sum_{x_i \in \mathbf{x}_M} (x_i - 0.5)^2 \end{aligned} \quad (4)$$

## Test function DTLZ3

$$\begin{aligned}
 f_1(x) &= (1 + g(x_M)) \cos(x_1\pi/2) \cos(x_2\pi/2) \\
 f_2(x) &= (1 + g(x_M)) \cos(x_1\pi/2) \sin(x_2\pi/2) \\
 f_3(x) &= (1 + g(x_M)) (\sin(x_1\pi/2)) \\
 g(x_M) &= 100(|x_M| + \sum_{x_i \in X_M} ((x_i - 0.5)^2 - \cos(20\pi(x_i - 0.5)))
 \end{aligned} \tag{5}$$

To fairly compare these four algorithms, they all use the simulated binary crossover (SBX) and polynomial mutation [15]. The crossover and mutation operators are  $\eta_c = 20$  and  $\eta_m = 20$  respectively. The crossover probability is 1, and the mutation probability is  $1/n$  for all test functions, where  $n$  is the number of decision variables. The population size is 100; the archive size is 100 in SPEA2 and NSGA-II. To compare the different algorithms fairly, the same number of individuals is evaluated for all algorithms. The numbers of the evaluated individuals are settled as follows: ZDT1, ZDT2 and ZDT3 are 25000; DTLZ1 and DTLZ2 are 30000; and DTLZ3 is 50000. The running generation of SPEA2 and NSGA-II equals the number of the evaluated individuals dividing the population size. The running generation of DTEA and EDTEA equals the number of the evaluated individuals dividing 2, since SBX generates two offspring in one generation. The parameter  $\delta$  is settled as 0.4 in EDTEA.

Each function is calculated 30 times, and then save the Pareto optimal set in the current population after each calculation. These Pareto optimal sets are combined to form a new set, and then calculate the Pareto optimal set in the new set as result. The aim is to reduce the random influence. The final results of different MOEAs are compared according to the defined criterions. The running time is the average time of the main loop.

Two popular measure criterions are used in this paper.  $C(X, Y)$  is used to compare the convergence of two Pareto optimal sets. It is used in some papers [13]. The value  $C(X, Y) = 1$  means that all solutions in  $Y$  are Pareto dominated by solutions in  $X$ . The opposite,  $C(X, Y) = 0$  represents the situation when none of the solutions in  $Y$  are covered by the set  $X$ . Noted that both  $C(X, Y)$  and  $C(Y, X)$  have to be considered, since  $C(Y, X)$  is not necessarily equal to  $1 - C(X, Y)$ .  $\Delta$  measures the extent of spread achieved among the obtained solutions. It is used in [16]. A good distribution would make  $\Delta$  equal to 0. These two measure criterions are independent of the true Pareto optimal front. They can reflect the quality of solutions to some extend. They also can be used in the high-dimension objective space.

Table 1 shows the comparing result of the convergence metric  $C$  obtained using four MOEAs. We not only compare DTEA with SPEA2 and NSGA-II by metric  $C$  (the last four rows), but also compare EDTEA with other three algorithms (the first six rows). Through comparing EDTEA with DTEA (the first row and fourth row), we found that EDTEA is able to converge better than DTEA significantly. For most test functions, it is difficult to say that DTEA converges better than SPEA2 and NSGA-II, but EDTEA obtains the better results than them. Through the comparison in convergence metric, the active elite archive improves the DTEA's performance indeed.

**Table 1.** Convergence comparison of different algorithms using  $C$ . E is EDTEA; D is DTEA; S is SPEA2; and N is NSGA-II.

	ZDT1	ZDT2	ZDT3	DTLZ1	DTLZ2	DTLZ3
$C(E, D)$	0.881	0.812	0.912	0.924	0.916	0.941
$C(E, S)$	0.742	0.663	0.952	0.582	0.663	0.631
$C(E, N)$	0.781	0.715	0.931	0.623	0.583	0.686
$C(D, E)$	0.003	0.004	0.000	0.013	0.024	0.017
$C(S, E)$	0.011	0.013	0.004	0.352	0.013	0.009
$C(N, E)$	0.030	0.021	0.002	0.481	0.005	0.010
$C(D, S)$	0.085	0.084	0.841	0.016	0.089	0.019
$C(D, N)$	0.054	0.041	0.824	0.006	0.118	0.019
$C(S, D)$	0.032	0.061	0.120	0.708	0.101	0.060
$C(N, D)$	0.070	0.061	0.123	0.572	0.081	0.056

**Table 2.** Distribution comparison of different algorithms using  $\Delta$ 

	ZDT1	ZDT2	ZDT3	DTLZ1	DTLZ2	DTLZ3
EDTEA	0.0009	0.0021	0.0028	0.102	0.158	0.201
DTEA	0.0031	0.0040	0.0075	0.137	0.235	0.249
SPEA2	0.0021	0.0038	0.0020	0.116	0.246	0.268
NSGA-II	0.0019	0.0044	0.0020	0.130	0.233	0.253

Table 2 shows the comparing result of the distribution metric  $\Delta$  using different algorithms. The distribution of EDTEA is better than that of DTEA for all test functions. For most test functions, the distribution of EDTEA is better than that of SPEA2 and NSGA-II. The experiment results show that the density estimation and the population controlling strategy make the solutions distribute evenly in the objective space.

Table 3 shows the average running time of these three MOEAs. The running times of DTEA are obviously lower than that of other algorithms for all test functions. The running time of EDTEA is larger than that of DTEA, but it is still smaller than that of SPEA2 and NSGA-II for the most test functions. Because the density estimation is used in EDTEA, the running time of EDTEA is larger than that of DTEA. The size of the elite archive becomes larger, it also cost more time when an individual is inserted into the tree. However, a new node only needs to calculate the density value with the nodes in the same sibling chain, so the running time is still smaller than that of SPEA2 and NSGA-II for most test functions.

**Table 3.** Compare running times of different algorithms. The time unit is millisecond.

	ZDT1	ZDT2	ZDT3	DTLZ1	DTLZ2	DTLZ3
EDTEA	1860	2130	1965	2352	2056	2816
DTEA	978	961	988	619	1083	1105
SPEA2	7848	7806	7828	10333	10630	17611
NSGA-II	2006	2444	2006	2039	2553	3502

In the experiment, the parameter  $\delta$  is settled as 0.4. We find that the algorithm is not sensitive to the parameter. The algorithm will obtain the good and steady solutions when the bound of  $\delta$  is from 0.3 to 0.5.

## 5 Conclusions

Recent research has indicated that the restricted elite archive and the absence of density estimate limit the potential of DTEA proposed by Shi [6]. The algorithm adds the density estimation, namely the nearest Euclid distance in the objective space between a node and other nodes in the same sibling chain is used as the density estimation. The nodes in the same sibling chain are sorted by their depth value first and then by their density estimation. All elite individuals are stored to avoid oscillating/retreating estimated Pareto front. The improved algorithm also designs the population growing and declining strategies to balance the performance and the cost of time and space. Six well-known test problems borrowed from the literatures are used in the experiment. For all test functions, the improved algorithm enhances its performance significantly. Comparing with NSGA-II and SPEA2, the improved algorithm is able to find the solutions that maintain a better distribution and converge better to the optimal front for most test functions. The experiment indicates the population growing and declining strategies and density estimation method are effective; it also indicates that the active elite archive can improve the DTEA's performance indeed.

## Acknowledgments

This work is supported by the National Science Foundation of China No. 60435010, National Basic Research Priorities Programme No. 2003CB317004 and the Nature Science Foundation of Beijing No. 4052025.

## References

1. Deb, K.: Multiobjective Optimization using Evolutionary Algorithms. U.K: Wiley, 2001.
2. Srinivas, N., Deb, K.: Multiobjective optimization using nondominated sorting in genetic algorithms. *Evol. Comput.*, vol.2, no.3, pp.221-248, 1995.
3. Zitzler,E., Thiele. L.: Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach. *IEEE Trans. Evol. Comp.*, vol.3, no 4, pp.257-271, 1999.

4. Deb, K., Pratab, A., Agarwal, S., MeyArivan, T.: A fast and Elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.*, vol.6, pp. 182-197, Apr.2002.
5. Zitzler, E., Laumanns, M., Thiele, L.: SPEA2: Improving the Strength Pareto Evolutionary Algorithm. TIK-Report 103, 2001, ETH Zentrum, Gloriastrasse 35, CH-8092 Zurich, Switzerland.
6. Shi, C., Li, Y., Kang, L.S.. A New Simple and Highly Efficient Multiobjective Optimal Evolutionary Algorithm. *Proceedings of 2003 IEEE Conference on Evolutionary Computation*, Australia, 2003, pp.1536- 1542.
7. Veldhuizen, D.A.V., Lamont, G.B.: Multiobjective Evolutionary Algorithms: Analyzing the State-of-the-Art. *Evol. Comput.*, vol.18, no.2, pp.125-147, 2000.
8. Zitzler, E., Deb, K., Thiele, L.: Comparison of Multiobjective Evolutionary Algorithms: Empirical Results. *Evol. Comput.*, vol.18, no.2, pp.173-195, 2000.
9. Fieldsend, J.E., Everson, R.M., Singh, S.: Using Unconstrained Elite Archives for Multiobjective Optimization. *IEEE Trans. Evol.Comput.*, vol.7, pp.305-323, Jun.2003.
10. Hanne, T.: On the convergence of multiobjective evolutionary algorithms. *Eur.J.Oper.Res.*, vol.117, pp.553-564,1999.
11. Yen, G., Lu, H.: Dynamic Multiobjective Evolutionary Algorithm: Adaptive Cell-Based Rank and Density Estimation. *IEEE Trans. Evol. Comput.*, vol.7, no3, pp. 253-274, 2003.
12. Bleuler, S., Laumanns, M., Thiele, L., Zitzler, E.: PISA- A Platform and Programming Language Independent interface for Search Algorithms. *Conference on Evolutionary Multi-Criterion Optimization (EMO 2003)*, 4/2003, pp.494-508.
13. Zitzler, E., Deb, K., Thiele, L.: Comparison of Multiobjective Evolutionary Algorithms, Empirical Results. *Evolutionary Computation*, 2000, 8(2): 173-195.
14. Deb, K., Thiele, L., Laumanns, M., Zitzler, E.: Scalable Multi-Objective Optimization Test Problems. *Proceedings of 2002 Congress on Evolutionary Computation*, vol.1, pp.825--830, May, 2002.
15. Deb, K., Agrawal, R.B.: Simulated binary crossover for continuous search space. *Complex Systems*, vol.9, pp.115-148, 1995.
16. Deb, K., Samir, A., etc: A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-objective Optimization: NSGA-II. KanGAL Report No.200001, Kanpur, PIN 208 016, India.

# Fuzzy Genetic System for Modelling Investment Portfolio

Rahib H. Abiyev and Mustafa Menekay

<sup>1</sup>Near East University, Department of Computer Engineering, Lefkosa, North Cyprus  
rahib@neu.edu.tr,  
mmenekay@neu.edu.tr

**Abstract.** The business environment is fully characterized with uncertainties. In order to minimize risk and maximize future returns proper portfolio model must be designed. In this paper the application of fuzzy theory to portfolio selection is presented. Fuzzy logic is utilized in the estimation of expected return and risk. Using fuzzy logic, managers can extract useful information and estimate expected return by using not only statistical data, but also economical and financial behaviours of the companies and business strategies. In the formulated fuzzy portfolio model, fuzzy set theory gives chance of possibility trade-off between risk and return. This is obtained by assigning satisfaction degree between criteria and constraints and defining tolerance for the constraints in order to obtain goal value in objective risk function. Using the formulated fuzzy portfolio model, a Genetic Algorithm (GA) is applied to find optimal values of risky securities. The obtained results satisfy the efficiency of the proposed method.

## 1 Introduction

The mathematical formulation of portfolio selection model was first formulated by Prof. Harry Markowitz, which is called mean-variance model. According to his model, for a given return rate, one can derive the minimum investment risk by minimizing the variance of a portfolio; or for a given risk level, one can derive the maximum returns by maximizing the expected returns of a portfolio. Based on this model mean absolute deviation portfolio optimization [2], and semiabsolute deviation portfolio optimization models [3] were proposed. These models are based on probability theory and can be effectively used by investors in cases where a sufficient amount of data can be collected. A number of researches work show limitation of using probabilistic approaches in characterizing the uncertainty in finance. When available data insufficient then the constructed models based on probability theory does not give desired results. In such situations, during construction of portfolio model there would be important not only statistical data taken from companies financial operations that determine average return and also standard deviation or variance defining risk of investment. Additionally, investor can take into account and complete his knowledge with other information, such as economical and financial behaviours of the companies, government policies, business strategies, etc. This information can be estimated quantitatively by the expert perceptions. In order to construct financial portfolio optimization model it is proposed to use fuzzy set theory, proposed by Zadeh [4].

Fuzzy set is a powerful tool used to describe an uncertain environment with vagueness, ambiguity, which is involved in finance. Some fuzzy optimization methods have been proposed in literature in order to deal with different aspects of soft constraints. Zimmerman has considered the fuzzy optimization as a symmetric problem [5,6]. In this formulation, fuzzy sets represent both the problem goals and the flexible constraints. The fuzziness arises because of definition of fuzzy maximization and the approximate inequality. These are defined by fuzzy goal and fuzzy constraints. In this formulation the fuzzy goals and the constraints are aggregated to a single function that is maximized. This framework can handle crisp constraints as well as fuzzy constraints.

A number of research works is devoted to the development of fuzzy portfolio selection models. In [7] the vague goals for expected return and risk are introduced to consider fuzzy portfolio selection problem. In [8,9] the possibility theory is applied to handle uncertainty and solve portfolio optimization problem. In this model possibility distribution is used to characterize experts' knowledge. It is identified using returns of securities associated with possibility grades offered by experts. However it is not always easy for an investor to specify them. In [10] a fuzzy approach is proposed to describe soft constraints and repair infeasibility in portfolio optimization problem. In [11]  $\alpha$  level procedure is used to solve fuzzy portfolio optimization problem. However the use of  $\alpha$  level procedure with Lagrange Multiplier method allows extension of the fuzziness of obtained results which decrease accuracy. In [12] using linear interval programming the solution of portfolio selection is considered. The portfolio modelling which is transformed to a linear interval programming model is presented. In [13] using evolutionary programming the portfolio optimization is considered. The constraint on return is fuzzified. This is provided by using flexible constraints. In these fuzzy portfolio models expected returns of securities are accepted as arithmetic means of historical returns. That is, the expected returns are determined with the crisp values. However, this would not lead to the best result when there are uncertainties in finance and/or available data is insufficient. In these conditions, the investor usually formulates his view about the expected return by using fuzzy terms. In this paper the fuzzy expected return is introduced and used in construction of portfolio model.

The fuzzy portfolio selection is performed in several complex situations. These are its difficult uncertain nature, selection of assets by their expected profitability. This is complex optimization problem. The existing fuzzy portfolio selection models are mainly oriented to partial fuzzification of deterministic optimization models. The solutions of these models are suffer from high computational complexity resulted from the conversion of fuzzy models into standard crisp one. In this situation a better solution can be obtained by applying directed random search method such as Genetic Algorithms (GA). The use of GA in portfolio optimization allows us to find global optimal solution with a computational complexity less than by existing method.

Taking into consideration above mentioned, in this paper fuzzy portfolio selection based on fuzzy linear programming solved by GA is considered. The modelling of fuzzy portfolio selection will be performed by using the statistical data taken from Istanbul Stock Exchange.

The paper is organised as follows. In section 2 fuzzy expected return and fuzzy risk are introduced and fuzzy portfolio optimization model is formulated. In section 3 the application of genetic algorithm for portfolio optimization problem is described.

In section 4 using statistical data taken from Istanbul Stock Exchange the results of computer modelling of fuzzy portfolio selection are presented. Finally, conclusions are given in section 5.

## 2 Fuzzy Portfolio Optimization

### 2.1 Fuzzy Risk and Fuzzy Return

Speranza and Mansini used deviation of the portfolio return below the average as the risk and formulated a semiabsolute deviation portfolio selection model [3]. In this model for given return level the investor penalizes the negative semiabsolute deviation that is defined as a risk. The mathematical formulation of semiabsolute deviation portfolio selection model is formulated as

$$z = \frac{1}{T} \sum_{t=1}^T \left| \min\{ 0, \sum_{j=1}^n (r_{jt} - R_j) x_j \} \right| \rightarrow \text{minimize} \quad (1)$$

Subject to

$$\sum_{j=1}^n R_j x_j \geq \rho D, \quad \sum_{j=1}^n x_j = D, \quad x_1, x_2, \dots, x_n \geq 0 \quad (2)$$

Here  $R_j$  is average return in security  $j$  over the entire period,  $x_j$  is portfolio allocation in security  $j$  over the entire period  $T$ .  $\rho$  is minimum rate of return,  $D$  is total budget invested in portfolio.  $r_{it}$  is return of security  $j$  over period  $t$ .

Historical returns and expected returns in the future are input data for semiabsolute deviation portfolio selection model. In this model expected returns of securities are determined by arithmetic means of historical returns. But this is not the best result when there are uncertainties in finance, historical data on a security are not enough. Due to data scarcity, one cannot accurately estimate the statistical parameters. If long period historical returns were stable and returns for the last few months become increasing, company economical condition is improving then expected return will be larger than calculated arithmetic mean. In this case arithmetic mean may be lower limit of expected return. Furthermore, by analyzing financial statements of companies, the upper limit can be assigned based on expert knowledge. In Fig.1 trajectory of historical returns of security (solid line), arithmetic mean of historical return and expert estimation of expected return are shown. By analyzing resent historical data

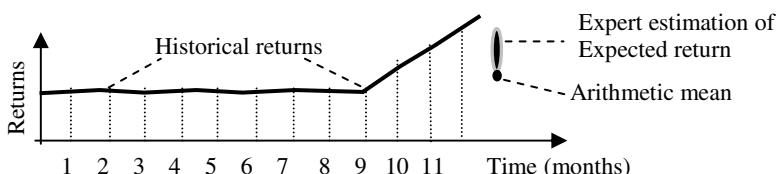


Fig. 1. Expert estimation of expected return

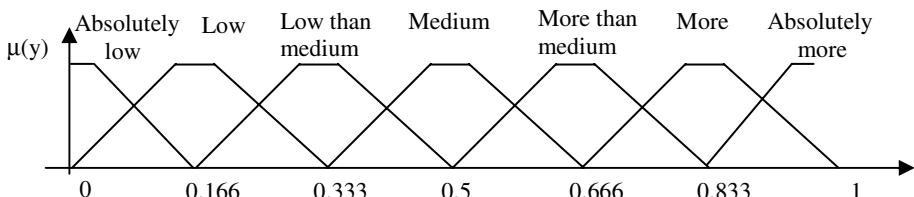
expert estimates the expected return in fuzzy form. Arithmetic mean doesn't characterize the expected return for the next month and it will be lower limit of expected return. In inverse case arithmetic mean will be upper limit of expected return. In the paper to represent uncertain values of expected returns the triangle fuzzy numbers are used.

One of the basic aims of portfolio management is to reduce risk. Risk exists when loss is possible. In the real world, the possibility of loss could not be defined with precision. In the work the mathematical measure of risk is taken as the fuzzy semi-variance model that define fuzzy distance determining the deviation of the portfolio return below the expected return.

$$\text{fuzzy\_risk} = E\{\max[0, \tilde{R} - r]\}$$

Here  $\tilde{R}$  is fuzzy value of expected return.

In practice, decision makers usually use three types of gradations for estimation of risk. These gradations are risk averse, neutral and risk sicker. Using fuzzy logic we have possibility to specify risk more accurately, based on linguistic terms, such as absolutely risky, more risky, medium risky, low risky etc. (Fig. 2). In existing previous works the risk estimation for portfolio construction problem is implemented on the base of crisp measure. In suggested definition the fuzzy measure is used for estimation of portfolio risk.



**Fig. 2.** Linguistic terms assigned for fuzzy risk

## 2.2 Formulation of Fuzzy Portfolio Optimization

Let's consider fuzzy optimization problem for portfolio modelling. Assume that the constraints are in linear order.

$$\underset{x \in \Sigma}{\text{Minimize}} \quad f(\tilde{c}, x) \quad (3)$$

$$\tilde{A}_1 x \geq \tilde{B}_1, \quad \tilde{A}_2 x \leq \tilde{B}_2, \quad x \geq 0 \quad (4)$$

Here  $\tilde{A}_1, \tilde{A}_2$  are matrices and  $\tilde{B}_1, \tilde{B}_2$  are vectors that are defined in  $\mathbf{R}$ .  $x$  is fuzzy decision variables.

In particular case investor may also provide set of bounds for securities, such as

$$L \leq x \leq U, \quad \sum_{i=1}^n x_i = 1$$

Here L and U lower and upper bound defined for the securities. This assumption does not imply any loss of generality of model.

Depend on importance of objective function or constraints we can define tolerance for them, in order to determine optimal solution. Let's consider the case when constraints in (4) are tolerable, that is we may relax the constraints in order to find an optimal compromised solution. The satisfaction of the constraints will be acceptable to some certain membership degree. This degree is called satisfaction degree that allows organizing flexibility of the constraints. In the portfolio modelling the investor opinion is used to choose satisfaction degree on the base of risk-return trade-off analysis. If the objective function attains its value more than or equal investor goal value then there is no need to define satisfaction degree to constraint and investor is fully satisfied. If the objective function attains its value less than investor goal value then there is need to define satisfaction degree for the constraints.

In the paper we use following function for membership degree. The degree of satisfaction of inequality constraint  $\tilde{A}x \geq \tilde{B}_1$  in (4) is defined as follows.

$$\mu_{B_i}(y) = \begin{cases} 1 & \text{if } y \geq b_i \\ 1 + \frac{y - b_i}{q_i} & \text{if } b_i - q_i \leq y < b_i \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

Here  $\mu_{B_i}(y)$  is membership degree of the constraint.  $q_i$  is maximum violation (tolerance) designated for the constraint. The value of  $q_i$  should be provided by investor that makes decision.

Similarly we can define degree of satisfaction of inequality constraint  $\tilde{A}x \leq \tilde{B}_2$

$$\mu_{B_j}(y) = \begin{cases} 1 & \text{if } y \leq b_j \\ 1 - \frac{y - b_j}{q_j} & \text{if } b_j \leq y < b_j + q_j \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

After defining satisfaction degree we can formulate fuzzy optimization problem as

$$\underset{x \in \sum}{\text{Minimize}} \quad f(\tilde{c}, x) \quad (7)$$

$$\begin{aligned} \tilde{A}_1 x &\geq \tilde{B}_1 - q(1 - \mu_B(y)) \\ \tilde{A}_2 x &\leq \tilde{B}_2 + q(1 - \mu_B(y)) \\ x &\geq 0 \end{aligned} \quad (8)$$

Here  $tol = q(1 - \mu_B(y))$  is tolerance designated for the constraints. For  $\leq$  and  $\geq$  constraints the different tolerances and satisfaction degrees can be assigned.

Taking into account above mentioned the fuzzy portfolio optimization model will be formulated as follows.

$$z = \frac{1}{T} \sum_{t=1}^T \left| \min\{ 0, \sum_{j=1}^n (r_{jt} - \tilde{R}_j) x_j \} \right| \rightarrow \text{fuzzy minimize} \quad (9)$$

$$\begin{aligned} \sum_{j=1}^n \tilde{R}_j x_j &\geq \tilde{\rho}D - q(1 - \mu_B(y)), \\ \sum_{j=1}^n x_j &= D, \quad x_1, x_2, \dots, x_n \geq 0 \end{aligned} \quad (10)$$

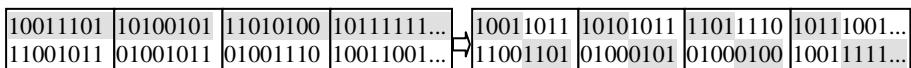
Here  $\tilde{R}_j$  is fuzzy value of expected return.  $x_j$  are fuzzy values of portfolio allocation in security  $j$  over the entire period  $T$  and  $D$  is total budget invested in portfolio.  $r_{jt}$  is return of security  $j$  over period  $t$ . The problem is to determine such values of  $x_j$  under fuzzy inequalities and equality conditions (10), by using them in objective function (9) the fuzzy value of objective function would be minimum.

### 3 Algorithm Description

In this paper GA learning is used to solve fuzzy portfolio selection problem to find optimal solution.

Genes represent the assets and form a chromosome. During portfolio optimization the set of chromosomes are generated randomly. Each chromosome represents assets values. Using GA learning asset values in chromosomes are adjusted by applying various genetic operators, in order to find their optimal values [14]. These genetic operations are selection, crossover and mutation. The aim of the selection is to give more reproductive chances to population members (or solutions) who have higher fitness. Crossover and mutation allow producing new “child” solutions by combining and modifying “parent” solutions. The purpose of crossover and mutation is to give children chances to differ from their parents, and hope that some of the children can be closer to the optimal destination than their parents. There are some forms of crossover: one-point, two-point, multipoint and uniform.

When chromosomes are very long then the use of one or two point crossover operation doesn't give desired results and learning of parameters values takes more computational time. The increasing number of crossover points gives desirable result and allow us to decrease the learning time. In the work multipoint crossover operation is used for learning the values of unknown parameters. In Fig.3 the fragment of multipoint crossover operation is shown. Here genes of parent 1 are described with grey colour, genes of parent 2- with white colour.



**Fig. 3.** Multipoint crossover operation

During mutation a mutation rate associated with the operator. The operator is applied to a bit string which represents a chromosome, it sweeps down the list of bits, re

placing each bit by a randomly selected bit if a probability test is passed. The selection, multipoint crossover and mutation operators are applied for learning the fuzzy values of securities.

## 4 Modelling

Assume that a portfolio manager wants to allocate his wealth among n risky securities and he wants to minimize risk under some given level of portfolio return. Let  $x_j$  are proportion of the total investment devoted to the stock  $j$ ,  $j=1,2,\dots,n$ . Assume that historical data for n stocks are obtained at period T. Obtained data are rate of return of asset  $j$  at period  $t$ , where  $t=1,2,\dots,T$ . To find the amount of investment proportions for each assets the above-described (9) and (10) fuzzy optimization model is applied. The problem is to find such optimal values of invested proportions by using them in equation (9) the value of risk that is given as objective function will be minimized. GA is applied to solve problem and find optimal fuzzy values of proportions. During modelling to describe fuzzy values of the parameters in the objective function and constraints the triangular numbers are used. Each triangular fuzzy number has three parameters- left, middle, and right. Chromosomes represent all left, middle and right sides of fuzzy values. Each chromosome consists of genes that are represented by binary numbers 0 and 1. In figure 3 structure of chromosome that contains twelve assets is shown. Each asset is represented with 36 genes, where 12 genes are used to represent left, 12 genes middle and 12 genes right sides of the fuzzy values of assets. The GA operators are applied to train the values of parameters. During learning 50 populations are used. The crossover rate is taken as 0.8, mutation rate – 0.08.

asset1 $x_1(1), x_m(1), x_r(1)$	asset2 $x_1(2), x_m(2), x_r(2)$	...	asset12 $x_1(12), x_m(12), x_r(12)$
------------------------------------	------------------------------------	-----	--

**Fig. 4.** Structure of chromosome that contain twelve fuzzy assets are shown

During simulation the historical values of return rates for twelve stocks from Istanbul Stock Exchange for twelve month are taken as an input data. The values of returns of stocks for twelve months are given in table 1. Here  $1,2,\dots,12$  month numbers,  $r_{jt}$  are the historical values of returns.

The values of expected returns are determined through arithmetic means of return rates or distribution function. This formula in case of uncertainty, for all stocks does not give accurate values of expected returns. For this reason the expert opinion is used to evaluate more appropriate values of expected returns. Based on analysis of previous historical data and financial reports of companies membership function is defined for each expected return. In the paper the fuzzy values of expected returns are taken in triangular form. The values of expected returns for assets are evaluated as

$$\begin{aligned} R1 &= [0.0789 \ 0.032 \ 0.0454 \ 0.0585 \ 0.0478 \ 0.064 \ 0.1948 \ 0.0771 \ 0.0549 \ 0.1155 \ 0.1096 \ 0.0622]; \\ R2 &= [0.0869 \ 0.037 \ 0.0504 \ 0.0635 \ 0.0528 \ 0.069 \ 0.2048 \ 0.0821 \ 0.0589 \ 0.1205 \ 0.1146 \ 0.0722]; \\ R3 &= [0.0949 \ 0.042 \ 0.0554 \ 0.0685 \ 0.0578 \ 0.074 \ 0.2148 \ 0.0871 \ 0.0639 \ 0.1255 \ 0.1196 \ 0.0822]; \end{aligned}$$

Here R1, R2, R3 are left, middle and right sides of fuzzy values of expected return, correspondingly.

**Table 1.** Return of securities over 12 month

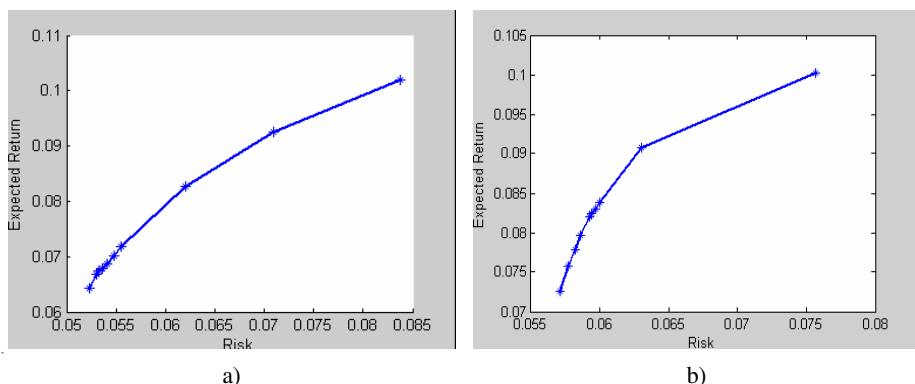
	Times (months)											
	1	2	3	4	5	6	7	8	9	10	11	12
r <sub>1t</sub>	0.1613	-0.016	0.2990	-0.0202	-0.0481	0.0833	-0.0303	-0.1281	0.8095	-0.2588	0.1184	0.1692
r <sub>2t</sub>	0.0787	0.0595	0.1667	-0.1724	0.0482	0.3387	-0.0159	-0.0597	0.2523	0.4658	-0.4786	-0.2391
r <sub>3t</sub>	0.2405	0.0	0.2553	0.1325	-0.117	-0.2656	0.3333	-0.0204	0.4627	-0.0290	-0.3000	-0.0877
r <sub>4t</sub>	0.2603	0.1587	0.1667	-0.1000	0.0	-0.0769	0.0947	-0.0686	0.4571	-0.0139	-0.2577	0.1412
r <sub>5t</sub>	0.2857	0.1667	0.2923	-0.3229	-0.040	0.0204	0.1136	-0.0784	0.3247	0.1324	-0.2609	0.0
r <sub>6t</sub>	0.1091	0.0377	0.3119	-0.1920	0.1111	0.0227	0.0233	-0.1224	0.6000	0.0533	-0.1758	0.1098
r <sub>7t</sub>	0.1463	0.9680	0.0714	-0.4355	0.1481	-0.1290	0.6316	1.6389	0.0	0.2414	-0.5167	-0.1864
r <sub>8t</sub>	-0.015	0.0984	0.3708	-0.2054	0.3176	-0.2917	0.4286	0.0769	0.2188	-0.0303	-0.0294	0.0462
r <sub>9t</sub>	0.0750	0.0811	0.3704	-0.2603	-0.0135	0.0571	0.0294	-0.1707	0.7083	0.0435	-0.0980	-0.0556
r <sub>10t</sub>	0.4590	-0.047	0.5422	-0.3712	-0.0959	-0.1310	0.0633	0.4630	0.4400	0.3636	-0.4149	0.1750
r <sub>11t</sub>	0.5152	0.4143	0.5054	-0.2377	-0.0758	-0.0294	0.0099	-0.114	0.4805	0.0132	-0.1648	0.0581
r <sub>12t</sub>	0.1169	0.6383	0.9583	-0.2208	-0.0833	-0.2186	0.0238	0.0095	0.1798	-0.0220	-0.3259	-0.0690

Using input data the objective function and constraints are formed. The accessible fuzzy value of portfolio return in the constraint is taken as [0.025 0.03 0.035]. By applying GA learning algorithm the optimal values of invested proportions have been determined. In table 2, after defuzification, the obtained values of assets are given. For these values of assets the objective risk function and portfolio return were determined as 0.0548 and 0.0701, correspondingly.

**Table 2.** Simulation result

x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12
0.2001	0.2303	0.0106	0.0381	0.0663	0.0891	0.0611	0.1835	0.0494	0.0162	0.0322	0.0183

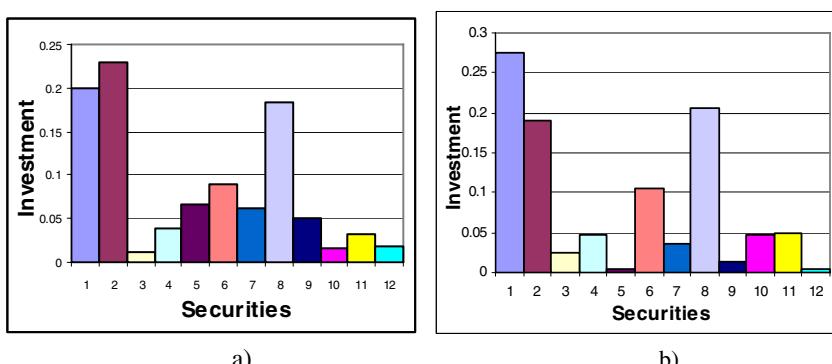
Using portfolio model (11) and (12) and statistical data efficient frontier for portfolio model is constructed. The efficient frontier is obtained for different values of portfolio return. During simulation fifty chromosomes are generated for each asset. In figure 5(a) the portfolio efficient frontier for twelve stocks is given. The efficient frontier describes the defuzification results of fuzzy expected returns and risks.

**Fig. 5.** Efficient frontiers for: a) Fuzzy portfolio, b) Deterministic portfolio

By assigning satisfaction degree investor can decrease the value of risk associated to portfolio to some level. Let us consider determining the risk value for different values of satisfaction degree 1, 0.55 and 0.2. We accept maximum tolerance as  $q=0.02$ . When satisfaction degree was equal 1 the values of expected return and risk were 0.0701 and 0.0548. For 0.55 and 0.2 values of satisfaction degree the model is again run and new values of invested proportions have been determined. For the case-0.55 the value of risk is determined as 0.0520 and the value of return- 0.0670. When satisfaction degree is taken as 0.2 the value of risk is determined as 0.0505 and the value of return- 0.064. The use of satisfaction degree allows the decreasing of the value of objective risk function to some certain values.

Using the same historical data the modelling of deterministic portfolio selection has been carried out. For each stock the expected returns are determined through arithmetic means of historical return rates. Then applying deterministic semiabsolute deviation model and genetic algorithm the optimal values of invested proportions have been determined. In figure 5(b) the efficient frontier which is constructed by deterministic model is given. As shown from the figures the values of objective risk function in fuzzy portfolio are less than in deterministic one. It can be said that deterministic model is more risky than fuzzy model.

In figure 6 the histograms for fuzzy (a) and deterministic (b) portfolios are given. For obtained values of invested proportions the value of risk in fuzzy portfolio was equal to 0.0548 and in deterministic portfolio it was equal to 0.0566. Obtained risk value in fuzzy model is less than in deterministic one. In deterministic portfolio model the portions of investment for fifth and twelfth assets are very low and investor invests his money basically to the first, second and eighth assets. As known from portfolio theory investing to wide variety of assets allows creating diversified portfolio that reduces the portfolio risk [1]. In fuzzy portfolio the invested portions for first and eight assets are less than in deterministic one and this is distributed between other assets. The use of fuzzy portfolio allows making more distributive investment among securities (figure 6). Analysis of obtained simulation results demonstrates that fuzzy portfolio selection model is more reasonable than deterministic one in practice.



**Fig. 6.** Histograms of the portions: a) Fuzzy portfolio, b) Deterministic portfolio

## 5 Conclusion

In this paper using genetic algorithm the construction of fuzzy portfolio model is considered. Because of insufficiency of information the values of expected returns are estimated by investor perception. Based on historical returns of assets and fuzzy values of expected returns the formulation of fuzzy portfolio optimization model is given. The applied model has soft constraint allowing to estimate the investor's preference about risk-return trade-off. The fuzzy GA is applied to find the values of securities. According to investor opinion the satisfaction degree is defined for the constraint. By assigning satisfaction degree the value of risk function is decreased to certain value. The model is tested using statistical data taken from Istanbul Stock Exchange. The obtained results from the modelling satisfy the efficiency of the presented fuzzy approach in portfolio selection.

## References

- [1] Harry Markowitz. "Portfolio Selection". *The Journal of Finance*, vol.VII,N:1(1952)
- [2] Konno, H. and H. Yamazaki (1991): "Mean-Absolute Deviation Portfolio Optimization Model and Its Applications to Tokyo Stock Market", *Management Science*, 37 (1991).
- [3] R. Mansini and M.G. Speranza, Heuristic algorithms for the portfolio selection problem with minimum transaction lots, *European Journal of Operational Research* 114. (1999).
- [4] Zadeh L.A. Fuzzy Sets, *Information and Control*, vol.8 (1965) 338-353.
- [5] H. J. Zimmermann. Description and optimization of fuzzy systems. *International Journal of General Systems*, 2: (1976) 209–215.
- [6] H.J. Zimmermann. *Fuzzy Set Theory*. Kluwer Academic Publishers, Boston, 1996
- [7] J. Watada, Fuzzy portfolio selection and its application to decision making. *Tatra Mountains Mathematical publication* 13 (1997) 219-248.
- [8] H.Tanaka and P.Guo. "Portfolio selection based on upper and lower exponential possibility distributions". *European Journal on Operational Research*, vol.114, (1999) 115-126.
- [9] Hideo Tanaka, Peijun Guo, I.Burhan Turksen. Portfolio selection based on fuzzy probabilities and possibility distributions. *Fuzzy Sets and Systems*,(2000)387-397
- [10] T.Leon, V.Liern, E.Vercher. Viability of infeasible portfolio selection problems: A fuzzy approach. *Eurepean Journal of Operation Research*, (2002) 178-189.
- [11] Rahib Abiyev, Mustafa Menekay. Linear programming models for portfolio selection. Second International conference on Softcomputing and Computing with Words in System Analysis,Decision and Control. ICSCCW 2003.Dedicated to Prof. Zadeh. Antalya, Turkey, September 9-11 (2003) 226-231
- [12] K.K.Lai, S.Y.Wang, J.P.Xu, S.S.Zhu and Y.Fang, "A class of Linear Interval Programming Problems and Its Application to Portfolio Selection". *IEEE Transactions on Fuzzy Systems*, Vol.10, No.6,pp.698-703, December, (2002).
- [13] Tu Van Le. Fuzzy evolutionary programming for portfolio selection in investment. Intelligent Processing and Manufacturing of Materials, 1999. IPMM '99. Proceedings of the Second International Conference on Volume 1, 10-15 July 1999, pp:675-679 vol. 1. Digital Object Identifier 10.1109/IPMM.1999.792575
- [14] Goldberg, David E. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison –Wesley Publishing Company, Inc.(1989)
- [15] J. Ramik, Soft Computing: Overview and Recent Developments in Fuzzy Optimization. Listopad (2001).

# Fuzzy Genetic Algorithms for Pairs Mining\*

Longbing Cao, Dan Luo, and Chengqi Zhang

Faculty of Information Technology, University of Technology Sydney, Australia  
{lbciao, dluo, chengqi}@it.uts.edu.au

**Abstract.** Pairs mining targets to mine pairs relationship between entities such as between stocks and markets in financial data mining. It has emerged as a kind of promising data mining applications. Due to practical complexities in the real-world pairs mining such as mining high dimensional data and considering user preference, it is challenging to mine pairs of interest to traders in business situations. This paper presents fuzzy genetic algorithms to deal with these issues. We introduce a fuzzy genetic algorithm framework to mine pairs relationship, and propose strategies for the fuzzy aggregation and ranking of identified pairs to generate final optimum pairs for decision making. The proposed approaches are illustrated through mining stock pairs and stock-trading rule pairs in stock market. The performance shows that the proposed approach is promising for mining pairs helpful for real trading decision making.

## 1 Introduction

*Pairs mining* aims to mine pair relationships between a couple of instances. A *pair relationship* represents that two instances are correlated or associated in terms of some form of statistic or probabilistic measures. Pairs identified, either intra class or inter class, may indicate dynamics of interest to users. For instance, in stock data mining [1,6,7], pairs mining can discover stock pairs linked by correlation relationship from a series of stocks in the market. Pairs mining may also find pair relationships between stocks and derivatives (namely stock-derivative pairs) lodged in an exchange or of the same listed companies. These pairs identified are helpful for traders to make smart decisions. For example, pairs trading strategy can be designed to trade a basket of stocks to distribute potential trading and investment risk rather than putting all money on one instrument.

Real-world pairs mining is quite complicated because the pairs are hidden in high dimensional data and must satisfy uncertain user requests. For instance, stock pairs may hide in all combinations of stocks validly listed in an exchange, while the number of listed stocks can be over 1,000. This may lead to big computational cost. More challenging issue is to mine pairs showing the correlation between stocks and trading rules in a market. On the other hand, user preference and business needs are the drivers to enhance pairs of interest to real user needs [3]. Therefore, it is important to tackle the above factors in identifying and evaluating pairs.

---

\* This work is sponsored by Australian Research Council Discovery Grant (DP0667060, DP0449535), China Overseas Outstanding Talent Research Program of Chinese Academy of Sciences (06S3011S01), and UTS ECRG and Chancellor grants.

Correlation [5] and association mining [5] can be used to mine pair relationships. However, the problem for them to mine pairs from the above real-world situations is the ineffectivity in mining pairs from high dimensional data and considering business preference into the mining process. To efficiently address high dimensional data, genetic algorithms [4] are widely used. However, genetic algorithms are not good at dealing with domain-oriented business requests and user preference.

This paper investigates pairs mining in high dimensional data and by involving user preference. The main contribution of the paper is that it proposes fuzzy genetic algorithms by integrating fuzzy set [8] and genetic algorithms, and fuzzy aggregation and ranking mechanisms for generating optimum pair set for final decision support. Correlation analysis techniques are used as a pair selection method and merged into fuzzy genetic algorithms.

The remaining sections are organized as follows. Section 2 introduces pairs mining. Section 3 presents a framework for fuzzy genetic algorithms and fuzzy aggregation and ranking of mined pairs. We illustrate fuzzy genetic algorithm-based rule-stock pairs mining in stock market in Section 4. Performance evaluation is presented in Section 5. Section 6 concludes this work.

## 2 Pairs Mining

Pairs mining targets mining *pair relationships* existing in a pair of instances of an attribute, or between a pair or a number of attributes. Pair relationship indicates that there are two instances that are highly correlated or associated in terms of some form of probabilistic or statistic metrics. For instance, in stock data mining, pair relationships between stocks can be analyzed according to the coefficient of correlation of stock prices. We call these pairs of stocks as *stock pairs*. We can also mine pair relationships between stocks and markets, namely stock-market pairs, based on the correlation between stock price and market index.

Pairs coming from the same class are called *kindred* or *homogeneous* pairs, for instance, stock pairs and market pairs in stock data mining. On the other hand, those come from different families are *alien* or *heterogeneous* pairs such as stock-market pairs and stock-trading rule pairs. Both kindred and alien pairs are noteworthy, they are exchangeable in terms of some conditions. For instance, normally the pair relationship among stocks is kindred. While they may present as alien pairs if the two parties come from different sectors and we want to highlight the significance of varying sectors in assessing the pair relationship.

Furthermore, pairs may be linked together in either positive or negative pair relationships. Positive pair relationships mean that the two parties are linked through certain obverse relation such that both of them follow similar patterns. In correlation-based pairs mining, pairs represented by positive correlation coefficients follow obverse change trends. For instance, in stock pair mining, some stocks are found to be positively linked with some others when both prices go up following similar patterns.

Negative pair relationships indicate that two parties are coupled in opposite trends or patterns. In correlation-based pairs mining, if the correlation coefficient is negative, then one goes up while another goes towards the reverse direction. For instance, in stock pairs identified, the trend of price dynamics of some stocks are found to be

opposite to its paired partners. Another interesting type of negative pairs exist in terms of negative association rules. In this case, one item presented indicates that another definitely will not come up, which is against commonsense.

Correlation mining and association mining can be used for the above pairs mining. However, they are not good at handling high dimensional data and user preference. To tackle these issues, we present an effective framework of fuzzy genetic algorithms.

### 3 Fuzzy Genetic Algorithms

This section introduces a framework of fuzzy genetic algorithms. We also introduce the mechanisms for aggregating and ranking pairs to generate final workable pairs.

#### 3.1 A Fuzzy Genetic Algorithm Framework

A fuzzy genetic algorithm [2] is a fuzzy set-coded genetic algorithm where each individual (chromosome) is composed of a set of membership functions. The principle of fuzzy optimization is as follows. Suppose we have a membership function  $F$  so that  $\bar{y} = F(\bar{x})$ , where  $\bar{x}$  is fuzzy set-based input, fuzzy set  $\bar{y}$  is the output from  $F$  given  $\bar{x}$ . Fuzzy optimization is to find a proper  $\bar{x}$  in its valid value range to “maximize” the fuzzy set  $\bar{y}$ . This is achieved through maximizing a suitable mapped measure of  $\bar{y}$ , for instance the center of gravity centroid( $\bar{y}$ ). Here the measure can map the fuzzy sets into real numbers.

In designing fuzzy genetic algorithms, issues in conventional genetic algorithms are put into fuzzy context and converted into fuzzy versions, for instance, fuzzy representation, fuzzy genetic operators, etc. In particular, fuzzy genetic algorithms need to consider the validation and ranking of the created fuzzy sets. In mining pairs in financial markets, we develop the following fuzzy genetic algorithm framework.

**ALGORITHM 1.** pseudo code for fuzzy genetic algorithm

Input: real number set  $X$

Output: optimal fuzzy set  $Y$  for decision support

Procedure: FGA( $\mu$ ,  $X(t)$ ,  $\bar{X}(t)$ ,  $\bar{X}'(t)$ ,  $Y$ )

//start with an initial time

$t := 0$ ;

//initialize a fuzzy random population of individuals  $\bar{X}(t)$  by fuzzifying the real number sets  $X(t)$  with proper membership functions  $\mu_{\bar{x}}$ ,

initialize  $\bar{X}(t) = \{(x, \mu_{\bar{x}}(x)) \mid x \in X(t), \mu_{\bar{x}} : X(t) \rightarrow [0,1]\}$ ;

//evaluate the fitness of all initial individuals of population based on fuzzy evaluation  
evaluate  $\bar{X}(t)$ ;

//test for termination criterion

While (not done) do

//increase the time counter

$t := t + 1$ ;

//select a fuzzy sub-population set  $\bar{X}'(t)$  for offspring production

$\bar{X}'(t) := \text{select } \bar{X}(t)$ ;

//crossover the “genes” of the selected parents  $\bar{X}'(t)$

```

crossover  $\bar{X}'(t)$ ;
//perturb the mated population stochastically
mutate  $\bar{X}'(t)$ ;
//fuzzily evaluate its new fitness
evaluate  $\bar{X}'(t)$ ;
//select the survivors  $\bar{Y}$  from actual fitness
 $\bar{Y} := \text{survive } \bar{X}(t), \bar{X}'(t)$ ;
End
//fuzzily rank the survivors
rank  $\bar{Y}$ ;
//defuzzify and export the final survivors
export  $Y$ ;
-----
```

The above fuzzy genetic algorithm framework deals with all basic issues such as initialization, selection, crossover, mutation and evaluation of stock pairs mining in fuzzy context. For initialization, all individuals are sampled randomly within the valid domain. In stock pair mining, we identify the business interestingness measure *sharpe ratio* as the fitness function for all individuals in the pair population. Real coded sharpe ratio measures the performance of a pair in terms of both return and risk. If sharpe ratio is high, the rule leads to high return with low risk. The following defines real number sharpe ratio  $SR$  used in stock market.

$SR = (R_p - R_f) / \sigma_p$  Where  $R_p$  is expected portfolio return,  $R_f$  is risk free rate,  $\sigma_p$  is portfolio standard deviation.

We fuzzify the real coded  $SR$  into the interval  $[0,1]$  to get its fuzzy sets  $\bar{SR}$ . We use triangle piecewise linear membership function to fuzzify the universal sets. For instance, we specify ten levels of linguistic values, namely 1<sup>st</sup>, 2<sup>nd</sup>, ..., 10<sup>th</sup> from the lowest to the highest, for the fuzzy linguistic variable *sharpe ratio*  $\bar{SR}$ . Hereby we generate top  $\bar{N}$  target objects, for instance the corresponding trading rules,  $\bar{N}$  refers to those rules which correspond to the first  $N$  highest linguistic values.

In our case, the genetic algorithms for pairs mining are real coded, therefore its crossover can be in an arithmetic and/or multiple-point manner. We provide multi-point arbitrary crossover in a shuffling probability  $p (0 \leq p \leq 1)$  of alleles on top of the top  $\bar{N}$  selected sub-populations. Suppose  $\gamma$  is a random number in  $[0,1]$ , the following illustrates the shuffling situation if  $\gamma \leq p$ .

$$\begin{aligned}\bar{X}_1' &= (x_{10}, \dots, x_{2i}, x_{1(i+1)}, \dots, x_{2k}, \dots) \\ \bar{X}_2' &= (x_{20}, \dots, x_{1i}, x_{2(i+1)}, \dots, x_{1k}, \dots)\end{aligned}$$

On the other hand, the mutation is based on changing the original value stochastically by the mutation rate  $q (0 \leq q \leq 1)$  either positively or negatively. Our strategy for the mutation is to conduct the mutation operation on top of the shuffled sets  $\bar{X}'(t)$  with the rate  $q$  around 0.03.

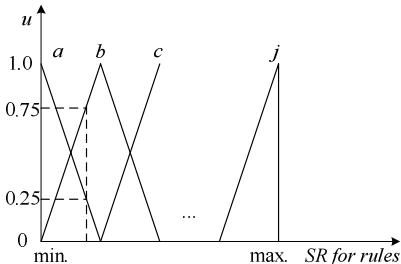
Based on the above operations, a collection of optimized individuals emerge from the candidate population. They are possible optimal candidates for the final recommendation. In order to generate the final optimal list, special attention should be paid to the aggregation, evaluation and ranking of fuzzy functions and fuzzy sets.

### 3.2 Fuzzy Evaluation, Aggregation and Ranking

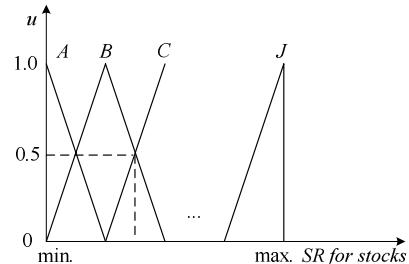
Fuzzy evaluation refers to evaluate the validity of fitness functions, fuzzy sets and individuals. The validity is directly induced and embodied through membership functions. Therefore, it is essential to check the validity of membership functions and fitness functions. Taking the fuzzification of sharpe ratio as an instance, its real values can be negative or positive in the real number field. However, in fuzzy genetic algorithms, the universe of the discourse of the fuzzified sharpe ratio  $\bar{SR}$  must be in  $[0,1]$ . Otherwise, they are abnormal and invalid, and we need to normalize membership grades into the interval of  $[0,1]$ . We also check the monotone of  $\bar{SR}$  membership grades to guarantee that only one grade exists for each  $\bar{SR}$  element, which is mapped to single real value of sharpe ratio. If it is not monotonic, corresponding strategy should be taken to monotonize it. Based on these policies, we can check the validity of individuals and the fuzzy sets.

The output of fuzzy genetic algorithms is to recommend a set of optimal individuals  $Y$ . To this end, fuzzy aggregation and fuzzy ranking play an important role in generating the final survivors. In the following paragraphs, we illustrate the ranking strategy for discovering trading rule-stock pairs. There are three steps for us to find out the actionable trading rules highly correlated to given stocks. One of the steps is to mine and rank the in-depth trading rules [7] for a specific stock. Another step is to detect and order the very appropriate stocks for a given trading rule. Then we aggregate these two lists through fuzzy aggregation rules to obtain a set of composite optimal stock-trading rule pairs. Finally, we fuzzily rank the trading rule-stock pairs, and further defuzzify them to generate final outputs. See Section 4 for more details about discovering rule-stock pairs.

Let  $SR$  be fitness function for the above first two steps. Suppose we build ten ascending linguistic values from  $1^{st}$  to  $10^{th}$ . To distinguish the two cases, as illustrated in Figure 1 and 2, we use fuzzy linguistic terms  $a$  to  $j$  and fuzzy values  $A$  to  $J$  to label the fuzzy sets for the optimal trading rules given a specific stock (we called rule sets), and for the appropriate stocks given a trading rule (called stock sets), respectively.



**Fig. 1.** Fuzzy set for trading rules given stock

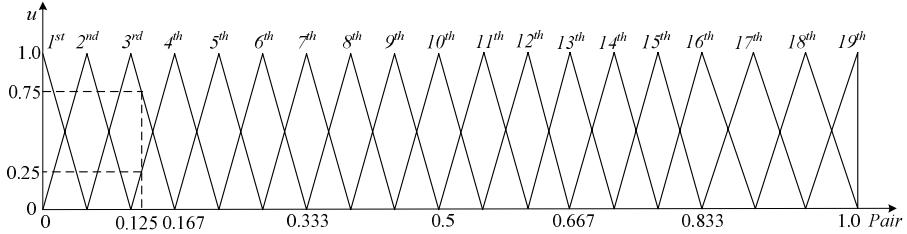


**Fig. 2.** Fuzzy set for stocks given trading rule

In practice, even though sharpe ratio is used as the fitness and similar linguistic measures are used for both rule set and stock set situations, the meaning of a specific corresponding linguistic term, say  $b$  and  $B$  in this case, may be highly varying. This

means that we cannot aggregate the stock-rule pairs based on the equal matching of two linguistic values from different sets. Instead, we develop the following solution to aggregate the two fuzzy groups.

To aggregate the two groups, we set up another fuzzy variable called *rule-stock pair* (in short *pair*) to correlate close partners between the rule set and the stock set. The *pair* has 19 linguistic values ascending from 1<sup>st</sup>, 2<sup>nd</sup> to 19<sup>th</sup>. Figure 3 defines its triangle fuzzy sets. Further, the following fuzzy aggregation rules are defined to merge the fuzzy sets from different groups.



**Fig. 3.** Fuzzy set for trading rule-stock pairs

**DEFINITION 1.** (*Fuzzy aggregation rule*) if the fuzzy rule set is  $m$ -th, and the fuzzy stock set is  $n$ -th, then the rule-stock pair is  $(m+n-1)$ -th.

For instance, if the rule set is  $c$  (i.e., 3<sup>rd</sup>), and the stock rule is  $d$  (4<sup>th</sup>), then the rule-stock pair is ranked as 6<sup>th</sup> (3+4-1). Table 1 illustrates the fuzzy aggregation and ranking of all linguistic values from the rule set and the stock set, respectively. The fuzzy rule makes it possible to integrate the rule set and the stock set, and output the higher ranked rule-stock pairs as final survivors for optimal decision-making.

**Table 1.** Fuzzy aggregation and ranking of rule sets and stock sets

	$a$	$b$	$c$	$d$	$e$	$f$	$g$	$h$	$i$	$j$
$A$	1 <sup>st</sup>	2 <sup>nd</sup>	3 <sup>rd</sup>	4 <sup>th</sup>	5 <sup>th</sup>	6 <sup>th</sup>	7 <sup>th</sup>	8 <sup>th</sup>	9 <sup>th</sup>	10 <sup>th</sup>
$B$	2 <sup>nd</sup>	3 <sup>rd</sup>	4 <sup>th</sup>	5 <sup>th</sup>	6 <sup>th</sup>	7 <sup>th</sup>	8 <sup>th</sup>	9 <sup>th</sup>	10 <sup>th</sup>	11 <sup>th</sup>
$C$	3 <sup>rd</sup>	4 <sup>th</sup>	5 <sup>th</sup>	6 <sup>th</sup>	7 <sup>th</sup>	8 <sup>th</sup>	9 <sup>th</sup>	10 <sup>th</sup>	11 <sup>th</sup>	12 <sup>th</sup>
$D$	4 <sup>th</sup>	5 <sup>th</sup>	6 <sup>th</sup>	7 <sup>th</sup>	8 <sup>th</sup>	9 <sup>th</sup>	10 <sup>th</sup>	11 <sup>th</sup>	12 <sup>th</sup>	13 <sup>th</sup>
$E$	5 <sup>th</sup>	6 <sup>th</sup>	7 <sup>th</sup>	8 <sup>th</sup>	9 <sup>th</sup>	10 <sup>th</sup>	11 <sup>th</sup>	12 <sup>th</sup>	13 <sup>th</sup>	14 <sup>th</sup>
$F$	6 <sup>th</sup>	7 <sup>th</sup>	8 <sup>th</sup>	9 <sup>th</sup>	10 <sup>th</sup>	11 <sup>th</sup>	12 <sup>th</sup>	13 <sup>th</sup>	14 <sup>th</sup>	15 <sup>th</sup>
$G$	7 <sup>th</sup>	8 <sup>th</sup>	9 <sup>th</sup>	10 <sup>th</sup>	11 <sup>th</sup>	12 <sup>th</sup>	13 <sup>th</sup>	14 <sup>th</sup>	15 <sup>th</sup>	16 <sup>th</sup>
$H$	8 <sup>th</sup>	9 <sup>th</sup>	10 <sup>th</sup>	11 <sup>th</sup>	12 <sup>th</sup>	13 <sup>th</sup>	14 <sup>th</sup>	15 <sup>th</sup>	16 <sup>th</sup>	17 <sup>th</sup>
$I$	9 <sup>th</sup>	10 <sup>th</sup>	11 <sup>th</sup>	12 <sup>th</sup>	13 <sup>th</sup>	14 <sup>th</sup>	15 <sup>th</sup>	16 <sup>th</sup>	17 <sup>th</sup>	18 <sup>th</sup>
$J$	10 <sup>th</sup>	11 <sup>th</sup>	12 <sup>th</sup>	13 <sup>th</sup>	14 <sup>th</sup>	15 <sup>th</sup>	16 <sup>th</sup>	17 <sup>th</sup>	18 <sup>th</sup>	19 <sup>th</sup>

However, the above aggregation and ranking strategy is based on the fuzzification of fitness and membership functions. Therefore, a rule in fuzzy set  $c$  or a stock in fuzzy set  $D$  is basically from fuzzy rather than crisp perspective. For instance, as shown in Figure 1 and 2, a trading rule could be classified into fuzzy set  $b$  with a

membership grade  $\mu=0.75$  or set  $a$  with the grade  $\mu=0.25$ . Similar thing exists for stock set, a stock could be segmented into fuzzy set  $B$  or  $C$  with the same grade  $\mu=0.5$ . In this case, the outcome of the fuzzy aggregation and ranking could have four options, namely  $2^{nd}$ ,  $3^{rd}$ ,  $4^{th}$  or  $5^{th}$ .

To manage the above uncertain situations in the fuzzy aggregation and ranking, a ranking coefficient  $\rho$  based on moment defuzzification is introduced. It defuzzifies a fuzzy set returning a floating point that represents the fuzzy set. It actually measures how optimal a pair is.

$$\rho = \sum_{l=1}^m \eta_l \mu_l^R \mu_l^S / \sum_{l=1}^m \mu_l^R \mu_l^S$$

Where,  $m$  refers to the number of triggered linguistic values,  $l=1, 2, \dots, m$  corresponds to each triggered linguistic value.  $\mu_l^R$  is the membership grade of No.  $l$  linguistic term relevant to the sharpe ratio of a rule.  $\mu_l^S$  is the membership grade of No.  $l$  linguistic term corresponding to the sharpe ratio of a stock.  $\eta_l$  is the centroid of the No.  $l$  triggered linguistic value, it is calculated in terms of the moment and the area of each subdivision.

The ranking coefficient  $\rho$  provides a solution to deal with possible uncertainty when a rule-stock pair is aggregated. A real number can be obtained to measure a fuzzy rule-stock pair in a relatively crisp manner. For instance, we can calculate and get  $\rho=0.125$  in the above example. As shown in Figure 3, this clearly indicates that this rule-stock pair is ranked as  $3^{rd}$  fuzzy set since its membership grade is 0.75 much larger than fuzzy set  $4^{th}$  with grade 0.25.

## 4 Mining Rule-Stock Pairs

We instantiate the above fuzzy genetic algorithm framework to mine financial pairs such as stock pairs and rule-stock pairs [7]. Due to space limitation, here we only introduce the rule-stock pairs mining through analyzing the correlation between trading rules and tradable stocks. In market trading, some trading rules are tested more profitable to trade a class of stocks, while others are more suitable for other stocks. Using pairs mining we may evidence whether there exists pair relationship between trading rules and stocks or not. The above fuzzy genetic algorithm framework is used to develop algorithms for discovering the target rule-stock pairs.

In identifying rule-stock pairs which can be used for trading support, traders are invited to give suggestions on designing features, interestingness measures and parameter optimization strategy. They also helped us construct interestingness metrics and evaluate and refine rule-stock pairs. Taking the Australian Stock eXchange (ASX) as an instance, six types of trading rules such as Channel Breakout and 27 ASX stocks such as ANZ are chosen for the experiments in the orderbook data. Five different investment plans are used to trade the above identified rule-stock pairs by enhancing trading rules via considering and trading the correlated stocks. The following introduces the method for rule-stock pairs mining algorithms.

---

**ALGORITHM 2.** Method for improving trading rules by analyzing correlation between rules and stocks

Input: a set of historical intraday orderbook transactions  $T$ , a set of trading rules  $R$ , a set of stocks  $S$ , a coefficient threshold  $coeff_0$ , a sharpe ratio threshold  $sr_0$ , a return threshold  $r_0$ ,

Output: Fuzzily ranked trading rule-stock pairs

Method:

1. Given a stock  $S_i$ , and a type of trading rule  $R_j$ , mining actionable rules  $r_{ijm}$  ( $m = 0, 1, \dots$ ) for the stock in  $T$  using fuzzy genetic algorithms described in ALGORITHM 1 and 4;
  2. Mining all actionable rules  $r_{ijm}$  ( $i = 0, 1, \dots, j = 0, 1, \dots; m = 0, 1, \dots$ ) for all stocks  $S_i$  ( $i = 0, 1, \dots$ ) and all types of trading rules  $R_j$  ( $j = 0, 1, \dots$ );
  3. Fuzzily aggregating the rule set  $r_{ijm}$  to generate a fuzzily optimal rule for a given stock;
  4. Generating fuzzy optimal rule-stock pairs  $s_i - r_j$  ( $i = 0, 1, \dots, j = 0, 1, \dots$ );
  5. Evaluating the rule-stock pairs  $s_i - r_j$  by involving traders' concerns;
  6. Fuzzily ranking the rule-stock pairs  $s_i - r_j$ ;
  7. Exporting classified rule-stock pairs in terms of user acceptable linguistics.
- 

Figure 4 illustrates an excerpt of fuzzily ranked rule-stock pairs discovered in ASX intraday orderbook data in June 2001. Three types of rules (coded as 1 to 3) and 27 ASX stocks (coded from 1 to 27) are selected for this experiment. The total 81 rule-stock pairs are classified into five fuzzy groups in terms of sharpe ratio: VP-very positive, P-positive, W-watchable, N-negative and VN-very negative. We can see that Rule 2-Stock 24, Rule 2-Stock 26, and Rule 1-Stock 24 are three *very positive* pairs in June 2001. These high-ranking pairs may be useful to support trading.

Rule Code	Stock Code	Best Return	Sharpe Ratio	Fuzzy Category
2	24	0.25	0.71	VP
2	26	0.17	0.66	VP
2	24	0.25	0.61	VP
1	14	0.07	0.56	P
2	24	0.20	0.52	P
1	26	0.08	0.49	P
1	18	0.09	0.49	P
3	21	0.12	0.48	P
2	21	0.13	0.48	P
3	14	0.09	0.48	P
1	19	0.07	0.46	P
1	8	0.07	0.44	P
1	16	0.07	0.43	P
3	14	0.06	0.43	P
2	18	0.06	0.42	P
2	19	0.06	0.42	P
3	18	0.16	0.37	W
3	9	0.05	0.35	W
3	26	0.08	0.34	W
2	22	0.07	0.33	W

**Fig. 4.** Fuzzily ranked trading rule-stock pairs

## 5 Performance Evaluation

In real-world mining, performance evaluation not only demonstrates the advantage of a specific data mining algorithm but also justifies whether the developed approach can satisfy real user needs or not, besides. Therefore, the algorithm should satisfy both technical interestingness and business interestingness [3].

Technically, we focus on predictability and actionability which can be instantiated into different metrics in terms of particular domain problem. Let  $D$  be the number of total pairs (e.g., rule-stock pairs) found in in-sample and out-of-sample sets satisfying certain business interestingness (say return must be larger than a threshold  $R_0$ ).  $A$  ( $B$ ) be the number of total pairs in in-sample (out-of-sample) data set.  $AB$  be the number of pairs existing in both in-sample and out-of-sample sets, which satisfy business interestingness request. Then, the following statistics measures the actionability of trading rule-stock pairs in terms of in-sample and out-of-sample comparison.

**DEFINITION 2.** (*Probability of rule-stock pairs*) *The following probability functions are defined for rule-stock pairs in or across in-sample and out-of-sample data sets:*  $P(A)=A/D$ ,  $P(B)=B/D$ ,  $P(AB)=AB/D$ ,  $P(A|B)=P(AB)/P(B)$ ,  $P(B|A)=P(AB)/P(A)$ .

**Example 1.** In the rule-stock pair mining, 10 pairs are ranked as class 19<sup>th</sup> in in-sample data, which are the most promising pairs. While 7 of them also satisfy the same condition in out-of-sample set, another 2 new pairs jump into the class 19<sup>th</sup> in testing. Then  $P(A)=83\%$ ,  $P(B)=75\%$ ,  $P(AB)=58\%$ ,  $P(A|B)=78\%$ ,  $P(B|A)=70\%$ .

**DEFINITION 3.** (*Actionability metrics of rule-stock pairs*) *The following metrics: Confidence, All\_Confidence, Cosine and Coherence are defined for measuring the actionability of the trained rule-stock pairs in in-sample set when they are tested in out-of-sample data.*

$$\text{Confidence} = \max(P(A|B), P(B|A))$$

$$\text{All\_Confidence} = P(AB) / \max(P(A), P(B))$$

$$\text{Cosine} = P(AB) / \sqrt{P(A)P(B)}$$

$$\text{Coherence} = P(AB) / (P(A) + P(B) - P(AB))$$

In general, the value range of all the above metrics is in the interval of [0, 1]. Larger value donates bigger actionability of pairs when they are deployed into the real trading. We count the summarized statistics of actionability for the discovered rule-stock pairs from April to October 2001 using ASX intraday orderbook data using sliding window strategy for training and testing.

Table 2 lists the coherence and confidence of top 10% pairs (in this case,  $\text{Confidence} = \text{All_Confidence} = \text{Cosine}$ ). In this top 10% pairs, we find 11 pairs actionable in out-of-sample set. Among these pairs, two of them are relatively frequent pairs: Rule 1 – Stock 14, Rule 2 – Stock 24 (in this particular pair set, their association supports are larger than 20%).

**Table 2.** Actionability of top 10% rule-stock pairs

	Apr	May	Jun	Jul	Aug	Sept	Oct
<i>Coherence</i>	6.7%	14.3%	23%	14.3%	6.7%	14.3%	23%
<i>Confidence</i>	12.5%	25%	37.5%	25%	12.5%	25%	37.5%

In addition, we calculate the statistics in terms of fuzzy percentile ranking. For all pairs either in in-sample set or in out-of-sample set, we rank them in terms of five linguistic levels: *very positive*, *positive*, *watchable*, *negative* and *very negative* using asymmetry triangle membership function. We prune pair items at the point where more than 20% in-sample pairs (choose 5% in-sample pairs in this example) have presented in out-of-sample set. Table 3 provides the summarized statistics for the

same results. The above performance analysis shows that the confidence of fuzzy percentile ranked rule-stock pairs seems better than non-fuzzy top  $x\%$  pairs and crisp threshold based pairs. In this case, 9 actionable pairs are found, while the following pairs are found frequent: Rule 1-Stock 14, Rule 1-Stock 24, Rule 2-Stock 14, Rule 2-Stock 24 (association support  $\geq 20\%$ ).

**Table 3.** Actionability of fuzzy percentile ranked rule-stock pairs

	Apr	May	Jun	Jul	Aug	Sept	Oct
<i>Coherence</i>	20%	13.3%	37.5%	22.2%	13.3%	10%	9.1%
<i>Confidence</i>	50%	50%	75%	50%	50%	25%	25%
<i>All_Confidence</i>	40%	15.4%	42.9%	28.6%	15.4%	14.3%	12.5%
<i>Consine</i>	35.4%	27.7%	56.7%	37.8%	27.7%	18.9%	17.7%

However, our field studies in the market show that in real-world stock mining it is very hard to get predictability rate as high (say 80%) as reported in KDD literature. And even difficult thing is that it is very time-consuming and costly to develop a trading rule-stock pair which is workable when deployed into real market.

## 6 Conclusions

Pairs mining is emerging as a kind of very useful data mining applications, which can discover interesting pair relationship between a pair of instances in one or many classes. Real-world pairs mining must identify pairs in high dimensional data and consider user preference. This brings challenge to the existing association rule mining, correlation mining and genetic algorithms. This paper has proposed fuzzy genetic algorithms to tackle real-world pairs mining. A fuzzy genetic algorithm framework has been designed, which integrates fuzzy set and genetic algorithms and embeds user preference by developing fuzzy aggregation and ranking rules. They are used for mining financial pairs in stock markets. The financial pairs mining in ASX orderbook data has shown that the proposed approach is promising for detecting interesting pairs in high dimensional data and considering user preference.

## References

- [1] Allen, F. and Karjalainen, R. Using genetic algorithms to find technical trading rules, *Journal of Financial Economics*, 51, 245-271, 1999.
- [2] Buckley, J. Hayashi, Y. Fuzzy genetic algorithm and applications, *Fuzzy Sets and Systems* 61: 129-136, 1994.
- [3] Cao, L., Zhang, C. Domain-driven actionable knowledge discovery in the real world, *PAKDD2006, LNAI 3918*, 821 – 830, 2006.
- [4] Davis, L. (Ed.). *Handbook of genetic algorithms*. New York: Van Nostrand Reinhold, 1991
- [5] Han, J W., Kamber, M.: Data Mining: Technologies, Techniques, Tools, and Trends. Morgan Kaufmann Publishers.
- [6] Kovalerchuk, B., & Vityaev, E. *Data Mining in Finance: Advances in Relational and Hybrid Methods*, Kluwer Academic, 2000.
- [7] Lin, L., Cao, L. Mining In-Depth Patterns in Stock Market, *Int. J. Intelligent System Technologies and Applications*, 2006.
- [8] Zadeh, L.A. Fuzzy sets. *Information and Control*, 83: 338–353, 1965.

# A Novel Feature Selection Approach by Hybrid Genetic Algorithm

Jinjie Huang<sup>1,2</sup>, Ning Lv<sup>2</sup>, and Wenlong Li<sup>2</sup>

<sup>1</sup> Department of Automation, Shanghai Jiaotong University, Dongchuan Road 800,  
200240, China

jinjiehyh@yahoo.com.cn

<sup>2</sup> Department of Automation, Harbin University of Sci. & Tech., Xuefu Road 52, Harbin  
150080, China

ninglv@yahoo.com.cn

**Abstract.** Feature selection plays an important role in pattern classification. In this paper, a hybrid genetic algorithm (HGA) is adopted to find a subset of the most relevant features. The approach utilizes an improved estimation of the conditional mutual information as an independent measure for feature ranking in the local search operations. It takes account of not only the relevance of the candidate feature to the output classes but also the redundancy between the candidate feature and the already-selected features. Thus, the ability of the HGA to search for the optimal subset of features has been greatly enhanced. Experimental results on a range of benchmark datasets demonstrate that the proposed method can usually find the excellent subset of features on which high classification accuracy is achieved.

## 1 Introduction

Feature selection is one of the key topics in machine learning and other related fields [1]-[3]. Real-life datasets are often characterized by a large number of irrelevant or redundant features that may significantly hamper model accuracy and learning speed if they are not properly excluded. Feature selection involves finding a subset of features to improve prediction accuracy or decrease the size of the structure without significantly decreasing prediction accuracy of the classifier built using only the selected features [4].

Researches on feature selection are mainly focused on the twofold: criteria and searching strategies. An optimal subset is always relative to a certain criterion. Various criteria like *distance*, *dependence* and *consistency* measures have been used for feature selection [5] [6]. But these measures are all sensitive to the concrete values of the training data; hence they are easily affected by the noise or outlier data. Whereas the information measures, such as the entropy or mutual information, investigate the amount of information or the uncertainty of a feature for classification. In Shannon's information theory [7] [8], a model that does not increase the amount of information is useless and its prediction accuracy is not expected to be better than just a random guess [9]. Thus, the information measure is different from the above three measures by its metric-free nature: it depends only on the probability distribution of a random

variable rather than on its concrete values. Information measures have been widely used in feature selection [10]-[12], including the famous learning algorithm C4.5 [13].

The search for the best  $m$  features out of the  $n$  available is known to be an NP-hard problem and the number of local minima can be quite large [14]. Exhaustive evaluation of possible feature subsets is usually unfeasible in practice due to the large amount of computational effort required. A wide range of heuristic search strategies have been used including forward selection, backward elimination, hill-climbing, branch & bound algorithms [15], kernels [16] and the randomized algorithms like simulated annealing and genetic algorithms (GAs) [17]. Among these methods, GAs, under certain conditions, are able to find the global optimum of a multiple local-minima problem, and have shown their advantages for feature selection. However, the limitations accompanied with a simple GA have been uncovered in many applications, for example, premature convergence, poor ability of fine-tuning near local optimum points etc.. A practical and effective way to overcome these limitations is to hybridize the GA by incorporating domain-specific knowledge [18].

A novel hybrid GA designed to solve the feature selection problem with an improved mutual information criterion is proposed in this paper. The method utilizes an improved formula to compute the conditional mutual information between the candidate feature and the classes given a subset of features selected, and ranks every candidate feature in the local search process. The conditional mutual information takes account of not only the relevance of the candidate feature to the classes but also the redundancy between the candidate feature and the already-selected features. So a good subset of features can be comprised for machine learning, and the powerful global search ability of GA and the efficient local search heuristic strategies are combined successfully with the merits of both filters and wrappers inherited.

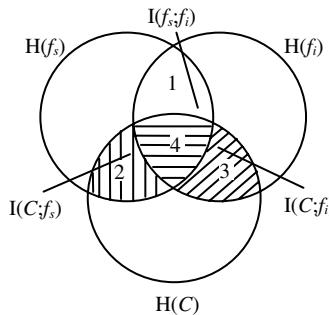
The rest of the paper is organized as follows: In Section 2, an improved estimation the conditional mutual information between the candidate feature and the classes given a subset of features selected is derived, which is used to rank candidate features in the local search process. In Section 3, a hybrid genetic algorithm for feature selection is proposed, and the local search schemes are discussed in detail. Then, the results of various experiments performed on a range of real-world datasets are reported in Section 4. Finally, Section 5 closes the paper with some concluding remarks.

## 2 Feature Ranking by Conditional Mutual Information

In feature selection problems, the task is to find those input features that contain as much information about output as possible. For this purpose, Shannon's information theory provides us a way to measure the information of random variables with entropy and mutual information [7] [8]. Mutual information between the class labels and the input features acts as a more general criterion. A large amount of mutual information between two random variables means that the two variables are closely related. The reasons why mutual information is not in wider use currently lie in computational difficulties for high dimensional space. The probability density functions of the variables are required, and mutual information involves numerical integration of functions of those, which leads to a high computational complexity [19]. However, evaluating mutual information between two scalar variables is feasible through histograms. Here

we derive an approximate estimation of mutual information for feature selection with only a series of two-variable mutual information calculations.

Assume  $A$  is the set of initial features describing a dataset,  $A = S \cup F$ , where  $S$  is the subset of already-selected features,  $F$  is the subset of unselected features,  $S \cap F = \emptyset$ .  $C$  is the output classes. For a feature  $f_i \in F$  to be selected, the amount of information about the class  $C$  newly provided by feature  $f_i$  rather than by the already-selected features in the current set  $S$  should be the largest among all the candidate features in  $F$ , that is, the conditional mutual information  $I(C; f_i | S)$  is maximized. For any  $f_s \in S$ ,  $f_i \in F$ , the relations between the entropies of  $f_s, f_i$  and  $C$  are illustrated in Fig.1.



**Fig. 1.** The relation between input features and output classes

As one can see that,

$$I(C; f_i, f_s) = I(C; f_s) + I(C; f_i | f_s) \quad (1)$$

where the joint mutual information  $I(C; f_i, f_s)$  is the area 2, 3 and 4, represented by the dashed area in Fig. 1.  $I(C; f_s)$  corresponds to the area 2 and 4, while  $I(C; f_i | f_s)$  corresponds to the area 3 in Fig. 1, representing the remaining mutual information between the output class  $C$  and the feature  $f_i$  for a given  $f_s$ .

Battiti's greedy feature selection algorithm [10] is to choose the feature  $f_i$  that maximizes the joint mutual information  $I(C; f_i, f_s)$ . As seen from Fig. 1, because  $I(C; f_s)$  is common for all the unselected features  $f_i \in F$  in computing  $I(C; f_i, f_s)$ , maximizing  $I(C; f_i, f_s)$  is in fact equivalent to maximizing  $I(C; f_i | f_s)$ , and there is no need to compute  $I(C; f_s)$ .

The conditional mutual information  $I(C; f_i | f_s)$  can be represented as

$$I(C; f_i | f_s) = I(C; f_i) - \{I(f_s; f_i) - I(f_s; f_i | C)\} \quad (2)$$

Under the suppose that the information is distributed uniformly throughout the regions of  $H(f_s), H(f_i)$  and  $H(C)$  in Fig. 1, and that the class  $C$  does not change the

ratio of the entropy of  $f_s$  to the mutual information between  $f_s$  and  $f_i$ , i.e. the following relation holds:

$$\frac{I(f_s; f_i | C)}{H(f_s | C)} = \frac{I(f_s; f_i)}{H(f_s)} \quad (3)$$

Then,  $I(f_s; f_i | C)$  can be represented as

$$I(f_s; f_i | C) = \frac{I(f_s; f_i)}{H(f_s)} H(f_s | C) \quad (4)$$

Substituting for (10)

$$\begin{aligned} I(C; f_i | f_s) &= I(C; f_i) - (1 - \frac{H(f_s | C)}{H(f_s)}) I(f_s; f_i) \\ &= I(C; f_i) - \frac{I(C; f_s)}{H(f_s)} I(f_s; f_i) \end{aligned} \quad (5)$$

However, when considering all of the already-selected features in  $S$ , there are in fact two extreme cases:

Case 1: For all selected features in  $S$ , their entropies are identical. This behaves in Fig. 1 that all the entropies of  $H(f_s)$ ,  $f_s \in S$  are overlapped exactly. Hence, we have the maximal value of  $I(C; f_i | S)$

$$I(C; f_i | S)_{\max} = I(C; f_i) - \frac{I(C; f_s)}{H(f_s)} I(f_s; f_i), \quad \forall f_s \in S \quad (6)$$

Case 2: For any two selected features in  $S$ , they are independent each other and their entropies are totally different. This behaves in Fig. 1 that there are no overlapping areas among all the entropies of  $H(f_s)$ ,  $f_s \in S$ . Hence, we have the minimal value of  $I(C; f_i | S)$

$$I(C; f_i | S)_{\min} = I(C; f_i) - \sum_{f_s \in S} \frac{I(C; f_s)}{H(f_s)} I(f_s; f_i), \quad \forall f_s \in S \quad (7)$$

In general cases, the entropies of selected features in  $S$  have some redundant information, and the corresponding areas in Fig. 1 are overlapped to a certain degree. So the conditional mutual information  $I(C; f_i | S)$  is bounded within the above two extreme cases.

$$I(C; f_i | S)_{\min} \leq I(C; f_i | S) \leq I(C; f_i | S)_{\max} \quad (8)$$

Therefore, it is reasonable to take the average redundancy for all the selected features in  $S$  to estimate  $I(C; f_i | S)$ . That is

$$I(C; f_i | S) = I(C; f_i) - \frac{1}{|S|} \sum_{f_s \in S} \frac{I(C; f_s)}{H(f_s)} I(f_s; f_i) \quad (9)$$

Here,  $|S|$  denotes the number of features in  $S$ . Furthermore, by many data experiments we find intuitively that the sum of information corresponding to the second term in (9) should be increased with the number of selected features in  $S$  increased, but the increment of information becomes smaller and smaller. Hence, we use a logarithm function to fit the likely tendency of variation of the conditional mutual information corresponding to the second term in (9).

$$I(C; f_i | S) = I(C; f_i) - \frac{1 + \beta \log_2^{|S|}}{|S|} \sum_{f_s \in S} \frac{I(C; f_s)}{H(f_s)} I(f_s; f_i) \quad (10)$$

Here a tuning factor  $\beta$  is added with a default value  $\beta = 1$  or  $\beta = 0.5$ . From the point of view of information theory, the base 2 logarithm  $\log_2^{|S|}$  is adopted.

With equation (10), we can now approximately compute the conditional mutual information  $I(C; f_i | S)$ , and only two-variable mutual information is needed. Thus, the “computationally impossible” calculation of the exact mutual information is substituted with a series of feasible calculations. A big value of  $I(C; f_i | S)$  indicates that the amount of information about the output class newly provided by the candidate feature  $f_i$  rather than by the current selected features in  $S$  is large. So feature  $f_i$  should be selected with a strong reason.

### 3 Feature Selection by Hybrid Genetic Algorithm

It has been proved theoretically that genetic algorithms can find the optimal solution to a problem in the sense of probability in a random manner. However, simple genetic algorithms have some weaknesses such as premature convergence and poor ability of fine-tuning near local optimum points in applications. To improve the fine-tuning capability and efficiency of simple GAs in feature selection problems, we here propose a hybrid GA with local search operations, in which the feature’s conditional mutual information  $I(C; f_i | S)$  is used as a measure to rank candidate features.

The local search measure  $I(C; f_i | S)$  has the following three properties:

- a) If  $I(C; f_i | S) = 0$ , it means that feature  $f_i$  can not offer any new classification information after the feature subset  $S$  is selected, and  $f_i$  should not be included in  $S$ ;
- b) If  $I(C; f_i | S) > 0$ , it means that feature  $f_i$  can offer some additional new classification information after the feature subset  $S$  is selected, and  $f_i$  has a close relationship to the output class  $C$ , so feature  $f_i$  should be included into  $S$ ;
- c) If  $I(C; f_i | S) < 0$ , it means that, instead of offering any new classification information after the feature subset  $S$  is selected, feature  $f_i$  does make a reduction in the amount of mutual information between the output class  $C$  and the feature subset  $S$ . This may be resulted from the fact that the amount of redundant information between feature  $f_i$  and the selected feature subset  $S$  is larger than the

amount of mutual information between feature  $f_i$  and the output class  $C$ . Obviously in this case, feature  $f_i$  should be excluded out of the consideration to be selected.

The local search operations are performed in a filter manner and inherit all the merits of filters such as high efficiency, fast computation and simplicity. They first remove the “bad genes” of every chromosome of a population, corresponding to remove the insignificant features of every subset generated by GA in each generation, and then learning machines are trained on the preprocessed feature subsets. The optimal subset of features is then found by the GA optimization through generations according to the performance of the series of learning machines trained.

The implementation of the hybrid genetic algorithm for feature selection mainly includes the encoding schemes of chromosomes, evaluating fitness function, local searching operations, designing for the selection, crossover and mutation genetic operations, and stopping criterion. In our algorithm, each individual in the population of chromosomes represents a candidate solution to the feature subset selection problem. A chromosome is encoded by a binary digit series that “1” means “selected” and “0” means “unselected”. Each digit (or gene) corresponds to a feature, so the gene length of a chromosome is equal to the total number of input features available. As a wrapper, the classification accuracy of the trained SVM classifier on the selected feature subset is used as the fitness function of the hybrid genetic algorithm. It is evaluated by the *k-fold cross-validation* method and  $k$  takes value 5~10 depending on the dataset scale.

As for genetic operations, the strategies are as follows. First, our design adopts the *rank-based roulette wheel* selection scheme. To guarantee the fast convergence ability, an *elitism* strategy is also used so that the best 10 percent of chromosomes in current population can enter the next generation directly without undergoing the crossover and mutation operations. Next, an adaptive crossover strategy is employed. When the total number of features is less than 20, the *single-point* crossover operator is used; while the total number of features is more than 20, the *double-point* crossover operator is used. The crossover probability  $p_c$  is assigned 0.7. Last, a simple mutation operator with the probability 0.1 is used.

The hybrid GA stops when the number of generations reaches the preset maximum generation  $T$ . So far, the overall procedure of the hybrid GA for feature selection is outlined below.

Procedure HGA for feature selection (HGAFS):  $M$  is the size of population  $P(t)$

```

Begin
  Initialize  $P(0)$ 
   $t=0;$ 
  While ( $t \leq T$ ) do
    Local improvement of each chromosome of  $P(t)$  by  $I(C; f_i | S)$ ;
    for  $i = 1$  to  $M$  do
      Train a classifier on each selected feature subset corresponding to each individual of  $P(t)$  and evaluate the fitness;
    end for
  
```

```

if stopping conditions are satisfied, break;
Selection operation to  $P(t)$  ;
Crossover operation to  $P(t)$  ;
Mutation operation to  $P(t)$  ;
for  $i = 1$  to  $M$  do
 $P(t+1) = P(t)$ 
End for
 $t = t + 1$ 
End while
End

```

## 4 Experimental Results

To study the performance of the hybrid GA for feature selection, 6 benchmark datasets are chosen from the UCI Machine Learning Repository (<http://www.ics.uci.edu/~mlearn/MLRepository.html>) for experiments. The datasets are summarized in Table 1. All these datasets are widely used by the data mining community for evaluating learning algorithms.

**Table 1.** Description of datasets

Dataset	Size	Classes	Candidate features		
			Continuous	Nominal	Total
Iris	150	3	4	0	4
Wine	178	3	13	0	13
Credit	690	2	6	8	14
WDBC	569	2	30	0	30
Satellite	6435	6	36	0	36
Sonar	208	2	60	0	60

The performance of the hybrid GA wrapper approach is compared with that of other two feature selection algorithms: Battiti's MIFS greedy selection algorithm and Quinlan's C4.5 decision tree algorithm. The results are mainly compared in terms of dimensionality reduction and the predictive accuracy.

Table 2 shows the initial number of candidate input features in each data set, the number of input features selected by the evaluated algorithms (Battiti's MIFS, C4.5 and HGAFS), and the reduction in data dimensionality (the portion of candidate input features that were excluded from the model). The C4.5 trees were built using Version 8 of the algorithm, and all the features that appear in at least one tree path were counted. The results show that the models produced by the HGAFS are significantly smaller than the decision trees built by C4.5 in four data sets out of six, and that the average difference between the two methods in dimensionality reduction is 15 percent of the number of available features. The Battiti's MIFS method tends to use fewer features than C4.5, but its average dimensionality reduction is still lower than the HGAFS average by 9 percent. This means that the HGAFS approach is a much "aggressive" dimensionality reducer than C4.5 and Battiti's MIFS methods.

Table 2 also shows, for each data set, the estimated predictive accuracy of the HGAFS approach versus the other two methods. The results were obtained with the *k-fold cross-validation* approach. Following a common practice, we have chosen the value of *k* to be 5 or 10 according to the size of the data sets.

**Table 2.** Comparative performances of the three methods

Datasets	Candidate input features	Selected input features			Dim. Reduction			Predictive accuracy		
		MIFS	C4.5	HGAFS	MIFS	C4.5	HGAFS	MIFS	C4.5	HGAFS
Iris	4	2	2	2	50.00%	50.00%	50.00%	95.33%	92.60%	96.67%
Wine	13	1	4	4	92.31%	69.23%	69.23%	79.78%	94.00%	96.63%
Credit	14	1	8	2	92.86%	42.86%	85.71%	86.43%	85.98%	86.73%
WDBC	30	15	7	6	50.00%	76.67%	80.00%	93.15%	93.68%	97.36%
Satellite	36	22	35	21	38.89%	2.78%	41.67%	90.45%	85.70%	91.47%
Sonar	60	39	12	8	35.00%	80.00%	86.67%	78.85%	72.14%	83.17%
Average	-	-	-	-	59.84%	53.59%	68.88%	87.33%	87.35%	92.01%

As one can see from Table 2, the predictive accuracy of the HGAFS method tends to be better than the accuracy of both C4.5 and Battiti's MIFS in almost all cases. Whereas the Battiti's MIFS algorithm does not show any advantages in predictive accuracy compared to the HGAFS method. The average predictive accuracy of Battiti's MIFS is nearly the same as that of C4.5, but it is lower than that of HGAFS by about 4 percent.

In all, the HGAFS method achieves the highest average predictive accuracy using only 30 percent of the number of available features on average.

## 5 Conclusions

A hybrid genetic algorithm for feature selection based on mutual information has been presented in this paper. The method utilizes the conditional mutual information between the candidate feature and the classes given a subset of already-selected features,  $I(C; f_i | S)$ , as a measure for feature ranking in the local search operations. Thus, both the relevance of the candidate feature to the output classes and the redundancy between the candidate feature and the already-selected features are taken into account. Experimental results on various types of real-world datasets strongly demonstrate the effectiveness of the proposed approach.

However, since the overall framework for feature selection adopts the wrapper method, so it inevitably inherits the weakness like a long run time needed. These problems should be further studied and improved in the future.

## Acknowledgement

This work was partly supported by the National Grand Fundamental Research 973 Program of China under Grant No.2002cb312200, by the National Natural Science

Foundation of China under Grant No.60575036, by the Natural Science Foundation of Heilongjiang Province of China under Grant No.F0316, and by the China Postdoctoral Science Foundation under Grant No.2004036321.

## References

1. B Guyon, I., Elisseeff A.: An introduction to variable and feature selection. *Journal of Machine Learning Research* 3 (2003), 1157-1182
2. Dash, M., Liu, H.: Feature selection for classification. *Intelligent Data Analysis* 1(1997) 131-156
3. Kohavi, R., John, G.H.: Wrappers for feature subset selection. *Artificial Intelligence* 97 (1997) 273-324
4. Koller, D., Sahami, M.: Toward optimal feature selection. In: *Proceedings of International Conference on Machine Learning*, Bari, Italy, (1996) 284-292
5. Yu L., Liu, H.: Efficient Feature Selection via Analysis of Relevance and Redundancy. *Journal of Machine Learning Research* 5 (2004) 1205-1224
6. Dash, M., Liu, H.: Consistency-based search in feature selection. *Artificial Intelligence* 151(1-2) (2003) 155-176
7. Shannon, C. E., Weaver, W.: *The Mathematical Theory of Communication*. Univ. Illinois Press, Urbana, IL (1949)
8. Cover, T. M., Thomas, J. A.: *Elements of Information Theory*, Wiley, New York (1991)
9. Last, M., Maimon, O.: A compact and accurate model for classification. *IEEE Transactions on Knowledge and Data Engineering* 16(2) (2004) 203-215
10. Battiti, R.: Using mutual information for selecting features in supervised neural net learning. *IEEE Transactions on Neural Networks* 5(4) (1994) 537-550
11. Kwak, N., Choi, C. H.: Input feature selection for classification problems. *IEEE Transactions on Neural Networks* 13(1) (2002) 143-159
12. Grall-Maes, E., Beauseroy, P.: Mutual information-based feature extraction on the time-frequency plane. *IEEE Transactions on Signal Processing* 50(4) (2002) 779-790
13. Quinlan, J. R.: Improved use of continuous attributes in C4.5. *Journal of Artificial Intelligence Research* 4 (1996) 77-90
14. Amaldi, E., Kann, V.: On the approximation of minimizing non zero variables or unsatisfied relations in linear systems. *Theoretical Computer Science* 209,(1998) 237-260
15. Somol, P., Pudil, P., Kittler, J.: Fast branch & bound algorithms for optimal feature selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26(7), (2004) 900-912
16. Baudat, G., Anouar, F.: Feature vector selection and projection using kernels. *Neurocomputing* 55(1-2), (2003) 21-38
17. Bhanu, B., Lin, Y. : Genetic algorithm based feature selection for target detection in SAR images. *Image and Vision Computing* 21(7), (2003) 591-608
18. Il-Seok Oh, Jin-Seon Lee, Byung-Ro Moon: Hybrid Genetic Algorithms for Feature Selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26(11), (2004) 1424-1437
19. Torkkola K.: Feature Extraction by Non-Parametric Mutual Information Maximization. *Journal of Machine Learning Research* 3 (2003) 1415-1438

# Evolutionary Ensemble Based Pattern Recognition by Data Context Definition

Mi Young Nam, In Ja Jeon, and Phill Kyu Rhee

Dept. of Computer Science & Engineering, Inha University

253, Yong-Hyun Dong, Incheon, South Korea

rera@im.inha.ac.kr, {juninjaa, pkrhee}@inha.ac.kr

**Abstract.** In this paper, we proposed evolutionary filter and classifier ensemble. We designed the face recognition system that consists of training module and testing module. The training module is evolution step, that process make filter and classifier combination. In testing step, we identified face recognition using knowledge from made training step. The filters are applied preprocessing step. Captured images are varying illuminant images so we proposed evolutionary preprocessing filter combinations and classifier ensemble for data context. The Proposed classifier selection for efficient object recognition based on evolutionary computation and data context knowledge called context based evolutionary system. In proposed method, we distinguish the data characteristics of input image and filter selects a classifier system accordingly using evolutionary algorithm. The proposed method is high more than single method.

## 1 Introduction

We adopt the genetic algorithm to explore a most effective subset of classifiers that produce best outputs for an identified context. Evolution method guides the scheme adaptive to varying environment. Adaptation to changing environments is an important application of GAs[1,2]. Genetic algorithm have been developed and used in many application areas such as optimization and search problems. GA is inspired by biological evolution process [11, 12]. GA is a well known searching algorithm in a huge search space via populationVarious approaches using GAs for feature selection problem can be found in [3, 4]. In [2], GA is used to select an optimal feature space of each classifier in a classifier fusion system. The goal of optimizing feature space is to search an minimum relevant subset from full feature space that can represent a target concept as well as possible. In this paper, we present a novel classifier selection method aiming at highly efficient object recognition by taking advantage of the evolutionary computing and the context-awareness. The knowledge of an individual context category and its associated chromosomes of effective classifiers are stored in the context knowledge base. Similar research can be found by [5]. We will deal with image objects the spatial boundaries of which can be well estimated in prior. Face images are in the class of well-defined image objects, the spatial boundaries of which can be well estimated in prior. Recently, face recognition becomes a popular task in object recognition area [6, 7].

The paper is organized as follows. In the section 2, we present the proposed context-awareness module and the overview of the proposed object recognition scheme, in the section 3, we discuss about the Proposed Classifier and Filter Selection. Finally, we give the experimental results and the concluding remarks in the section 4 and 5, respectively.

## 2 Data Context-Awareness

Context modeling can be performed by an unsupervised learning algorithm [16, 17]. Several types of unsupervised learning methods have been investigated, and Kohonen network is selected to be the most promising algorithm for constructing the data context model for face recognition. Kohonen network can be used to create an intuitive model of the important concepts contained in information [8, 9]. After a sufficient number of input vectors have been presented, network connection weights specify clusters, the point density function of which tends to approximate the probability density function of the input vectors. In addition, the connection weights will be organized such that topologically close nodes are sensitive to inputs that are similar. One can find that image in each cluster have similar light direction and intensity as expected. We use Kohonen algorithm, and this is constructed input layer and competitive layer. Also competitive layer is context cluster's center. Fig 1 shows example face images for Kohonen network's second layer, cluster's center. The context model generated the hybrid scan method.



**Fig. 1.** The context model of face images clustered into using hybrid scan vectorization and Kohonen network

Continuous-valued vectors of face image features which are presented sequentially without specifying the desired output. After a sufficient number of input vectors have been presented, network connection weights specify clusters, the point density function of which tends to approximate the probability density function of the input vectors. In addition, the connection weights will be organized such that topologically close nodes are sensitive to inputs that are similar. Table 1 and table 2 show vectorization result according vectorization method. We compared two methods, vertical scan and hybrid scan.

**Table 1.** Vertical vectorization

Vertical	class 0	class 1	class2	class3	class4	class5	class6	class7	class8
3	3617	3344	2381						
6	1448	1522	1522	1533	1794	1523			
9	1420	827	1315	1336	1243	697	999	726	779

**Table 2.** Hybrid vectorization

Hybrid	class 0	class 1	class2	class3	class4	class5	class6	class7	class8
3	2434	3854	3054						
6	1482	1481	1517	1563	1572	1727			
9	1195	953	988	971	856	1204	1281	1139	755

### 3 Classifier and Filter Ensemble

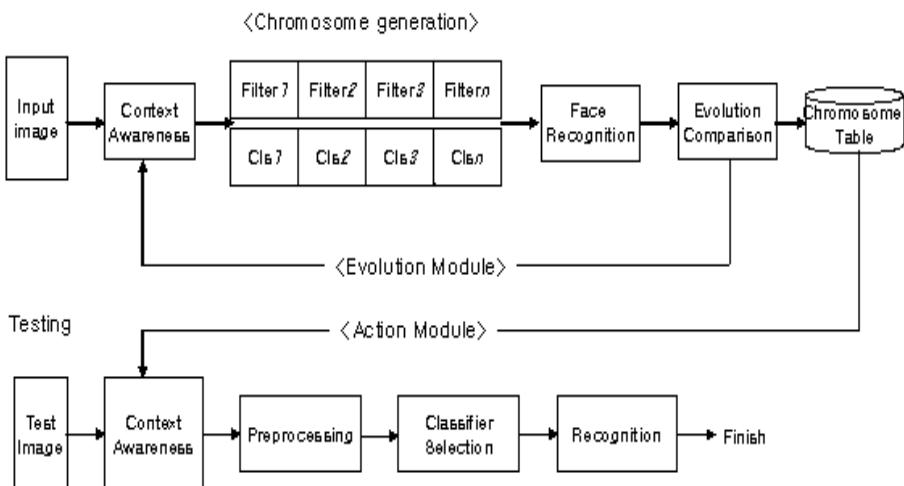
The proposed system is constructed evolution module and action module. Evolution module is training step that is decided filter and classifier selection and fusion for context cluster. Action module is testing step that process filtering and classifier following knowledge from evolution module, the action primitives of object recognition can be divided into three stages: preprocessing, feature representation, and class decision. We proposed method include the preprocessing and decision step.

In the proposed scheme, two types of processing are used: context awareness module and testing module inputs as discussed before.

In the testing step, data is a normal input to be combine filter, and the context data identified a data context of the knowledge dataset. We use a single input image as both context and action data in this paper. The proposed system has been tested in the area of face recognition. The illumination context can include lighting direction, brightness, contrast, and spectral composition. We adopt lighting direction and brightness in this experiment.

The evolutionary module evolution module of proposed system is implemented by Genetic algorithm (GA). GA explores the structure of the action module adaptive to a give image data subset. In the recognition mode, the system searches for a most effective classifier combination based on the identified category. The knowledge of effective classifier structure for a data context is described by the pair of data context category and corresponding artificial chromosome. Filter knowledge database is made classifier and filter selection and fusion. Context data is defined as any observable and relevant attributes that affects system environment, and its interaction with other entities and/or surrounding environment at an instance of time[10]. In this session, we will discuss data context modeling, identification, and knowledge accumulation for efficient object recognition.

## Training



**Fig. 2.** Block diagram of the proposed Evolutionary architecture for object recognition

**Step 1.** Identify the illumination situation in the context awareness module.

**Step 2.** Search for the chromosome from the chromosome database representing the optimal classifier structure corresponding to the identified illumination category.

**Step 3.** Perform the task of recognition using the restructured feature vector.

**Step 4.** If the system performance is measured to fall down below the predefined criterion, the system activates the evolution mode, and/or evolves the system periodically or when it is needed.

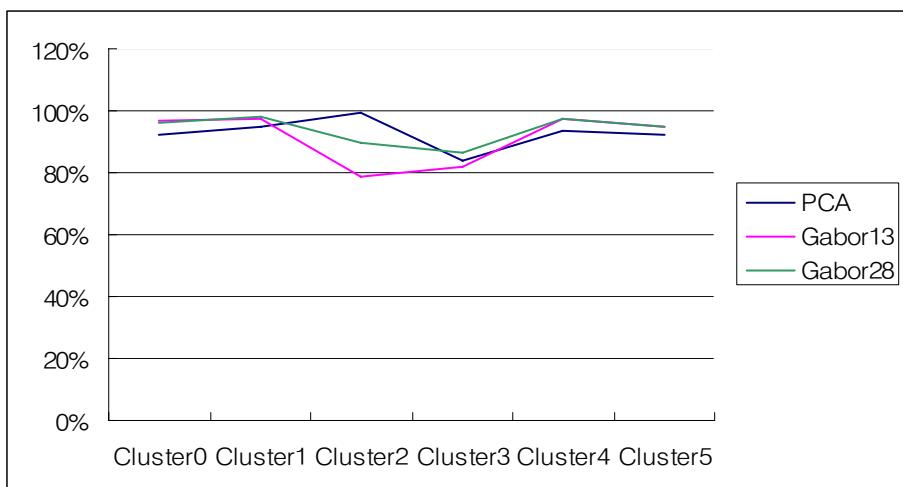
In the action module, we use three preprocessing components, histogram equalization, end-in contrast stretching, and illumination compensation. We use five feature representation, Eigenface, Gabor13, Gabor 28, and Gabor30. Three class decision methods are K-nn. And filter processing method is median filter, histogram equalization, contrast stretching filter and illumination compensation filter. Various approaches using GA for feature selection problem can be found in [11]. GA based optimization of classifier components can also be found in adaptive preprocessing based recognition [10], and neural network classifier systems. In this paper, all possible classifier system combinations are encoded as artificial chromosomes. However, GA can hardly be used under dynamically changing environment alone since they usually consume too much time to evaluate the population of chromosomes in the genospace until finding an optimal solution. The knowledge of an individual context category and its associated chromosomes of effective classifier systems are stored in the context knowledge base. In addition, once the context knowledge is constructed, the system can react to changing environments at run-time.

## 4 Experimental Results

The proposed method is tested in the area of face recognition using four data sets: Inha, FERET[20], AR, Yale[19]. Its performance is evaluated through extensive experiments to be superior to those of most popular methods, especially in changing illumination. In experiment, we decide filter combination and classifier combination. Table 3 shows the performance of successful face recognition resulting of six cluster modeling for non-evolutionary face recognition, proposed face recognition with hybrid scanning. Single classifier limited to enhance. Because the face recognition face images vary dynamic change. Therefore face recognition ratio is low in real time system. But our proposed method was efficient in bed illuminant images.

**Table 3.** The comparison of successful recognition rates between the non-evolutionary method and the proposed face recognition method in hybrid vectorization

Data context category	Non-evolutionary method	Proposed face recognition with the hybrid scanning
Cluster -0	91.10%	94.66%
Cluster -1	94.27%	97.45%
Cluster-2	92.37%	96.18%
Cluster-3	92.57%	96.26%
Cluster-4	94.27%	97.45%
Cluster-5	88.94%	92.63%
Average	91.93%	95.51%



**Fig. 3.** Face recognition using single classifier

**Table 4.** Face recognition ratio of FERET and Yale combined dataset

	Histogram Equalization	Median Filtering	Illumination Compensation	Proposed Method
Cluster-0	92.6%	93.5%	85.6%	94.2%
Cluster-1	90.5%	92.6%	83.2%	93.5%
Cluster-2	87.3%	85.2%	75.3%	82.4%
Cluster-3	91.2%	90.0%	83.2%	92.3%
Cluster-4	73.5%	76.5%	65.3%	80.3%
Cluster-5	65.3%	85.3%	84.5%	85.5%

In Table 3 and Table 4, we can see that face recognition rate is different for each cluster. Therefore we need to ensemble classifier and clustering method to well-divide face images.

## 6 Conclusion

We proposed the ensemble method for preprocessing and identify in environment face images. Ensemble method is efficient face recognition bring varying brightness images. The proposed method is tested using IT Lab., FERET, Yale. Its performance is evaluated through extensive experiments to be superior to those of most popular methods, especially in each cluster. In the paper, we combine filter and classifier using evolvable architecture. The proposed ensemble method has been evaluated in the area of face recognition. We make data context using Kohonen unsupervised learning algorithm. The face data context can be decided based on the light direction or brightness. The proposed scheme can optimize itself to a given data by using the identified data context and previously derived chromosome.

## References

- [1] C. Liu and H. Wechsler, : Evolutionary Pursuit and Its Application to Face recognition, IEEE Trans. on PAMI, Vol. 22, No. 6 ( 2000) 570-582.
- [2] Kuncheva L.I. and L.C. Jain, : Designing classifier fusion systems by genetic algorithms, IEEE Transactions on Evolutionary Computation, Vol.4, No.4 (2000) 327-336.
- [3] A. S. Georghiades, P. N. Belhumeur, and D. J. Kriegman, From Few to Many: Illumination Cone Models for face recognition under Variable Lighting and Pose, IEEE Trans. on PAMI, vol. 23 no. 6 (2001) 643-660.
- [4] M. Potzsch, N. Kruger, and C. Von der Malsburg, : Improving Object recognition by Transforming Gabor Filter responses, Network: Computation in Neural Systems, Vol.7, No.2 (1996) 341-347.

- [5] Ludmila I. Kuncheva, : Switching Between Selection and Fusion in Combining Classifiers: An Experiment, *IEEE Transactions on Systems, Man, and Cybernetics - part B: cybernetics*, Vol.32, No.2 (2002) 146-156.
- [6] X. Wang and X. Tang, : Random Sampling LDA for Face Recognition, *in Proceedings of CVPR*, vol.2 (2004) II-259- II-265.
- [7] X. Wang and X. Tang, : A Unified Framework for Subspace Face Recognition, *IEEE Trans. on PAMI*, Vol. 26, No. 9 ( 2004) 1222- 1228.
- [8] M.Y. Nam and P.K. Rhee, : A Novel Image Preprocessing by Evolvable Neural Network, *LNAI3214*, Vol.3, (2004) 843-854.
- [9] M.Y. Nam and P.K. Rhee, : An Efficient Face Recognition for Variant Illumination Condition, *ISPACS2005*, Vol.1 (2004) 111-115.
- [10] In Ja Jeon, Ki Sang Kwon, and Phill Kyu Rhee, Optimal Gabor Encoding Scheme for Face Recognition Using Genetic Algorithm, *KES 2004*, (2004) 227-236.
- [11] Shafer, S., Brumitt, B., and Meyers, B., The EasyLiving Intelligent Environment System, *CHI Workshop on Research Directions in Situated Computing*, (2000).

# Quantum-Behaved Particle Swarm Optimization with a Hybrid Probability Distribution

Jun Sun, Wenbo Xu, and Wei Fang

Center of Intelligent and High Performance Computing,  
School of Information Technology, Southern Yangtze University,  
No. 1800, Lihudadao Road, Wuxi,  
214122 Jiangsu, China  
[{sunjun\\_wx, xwb\\_sytu, wxfangwei}@hotmail.com](mailto:{sunjun_wx, xwb_sytu, wxfangwei}@hotmail.com)

**Abstract.** Based on the previous introduced Quantum-behaved Particle Swarm Optimization (QPSO), in this paper, a revised QPSO with novel iterative equation is proposed. While the iterative equation in the QPSO is educed from exponential distribution, the novel one derives from the distribution function of the sum of two random variables with exponential and normal distribution, respectively. The Revised QPSO also maintains the mean best position of the swarm as in the previous QPSO to make the swarm more efficient in global search. The experiment results on benchmark functions show that Revised QPSO has stronger global search ability than QPSO and PSO.

## 1 Introduction

The Particle Swarm Optimization (PSO) method is a member of a wider class of Swarm Intelligence methods used for solving Global Optimization (GO) problems. The method was originally proposed by J. Kennedy and R.C. Eberhart as a simulation of social behavior of bird flock and was initially introduced as an optimization method in 1995 [8]. PSO can be easily implemented and it is computationally inexpensive, since its memory and CPU speed requirements are low. One reason is that the method requires only the function value and does not require gradient information of the objective function of the GO problem under consideration. Another reason is that the primitive mathematical operators are used.

However, as demonstrated by F. Van Den Bergh [3], PSO is not a global convergence guaranteed algorithm because the particle is restricted to a finite sampling space for each of the iterations. This restriction weakens the global search ability of the algorithm and may lead to premature convergence in many cases. To overcome the shortcomings of the PSO, early concept of a Quantum-behaved Particle Swarm Optimization (QPSO) was disseminated in conference reports such as [11], [12]. In this paper, we introduce a hybrid probability distribution into QPSO and therefore propose a revised version of QPSO with novel iterative equation.

The rest of the paper is structured as follows. In Section 2, the principle of the PSO is introduced. The origin and development of QPSO is presented in Section 3 and the revised QPSO is proposed in Section 4. Section 5 gives the numerical results on some

benchmark functions. Some concluding remarks and future work are presented in the last section.

## 2 Particle Swarm Optimization

As a population-based evolutionary technique, a Particle Swarm Optimization (PSO) system simulates the knowledge evolution of a social organism, in which individuals (particles) representing the candidate solutions to the problem at hand, fly through a multidimensional search space to find out the optima or sub-optima. The particle evaluate its position to a goal (fitness) at every iteration, and particles in a local neighborhood share memories of their “best” positions. These memories are used to adjust particle velocities and their subsequent positions.

In the original PSO with  $M$  individuals, each individual is treated as an infinitesimal particle in the  $D$ -dimensional space, with the position vector and velocity vector of particle  $i$ ,  $X_i(t) = (X_{i1}(t), X_{i2}(t), \dots, X_{iD}(t))$  and  $V_i(t) = (V_{i1}(t), V_{i2}(t), \dots, V_{iD}(t))$ . The particle moves according to the following equations:

$$V_{ij}(t+1) = V_{ij}(t) + c_1 \cdot r_1 \cdot (P_{ij}(t) - X_{ij}(t)) + c_2 \cdot r_2 \cdot (P_{gj}(t) - X_{ij}(t)) \quad (1)$$

$$X_{ij}(t+1) = X_{ij}(t) + V_{ij}(t+1) \quad (2)$$

for  $i=1,2,\dots,M; j=1,2,\dots,D$ . The parameters  $c_1$  and  $c_2$  are called the acceleration coefficients. Vector  $P_i = (P_{i1}, P_{i2}, \dots, P_{iD})$  known as the *personal best position*, is the best previous position (the position giving the best fitness value so far) of particle  $i$ ; vector  $P_g = (P_{g1}, P_{g2}, \dots, P_{gD})$  is the position of the best particle among all the particles and is known as the *global best position*. The parameters  $r_1$  and  $r_2$  are two random numbers distributed uniformly in  $(0,1)$ , that is  $r_1, r_2 \sim U(0,1)$ . Generally, the value of  $V_{ij}$  is restricted in the interval  $[-V_{\max}, V_{\max}]$ .

Many revised versions of PSO algorithm are proposed to improve the performance since its origin in 1995. Two most important improvements are the version with an Inertia Weight [15],  $w$ , and a Constriction Factor [4],  $K$ . In the inertia-weighted PSO the velocity is updated by using

$$V_{ij}(t+1) = w \cdot V_{ij}(t) + c_1 \cdot r_1 \cdot (P_{ij}(t) - X_{ij}(t)) + c_2 \cdot r_2 \cdot (P_{gj} - X_{ij}(t)) \quad (3)$$

while in the Constriction Factor model the velocity is calculated by using

$$V_{ij}(t+1) = K \cdot [V_{ij}(t) + c_1 \cdot r_1 \cdot (P_{ij}(t) - X_{ij}(t)) + c_2 \cdot r_2 \cdot (P_{gj} - X_{ij}(t))] \quad (4)$$

where

$$K = \frac{2}{|2 - \varphi - \sqrt{\varphi^2 - 4\varphi}|}, \quad \varphi = c_1 + c_2, \quad \varphi > 4 \quad (5)$$

The inertia-weighted PSO was introduced by Shi and Eberhart and is known as the Standard PSO [15]. The addition of inertia weight or Constriction Factor leads to faster convergence of the PSO algorithm. Other improvements of PSO can be seen in literatures such as [2], [9], [10], etc.

### 3 Quantum-Behaved Particle Swarm Optimization

Trajectory analyses in [5] demonstrated the fact that convergence of PSO algorithm may be achieved if each particle converges to its local attractor  $p_i = (p_{i1}, p_{i2}, \dots, p_{iD})$  with coordinates

$$p_{ij}(t) = (c_1 r_1 P_{ij}(t) + c_2 r_2 P_{gj}(t)) / (c_1 r_1 + c_2 r_2), \text{ or } p_{ij}(t) = \varphi \cdot P_{ij}(t) + (1 - \varphi) \cdot P_{gj}(t) \quad (6)$$

where  $\varphi = c_1 r_1 / (c_1 r_1 + c_2 r_2)$ . It can be seen that the local attractor is a stochastic attractor of particle  $i$  that lies in a hyper-rectangle with  $P_i$  and  $P_g$  being two ends of its diagonal. We introduce the concepts of QPSO as follows.

Assume that each individual particle move in the search space with a  $\delta$  potential on each dimension, of which the center is the point  $p_{ij}$ . For simplicity, we consider a particle in one-dimensional space, with point  $p$  the center of potential. Solving Schrödinger equation of one-dimensional  $\delta$  potential well, we can get the probability distribution function D

$$D(x) = e^{-2|p-x|/L} \quad (7)$$

Using Monte Carlo method, we obtain

$$x = p \pm \frac{L}{2} \ln(1/u), \quad u \sim U(0,1) \quad (8)$$

The above is the fundamental iterative equation of QPSO.

In [11],  $L$  is evaluated by the distance between the particle's current position and point  $p$ . That is  $L = 2\alpha |p - x|$  and thus the following equation is obtained.

$$x = p \pm \alpha \cdot |p - x| \cdot \ln(1/u) \quad (9)$$

where  $\alpha$  is a parameter of the algorithm. For the particle in D-dimensional space, equation (9) will become

$$X_{ij}(t+1) = p_{ij}(t) \pm \alpha \cdot |p_{ij}(t) - X_{ij}(t)| \cdot \ln(1/u) \quad (10)$$

In [12], a global point called Mainstream Thought or Mean Best Position of the population is introduced into PSO. The global point, denoted as  $C$ , is defined as the mean of the personal best positions among all particles. That is

$$C(t) = (C_1(t), C_2(t), \dots, C_D(t)) = \left( \frac{1}{M} \sum_{i=1}^M P_{i1}(t), \quad \frac{1}{M} \sum_{i=1}^M P_{i2}(t), \quad \dots, \quad \frac{1}{M} \sum_{i=1}^M P_{iD}(t) \right) \quad (11)$$

where  $M$  is the population size and  $P_i$  is the personal best position of particle  $i$ . Then the value of  $L$  is evaluated by  $L = 2\alpha \cdot |C_j(t) - X_{ij}(t)|$  and the position are updated by

$$X_{ij}(t+1) = P_{ij}(t) \pm \alpha \cdot |C_j(t) - X_{ij}(t)| \cdot \ln(1/u) \quad (12)$$

where parameter  $\alpha$ , the same as that in equation (9), is called Contraction-Expansion (CE) Coefficient, which can be tuned to control the convergence speed of the algorithms. Generally, we always call the PSO with equation (12) Quantum-behaved Particle Swarm Optimization (QPSO), where parameter  $\alpha$  must be set as  $\alpha < 1.782$  to guarantee convergence of the particle [13]. In most cases,  $\alpha$  can be controlled to decrease linearly from  $\alpha_0$  to  $\alpha_f$  ( $\alpha_0 < \alpha_f$ ).

## 4 The Proposed Algorithm

Many experiments on benchmark functions and application to real-world problems indicate that QPSO will be a promising optimization problem solver. First of all, the introduced exponential distribution of positions makes QPSO global convergent. We also tested the QPSO with the position of a particle assume to distribute normally. In essence, normal distribution corresponds to the quantum harmonic oscillator potential well, while exponential distribution derives from Delta potential well as described in Section 3. The results generated by the QPSO with normal distribution on many benchmark functions show that it is more prone to encounter premature convergence than that with exponential distribution.

Furthermore, the introduction of mean best position into QPSO is another improvement of QPSO. In original PSO, each particle converges to the global best position independently. On the other hand, in the QPSO with mean best position  $C$ , each particle cannot converge to global best position without regard to its colleagues for there are wait among the particles. It is because that the distance between particle's current position and  $C$  determines the position distribution of the particle for next iteration. If the personal best positions of several particles are far from the global best position (these particle called lagged particles) while those of the other particles are near the  $P_g$ , the position  $C$  may be pulled away from  $P_g$  by lagged particles. When the lagged particles are chasing after their colleagues, say converging to  $P_g$ , the position  $C$  will be approaching  $P_g$  slowly. The distances between position  $C$  and the personal best positions of a particle near  $P_g$  don't decrease quickly, decelerating the convergences of the particles near  $P_g$ , and making them explore globally around  $P_g$  temporarily until the lagged ones are close to  $P_g$ . Therefore, in the QPSO with mean best position, the particle swarm does never abandon any lagged particle and seems to be more intelligent and more cooperative social organism. In a word, the wait among particles enhances the global search ability of QPSO greatly.

To make an ulterior improvement for QPSO, we may adopt other position distribution (or potential well). However, it has been shown by our preliminary experiments

that the exponential distribution maybe the best unitary distribution for QPSO. It can be expected that a hybrid distribution may improve QPSO.

Now we consider the iterative equation (8) and rewrite it as

$$x(t+1) = p(t) + A(t) \quad (13)$$

where  $A(t)$  is a random sequence that should converge to zero to guarantee that  $x(t)$  reach to point  $p(t)$ . In QPSO,  $A(t)$  complies with exponential distribution. Assume that  $A(t)$  is the sum of two random sequences, that is

$$A(t) = a(t) + b(t) \quad (14)$$

where  $a(t)$  and  $b(t)$  are two random sequences complying with different probability distributions. In our proposed method, we adopt that  $a(t)$  and  $b(t)$  are subject to normal distribution and exponential distribution respectively. Concretely, the two random sequences can be given by

$$a(t) = \pm \alpha |p(t) - x(t)| \cdot \ln(1/u) \quad (15)$$

$$b(t) = \beta |C(t) - x(t)| \cdot Rn \quad (16)$$

where  $u$  is a uniformly distributed random number in interval  $[0,1]$  and  $Rn$  is random number with standard normal distribution. Both  $\alpha$  and  $\beta$  are two parameters and also called Contraction-Expansion Coefficients. Therefore the iterative equation (13) can be written as

$$x(t+1) = p(t) \pm \alpha |p(t) - x(t)| \cdot \ln(1/u) + \beta |C(t) - x(t)| \cdot Rn \quad (17)$$

In D-dimensional search space, the iterative equation of each particle is

$$X_{ij}(t+1) = p_{ij}(t) \pm \alpha |p_{ij}(t) - X_{ij}(t)| \cdot \ln(1/u) + \beta |C_{ij}(t) - X_{ij}(t)| \cdot Rn \quad (18)$$

The first two items on the right of the above equation is the same as equation (12). To enhance the search ability of the Revised QPSO algorithm, we also introduced into the third item the mean best position  $C$  of swarm, guaranteeing that there are wait among particles. The Revised QPSO algorithm is outlined as follows.

#### **Revised QPSO Algorithm**

```

Initialize particles with random position Xi=X[i] [:];
Initialize personal best position by set Pi=Xi;
while the stop criterion is not met do
    Compute the mean best position C[:] by equation (11);
    for i = 1 to swarm size M
        If f(Xi)<f(Pi) then Pi=Xi; Endif
        Find the Pg=arg min f(P[g] [:]);
        for j=1 to D
            φ=rand(0,1); u=rand(0,1);
            p=φ*P[i][j]+(1-φ)*P[g][j];
            if (rand(0,1)>0.5)
                X[i][j]=p+α*abs(p- X[i][j])*ln(1/u)
                +β*abs(C[j]-X[i][j])*Rn;
            
```

```

Else
    X[i][j]=p- $\alpha$ *abs(p-X[i][j])*ln(1/u)
            + $\beta$ *abs(C[j]-X[i][j])*Rn;
Endif
Endfor
Endfor
Endwhile

```

The values of parameter  $\alpha$  and  $\beta$  can controlled as in the QPSO [11], [12], [13]. In our experiment for this paper,  $\alpha$  is set to be a fixed number, and  $\beta$  varies linearly over the running of algorithm.

## 5 Experiment

In this section five benchmark functions listed in Table 1 are tested for the performance comparison of the Revised QPSO (RQPSO) with Standard PSO (SPSO) and QPSO algorithms. These functions are all minimization problems with minimum objective function values zeros. The initial range of the population listed in Table 2 is asymmetry as used in [14], [15]. Table 2 also lists  $V_{\max}$  for SPSO.

**Table 1.** Expression of the five tested benchmark functions

	Function Expression	Search Domain
Sphere	$f_1(X) = \sum_{i=1}^n x_i^2$	$-100 \leq x_i \leq 100$
Rosenbrock	$f_2(X) = \sum_{i=1}^{n-1} (100 \cdot (x_{i+1} - x_i^2)^2 + (x_i - 1)^2)$	$-100 \leq x_i \leq 100$
Rastrigin	$f_3(X) = \sum_{i=1}^n (x_i^2 - 10 \cdot \cos(2\pi x_i) - 10)$	$-10 \leq x_i \leq 10$
Griewank	$f_4(X) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	$-600 \leq x_i \leq 600$
Shaffer's	$f_5(X) = 0.5 + \frac{(\sin(\sqrt{x_1^2 + x_2^2}))^2}{(1.0 + 0.001(x_1^2 + x_2^2))^2}$	$-100 \leq x_i \leq 100$

The fitness value is set as function value and the neighborhood of a particle is the whole population. We had 50 trial runs for every instance and recorded mean best fitness and standard deviation. In order to investigate the scalability of the algorithm, different population sizes  $M$  are used for each function with different dimensions. The population sizes are 20, 40 and 80. The maximum generation (iteration) is set as 1000, 1500 and 2000 corresponding to the dimensions 10, 20 and 30 for first four functions, respectively. The maximum generation for the last function is 2000. For SPSO, the acceleration coefficients are set to be  $c_1=c_2=2$  and the inertia weight is decreasing linearly from 0.9 to 0.4 as in [14], [15]. In experiments for QPSO, the value of CE Coefficient  $\alpha$  varies from 1.0 to 0.5 linearly over the running of the algorithm as in

**Table 2.** The initial range of population for all the tested algorithms and  $V_{\max}$  for SPSO

	<b>Initial Range</b>	<b><math>V_{\max}</math></b>
$f_1$	(50, 100)	100
$f_2$	(15, 30)	100
$f_3$	(2.56, 5.12)	10
$f_4$	(300, 600)	600
$f_5$	(30, 100)	100

[12], [13]. For Revised QPSO (RQPSO), the parameter  $\alpha$  is fixed at 0.6 and the value of  $\beta$  decreasing linearly from 0.9 to 0.4. The mean values and standard deviations of best fitness values for 50 runs of each function are recorded in Table 3 to Table 7.

**Table 3.** Numerical results on Sphere function

<b>M</b>	<b>Dim.</b>	<b>Gmax</b>	<b>SPSO</b>		<b>QPSO</b>		<b>RQPSO</b>	
			<b>Mean Best</b>	<b>St. Dev.</b>	<b>Mean Best</b>	<b>St. Dev.</b>	<b>Mean Best</b>	<b>St. Dev.</b>
20	10	1000	3.16E-20	6.23E-20	2.29E-41	1.49E-40	1.26E-057	8.59E-057
	20	1500	5.29E-11	1.56E-10	1.68E-20	7.99E-20	1.62E-028	1.10E-027
	30	2000	2.45E-06	7.72E-06	1.34E-13	3.32E-13	2.85E-018	1.49E-017
40	10	1000	3.12E-23	8.01E-23	8.26E-72	5.83E-71	1.46E-083	7.71E-083
	20	1500	4.16E-14	9.73E-14	1.53E-41	7.48E-41	1.84E-057	1.27E-056
	30	2000	2.26E-10	5.10E-10	1.87E-28	6.73E-28	1.43E-040	7.11E-040
80	10	1000	6.15E-28	2.63E-27	3.10E-100	2.10E-99	5.53E-096	3.43E-095
	20	1500	2.68E-17	5.24E-17	1.56E-67	9.24E-67	3.23E-080	1.16E-079
	30	2000	2.47E-12	7.16E-12	1.10E-48	2.67E-48	7.57E-065	4.65E-064

**Table 4.** Numerical results on Rosenbrock function

<b>M</b>	<b>Dim.</b>	<b>Gmax</b>	<b>SPSO</b>		<b>QPSO</b>		<b>RQPSO</b>	
			<b>Mean Best</b>	<b>St. Dev.</b>	<b>Mean Best</b>	<b>St. Dev.</b>	<b>Mean Best</b>	<b>St. Dev.</b>
20	10	1000	94.1276	194.3648	59.4764	153.0842	29.7713	59.0517
	20	1500	204.337	293.4544	110.664	149.5483	66.8258	73.5706
	30	2000	313.734	547.2635	147.609	210.3262	89.8029	122.7985
40	10	1000	71.0239	174.1108	10.4238	14.4799	16.8486	21.4448
	20	1500	179.291	377.4305	46.5957	39.536	46.3673	39.6699
	30	2000	289.593	478.6273	59.0291	63.494	71.8062	59.9621
80	10	1000	37.3747	57.4734	8.63638	16.6746	8.8899	10.6006
	20	1500	83.6931	137.2637	35.8947	36.4702	42.6273	37.3609
	30	2000	202.672	289.9728	51.5479	40.849	61.1832	54.0061

The numerical results show that both QPSO and RQPSO are superior to SPSO except on Shaffer's f6 function. On Shpere Function the RQPSO works better than QPSO and SPSO in most cases except when swarm size is 80 and the dimension is 10. On Rosenbrock function, the RQPSO outperforms the QPSO when the swarm size is 20, but it does not show the better performance than QPSO when the swarm size is

**Table 5.** Numerical results on Rastrigrin function

M	Dim.	Gmax	SPSO		QPSO		RQPSO	
			Mean Best	St. Dev.	Mean Best	St. Dev.	Mean Best	St. Dev.
20	10	1000	5.5382	3.0477	5.2543	2.8952	5.2436	2.6304
	20	1500	23.1544	10.4739	16.2673	5.9771	15.4886	5.9381
	30	2000	47.4168	17.1595	31.4576	7.6882	27.1845	8.0140
40	10	1000	3.5778	2.1384	3.5685	2.0678	2.9874	1.5604
	20	1500	16.4337	5.4811	11.1351	3.6046	11.4061	3.6145
	30	2000	37.2796	14.2838	22.9594	7.2455	22.2962	6.4349
80	10	1000	2.5646	1.5728	2.1245	2.2353	2.2353	1.3886
	20	1500	13.3826	8.5137	10.2759	6.6244	8.6963	3.3260
	30	2000	28.6293	10.3431	16.7768	4.4858	17.1859	4.7305

**Table 6.** Numerical results on Griewank function

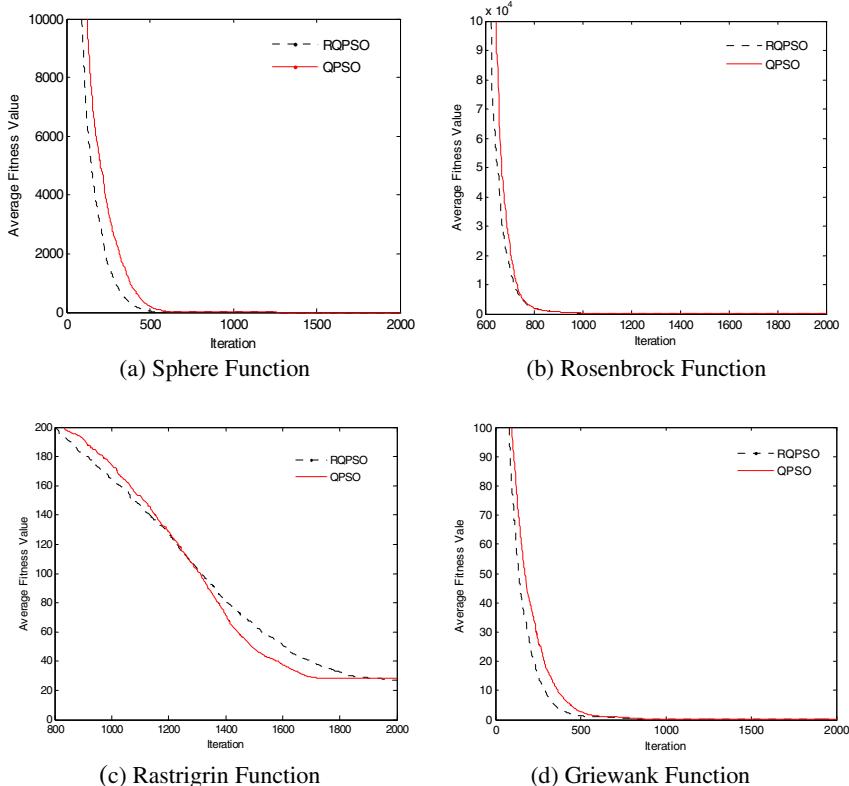
M	Dim.	Gmax	SPSO		QPSO		RQPSO	
			Mean Best	St. Dev.	Mean Best	St. Dev.	Mean Best	St. Dev.
20	10	1000	0.09217	0.0833	0.08331	0.06805	0.0748	0.0859
	20	1500	0.03002	0.03255	0.02033	0.02257	0.0166	0.0255
	30	2000	0.01811	0.02477	0.01119	0.01462	0.0088	0.0132
40	10	1000	0.08496	0.0726	0.06912	0.05093	0.0640	0.0563
	20	1500	0.02719	0.02517	0.01666	0.01755	0.0161	0.0155
	30	2000	0.01267	0.01479	0.01161	0.01246	0.0083	0.0112
80	10	1000	0.07484	0.07107	0.03508	0.02086	0.0398	0.0362
	20	1500	0.02854	0.0268	0.0146	0.01279	0.0138	0.0131
	30	2000	0.01258	0.01396	0.01136	0.01139	0.0066	0.0086

**Table 7.** Numerical results on Shaffer's f6 function

M	Dim.	Gmax	SPSO		QPSO		RQPSO	
			Mean Best	St. Dev.	Mean Best	St. Dev.	Mean Best	St. Dev.
20	2	2000	2.782E-04	0.001284	0.001361	0.003405	5.921E-004	0.0023
40	2	2000	4.744E-05	3.593E-05	3.891E-04	0.001923	3.698E-008	7.005E-008
80	2	2000	2.568E-10	3.134E-10	1.723E-09	3.303E-09	6.517E-009	1.428E-008

40 and 80. On Rastrigrin function, although it is shown that the RQPSO generated better results in some cases, its advantages over the QPSO are not remarkable. Therefore we can conclude that the two QPSOs have comparable performance on this function. On Griewank function, RQPSO is superior to the QPSO, particularly when the dimension of the problem is high. On Shaffer's function, the RQPSO shows its stronger ability to escape the local minima 0.0097 than the QPSO. Generally speaking, the Revised QPSO has better global search ability than QPSO.

The figure 1 shows the convergence process of the RQPSO and QPSO on the first four benchmark functions with dimension 30 and swarm size 20. It is shown that the RQPSO converge more quickly than the QPSO except on Rastrigrin function.



**Fig. 1.** Convergence process of the RQPSO and QPSO on the first four benchmark functions with dimension 30 and swarm size 20 averaged on 50 trail runs

## 6 Conclusion

In this paper, a Revised QPSO with novel iterative equation is proposed. The iterative equation derives from the hybrid of exponential and normal distributions. In the equation, the mean best position is also introduced to guarantee that there occurs wait among particles. The numerical results on benchmark functions show that the Revised QPSO enhance the global search ability of QPSO efficiently.

Our future work for this novel QPSO will focus on convergence analysis of the algorithm and finding out a more efficient parameter control method. Moreover, we will also be devoted to applying the novel QPSO to many real world problems.

## References

1. Angeline, P.J.: Evolutionary Optimization Versus Particle Swarm Optimization: Philosophy and performance Differences. *Evolutionary Programming VII*, Lecture Notes in Computer Science 1447. Springer-Verlag, Heidelberg (1998) 601-610

2. Angeline, P.J.: Using Selection to Improve Particle Swarm Optimization. Proc. 1998 IEEE International Conference on Evolutionary Computation. Piscataway, NJ (1998) 84-89
3. Van den Bergh, F.: An Analysis of Particle Swarm Optimizers. PhD Thesis. University of Pretoria, South Africa (2001)
4. Clerc, M.: The Swarm and Queen: Towards a Deterministic and Adaptive Particle Swarm Optimization. Proc. 1999 Congress on Evolutionary Computation. Piscataway, NJ (1999) 1951-1957
5. Clerc, M., Kennedy, J.: The Particle Swarm: Explosion, Stability, and Convergence in a Multi-dimensional Complex Space. IEEE Transactions on Evolutionary Computation, Vol. 6, No. 1. Piscataway, NJ (2002) 58-73
6. Eberhart, R.C., Shi, Y.: Comparison between Genetic Algorithm and Particle Swarm Optimization. Evolutionary Programming VII, Lecture Notes in Computer Science 1447, Springer-Verlag, Heidelberg (1998) 611-616
7. Holland, J.H.: Adaptation in Natural and Artificial Systems. The University of Michigan Press, Michigan (1975)
8. Kennedy, J., Eberhart, R.C.: Particle Swarm Optimization. Proc. IEEE 1995 International Conference on Neural Networks, IV. Piscataway, NJ (1995) 1942-1948
9. Kennedy, J.: Small worlds and Mega-minds: Effects of Neighborhood Topology on Particle Swarm Performance. Proc. 1999 Congress on Evolutionary Computation. Piscataway, NJ (1999) 1931-1938
10. Suganthan, P.N.: Particle Swarm Optimizer with Neighborhood Operator. Proc. 1999 Congress on Evolutionary Computation, Piscataway, NJ (1999) 1958-1962
11. Sun, J., Feng, B., Xu, W.-B.: Particle Swarm Optimization with Particles Having Quantum Behavior. Proc. 2004 Congress on Evolutionary Computation, Piscataway, NJ (2004) 325-331
12. Sun, J., Xu, W.-B., Feng, B.: A Global Search Strategy of Quantum-behaved Particle Swarm Optimization. Proc. 2004 IEEE Conference on Cybernetics and Intelligent Systems, Singapore (2004) 111-115
13. Sun, J., Xu, W.-B., Feng, B.: Adaptive Parameter Control for Quantum-behaved Particle Swarm Optimization on Individual Level. Proc. 2005 IEEE International Conference on Systems, Man and Cybernetics. Piscataway, NJ (2005) 3049-3054
14. Shi, Y., Eberhart, R.: Empirical Study of Particle Swarm Optimization. Proc. 1999 Congress on Evolutionary Computation. Piscataway, NJ (1999) 1945-1950
15. Shi, Y., Eberhart, R.C.: A Modified Particle Swarm. Proc. 1998 IEEE International Conference on Evolutionary Computation. Piscataway, NJ (1998) 69-73

# A Selection Scheme for Excluding Defective Rules of Evolutionary Fuzzy Path Planning

Jong-Hwan Park<sup>1</sup>, Jong-Hwan Kim<sup>2</sup>, Byung-Ha Ahn<sup>1</sup>, and Moon-Gu Jeon<sup>1</sup>

<sup>1</sup> Dept. of Mechatronics, Gwangju Institute of Science and Technology, South Korea  
*{jhpark, bayhay, mgjeon}@gist.ac.kr*

<sup>2</sup> Dept. of Electrical Engineering and Computer Science, Korea Advanced Institute of Science and Technology, South Korea  
*johkim@vivaldi.kaist.ac.kr*

**Abstract.** This paper proposes a new selection mechanism in evolutionary algorithm for fuzzy systems that can be applied to robot learning of shooting ability in robot soccer. In generic evolutionary algorithms, evaluation and selection are performed on the chromosome level, where a selected chromosome may include non-effective or bad genes. This may lead to an increase in the uncertainty of the solutions. To solve this problem, we propose a rule-scoring method for gene level selection, which grades genes at the same position in the chromosomes. This method is applied to a fuzzy path planner for the shooting of a soccer robot, where each fuzzy rule is encoded as a gene. Simulation and experimental results show the effectiveness and the applicability of the proposed method.

## 1 Introduction

Fuzzy logic control has been proven effective for complex, nonlinear, and imprecisely defined processes for which standard analytical model-based control techniques are impracticable [1,4]. However, a common bottleneck encountered is that the derivation of fuzzy control rules is often time consuming and difficult, and relies to a great extent on so-called process experts. An automated way to design fuzzy systems might be preferable. Therefore, more attention has been paid to the problem of how to construct a suitable rule base for a given task, and numerous researches have been conducted to automate the knowledge acquisition step in a fuzzy system design using evolutionary algorithms (EAs) [10,12].

The evolutionary fuzzy systems employ the paradigm of chromosomes whose components are rules as genes. While the chromosomes are struggling to survive, all the rules are not used every time, but a part of them is used for carrying out fuzzy reasoning. Accordingly, one best rule (gene) for a certain state may not be used for another state or even may negatively affect other states. Since the rule sets are evaluated and selected on a chromosome level, a defective gene contributing to few cases can survive to deteriorate the performance of other cases. Therefore, the convergence of the evolutions has a variance and brings out uncertainty, and the reliability of solutions is damaged.

To overcome this problem, we propose a rule-scoring methodology to deal with each gene in a selection step. In evaluation, each gene in a chromosome is graded by a score for survival competition and the parents for the next generation are reproduced on the

basis of superiority among the genes at the same position of chromosomes in the population (that are equivalent to alleles). We calculate the variance to compare solution reliability of the proposed approach with that of the conventional methodology. Evolutions are repeatedly carried out for a fuzzy inference system, and solution distributions are examined in a simulation analysis. As a result, the small variance in solutions shows that the consistency improves. The proposed method also brings better performance with fast convergence. Moreover, the real robot also runs faster with small variance of the records in repeated experiments.

The evolved fuzzy system to be examined in this study is a path planner for shooting in robot soccer. The shooting behavior is a fundamental function, and it can be viewed as a posture control where the non-holonomic constraints of wheeled mobile robots make it difficult to derive a stable trajectory control law [9]. The fuzzy logic controller (FLC) providing shooting ability consists of two levels: the fuzzy planner and the fuzzy motion controller [11]. The path generated by the fuzzy planner consists of singleton values for the direction at sampled positions for the optimal trajectory to the ball, like a univector field [8].

This paper is organized as follows. In Section 2, the overall structure of the FLC is described, in which a fuzzy planner and a fuzzy motion controller are developed for posture control. Section 3 explains the evolutionary fuzzy system and the rule-scoring methodology. The proposed scheme is analyzed in both simulations and experiments for a robot soccer system in Section 4. Finally, concluding remarks follow in Section 5.

## 2 Fuzzy Controller and Target System

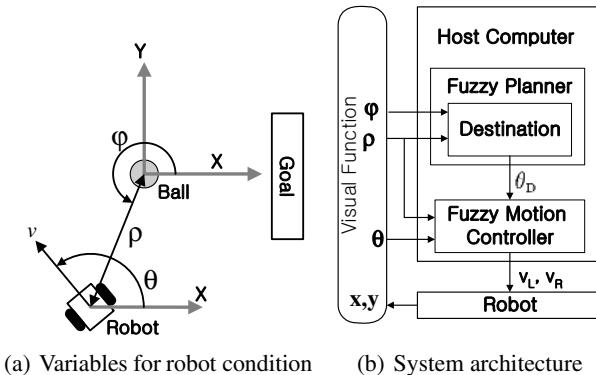
### 2.1 Fuzzy Logic Controller

FLCs are rule-based systems that successfully incorporate the flexibility of human-decision making into autonomous machinery by means of the use of fuzzy set theory. Rules take the form of IF *conditions* THEN *actions*, where conditions and actions are linguistic terms which are described by membership functions. The fuzzy rule-base of the FLC is composed of a number of such fuzzy rules, and this rule-base is used to produce precise output values according to actual input values.

The fuzzy membership function for the  $i$ -th variable is defined as an triangle, and singleton membership functions are used as the output fuzzy sets in order to simplify the defuzzification method [5,8]. This paper employs Mandani's style MIN-MAX type inference engine and the center of average defuzzification, since this combination yields the basic implementation of the fuzzy control algorithm.

### 2.2 Robot Soccer System with Fuzzy Controller

The robot soccer system for MiroSot [7,13] consists of three parts: a visual function of locating objects on the field, a host computer which calculates strategies and decides actions, and robots which follow actions transmitted by the computer as shown in figure 1. The fuzzy logic controller implemented in the host computer generates and sends the



**Fig. 1.** Overall structure of fuzzy posture controller

velocities to the robot through radio frequency communication to enable the shooting of the robot.

The three posture variables ( $\rho, \varphi, \theta$ ) (Fig. 1(a)) are required to achieve shooting ability. The controller is decomposed into two sub-controllers (Fig. 1(b)) in a hierarchical manner in which each takes only two variables as input.

One of the sub-controllers is the fuzzy planner and the other is the fuzzy motion controller. The planner is for generating a global path connecting the present robot posture to the ball, meeting the non-holonomic constraints. The fuzzy motion controller then commands using robot wheel velocities the robot to follow this desired path at the current robot posture.

**Fuzzy Planner.** The fuzzy planner is for generating a path globally that satisfies the constraints by calculating the desired robot's heading angle  $\theta_D$  at each relative position ( $\rho, \varphi$ ) in polar coordinates. Since the lower half plane is symmetric to x-axis, only the upper-half plane can be considered. The input space is divided into 7 membership functions of isosceles triangle from at intervals of 10 cm for  $\rho$  and 30 degree for  $\varphi$ , respectively. The forty-nine rules are obtained using  $\theta_D$  at sampled positions by evolutionary algorithm.

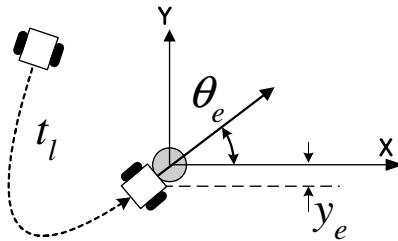
**Fuzzy Motion Controller.** In Figure 1, the fuzzy motion controller receives  $\theta_D$  from the fuzzy planner and robot posture information ( $\rho, \theta$ ) from the vision. Then the motion controller generates appropriate left and right wheel velocities to make  $\theta$  following  $\theta_D$ . For this conventional problem of mobile robots, the following heuristics are incorporated:

- If  $\rho$  large  $\rightarrow v_L, v_R$  large
- If  $|\theta_e| = |\theta_D - \theta|$  large  $\rightarrow |v_L - v_R|$  large

Table 1 shows the rule for right wheel velocity. Left wheel velocity is symmetrical with respect to  $\theta_e = 0$  and one unit corresponds to 1.534 cm/sec in the table.

**Table 1.** Rules for right wheel velocity

$v_R$	$\rho$						
$\theta_e$	NE	AN	SN	MD	SF	AF	FA
NB	-35	-27	-27	-3	-3	-3	-3
NM	-25	8	8	18	31	31	42
NS	-15	15	22	35	57	67	67
ZE	30	30	50	60	90	100	100
PS	15	40	44	65	82	92	92
PM	25	51	51	61	68	68	77
PB	35	63	63	67	67	67	67

**Fig. 2.** The performance factors of shooting ability

### 3 Evolutionary Fuzzy System

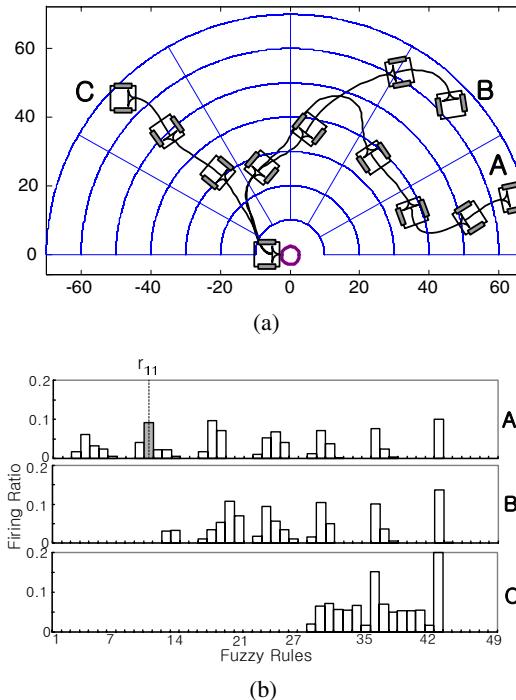
#### 3.1 Evolutionary Learning of Fuzzy System

In this paper, we simply define the rule combination set,  $R_i$ , as a chromosome in which the rule,  $r_{i,h}$  is contained in  $h$ -th gene of  $i$ -th chromosome, because all the membership functions are defined as isosceles triangles for partitioning input space and singleton values for output variables as described in Section 2.

Mutation is the self-adaptive Gaussian operator which is commonly used in evolutionary algorithms [6]. The selection scheme is based on  $q$ -tournament. A  $(\mu, \lambda)$ -evolution strategy [2] is used, and an elite chromosome is preserved by the elitism [3]. The performance index ( $PI$ ) of the  $i$ -th chromosome is defined as the sum of evaluations for  $j$ -th training trajectories as follows:

$$PI_i = \sum_j K_t \cdot t_l + K_p \cdot |\theta_e| + K_d \cdot y_e^2, \quad (1)$$

where  $K_t, K_p$  and  $K_d$  are coefficients for weights. How well the robot has shot the ball is evaluated in three parts, elapsed time to reach the ball,  $t_l$ , heading direction error,  $\theta_e$ , and drift at an impact point,  $y_e$ , to transfer the momentum to the ball. Fig. 2 shows these three parameters for a kicking situation.



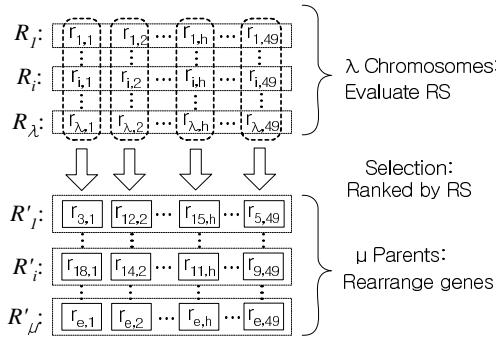
**Fig. 3.** Rule firing ratio for trajectories, A, B and C

### 3.2 Rule-Score Selection Method

Generic EA to obtain the optimal parameters of an evolutionary fuzzy system requires training for many points in the input space. When the evolutionary fuzzy system works with a given chromosome (rule set), all the genes (rules) may not contribute to the result of each training case. Only a part of genes is employed to obtain the result for a training point. Even if the fitness of a chromosome is excellent, inferior genes in the chromosome can survive, because the EA evaluates and selects chromosomes on the basis of their total fitness.

Figure 3 shows the normalized rule firing ratio for three trajectories (A, B and C) for a given fuzzy planner and uneven activation of rules. Let us assume that the rule set performs well for cases B and C, but not for case A. Rule  $r_{11}$  does not positively contribute to overall fitness, because it only triggers for the poor case A. As the evaluation in the conventional EA is carried out on the chromosome level, when the fitness of the chromosome is better than that of others, it will be selected. Thus, the rule  $r_{11}$  that performs poorly for A, will survive, since it is in the same chromosome that performs well generally. However rule  $r_{11}$  should be changed in order to search for a better solution at the other training point.

Obviously, the evaluation on the level of gene should be considered for a better solution. As the contributions being calculated and compared, the genes are distinguished from each other and the chromosome can be recomposed to remove the inferior genes.



**Fig. 4.** Reproduction with rule score ranking

The fuzzy system can compute each rule's contribution by its firing strength, and can rearrange the rules on the same column in chromosomes. Based on these properties, we propose a rule-scoring method as a new selection scheme in evolutionary fuzzy systems.

The rule-scoring method has two steps: calculation of each rule score (RS) and reproduction. The rule score is calculated to grade rules on the same column in rule sets. The rule score considers firing rate of each rule (gene) and its achievement for each training case. First, we calculate the firing strength ratio of  $h$ -th rule of  $i$ -th rule set for  $j$ -th training case as follows:

$$F_{i,j}(h) = \sum_t \frac{w_h}{\sum_h w_h} \Big|_{i,j}, \quad (2)$$

where  $w_h$  is the firing strength of the  $h$ -th rule and  $t$  is the travel time to the ball. Then, we normalize the firing ratio of each rule for the trajectory by

$$RF_{i,j,h} = \frac{F_{i,j}(h)}{\sum_h F_{i,j}(h)}. \quad (3)$$

Hence, the rule score for  $h$ -th rule in the  $i$ -th rule set for  $m$  training cases is obtained as

$$RS_{i,h} = \sum_{j=1}^m (RF_{i,j,h} \times CW_{i,j}), \quad (4)$$

where  $CW_{i,j}$  is the count of wins of the  $i$ -th rule set among  $n$  rule sets by  $q$ -tournament for the  $j$ -th training case.

The next step is shown in figure 4 to reproduce the parents for next generation. The rule set,  $R'_i$ ,  $i = 1, \dots, \mu$ , is recombined at the gene-level. The genes are ranked on the basis of their score in the same column (position) of chromosomes. The genes of the first rank in each position,  $h$ , are collected into the first chromosome,  $R'_1$  and the genes of the second rank make a second one,  $R'_2$  and so on. Through this process, eventually an appropriate chromosome is obtained.

## 4 Evolutionary Learning and Experiments

### 4.1 Robot Soccer System

To demonstrate the effectiveness and applicability of the proposed method, evolutionary learning of the fuzzy rules and real robot experiments were performed for the shooting ability of a soccer robot for MiroSot [7,13]. The MiroSot soccer playing field is of size 220cm  $\times$  180cm. and the dimension of soccer robot itself with two driving wheels, is 7.5cm  $\times$  7.5cm  $\times$  7.5cm. For real robot manipulation, the vision system consists of a Samsung SDC410 CCD camera with a MATROX Meteor-II image grabber that operates at a rate of 60 frames/second. The host computer is a Pentium processor with a clock speed of 2GHz. The vision system extracts an estimate for the posture information of both robot and ball, which is then transmitted to the FLC's. The path planning and control algorithms were implemented using Visual C++ 6.0.

### 4.2 Evolution Results

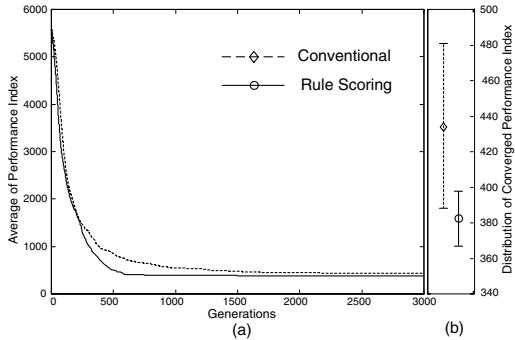
The evolutionary learning techniques introduced in the previous section were used to assist in the generation of the fuzzy rules for the path planning algorithm. Given the robot size (7.5cm) and the ball diameter (4.27cm), the robot is required to be stopped when the center of robot arrived at a distance of within 5 cm from the ball position and calculate performance index by eq. (1) in simulation.

Because the fuzzy planner consists of 49 rules for generating  $\theta_D$ , a chromosome consists of 49 genes which represent each rule. The evolutionary algorithm introduced in Section 3 was performed using a  $(\mu, \lambda)$  strategy where the number of parents ( $\mu$ ) and offspring ( $\lambda$ ) were set to 10 and 20, respectively. The  $q$ -tournament selects 10 competitors in each round. The coefficients to calculate the PI of eq. (1) were manually tuned and finalized at  $K_t = 10$ ,  $K_p = 1$  and  $K_d = 3$ . Forty-eight points were selected for an exercise in which all the genes were used at least once. The evolutionary algorithm was evolved for 3,000 generations in both the conventional and proposed rule-scoring methods. To conduct statistical analysis, the evolutionary algorithm was applied to each method 62 times.

Table 2 shows that the proposed method performs better with smaller mean value standard deviation, and figure 5 demonstrates faster convergence. Moreover, the coefficient of variation was reduced by about 50% on average. These results imply a tendency for the proposed algorithm to consistently find more optimal solutions than the conventional method.

**Table 2.** Converged performance index

Algorithm	Mean	Std
Conventional	434.39	45.88
Rule-score selection	382.51	15.22



**Fig. 5.** Convergence of performance index and standard deviation

### 4.3 Experimental Results

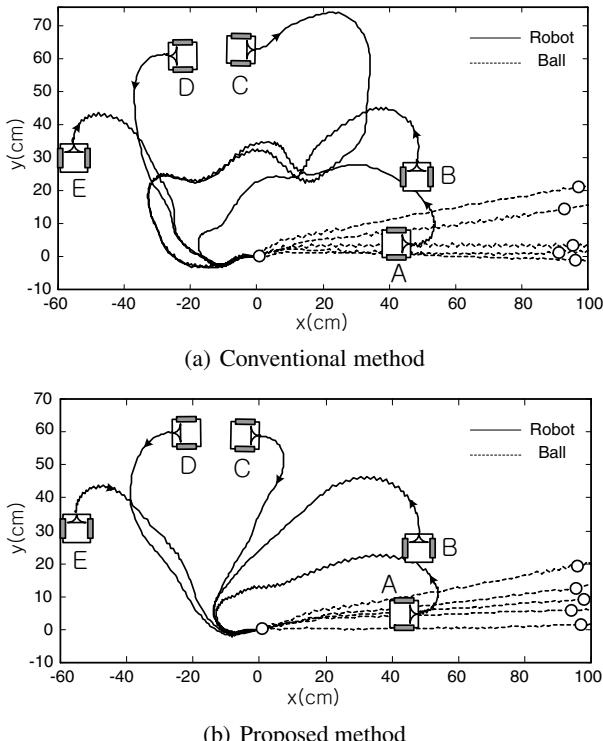
In order to validate the applicability of the proposed rule-scoring method, the mobile robot was required to show an improvement in shooting ability. Kicking was tested with 27 randomly selected rule sets generated from both conventional and proposed methods. The robot was initialized from five distinct positions with various facings as shown in Figure 6. To compensate for variation from the noise caused by physical disturbances and errors, the robot was tested with five kicks for each combination of starting point and applied rule set. Subsequently this resulted in 135 kicks for each method at a particular point and 1,350 kicks in total.

The elapsed traveling time (along the path) was used to evaluate the *path effectiveness* and its statistical analysis is shown in Table 3. The results indicate the proposed rule-scoring method consistently generates paths that have shorter elapsed times with significantly reduced variation.

**Table 3.** Elapsed time for shooting in experiments (sec)

Start point	Conventional		Rule-scoring	
	Mean	Std	Mean	Std
A	2.60	0.33	2.33	0.14
B	2.23	0.28	2.00	0.11
C	1.93	0.28	1.71	0.10
D	1.73	0.23	1.60	0.06
E	1.77	0.22	1.59	0.07
Average	2.05	0.27	1.85	0.09

Figure 6 illustrates several shooting solutions generated by both conventional and proposed rule-scoring methods. Figure 6(a) shows several trajectories generated by an applied rule set derived using the conventional method. As discussed earlier, chromosomes could be trapped in a local minimum, when they had relatively high PI including inferior genes (rules). The chromosomes were selected and defective rules were survived, and these often had genes (rules) that were only triggered for the paths on



**Fig. 6.** Experiment of robot shooting

which they performed poorly. Consequently, evolution of these genes (rules) did not occur and the final rules needed for these poorly evolved points remained inferior. In contrast, chromosomes holding genes (rules) which might better evolve solutions from these points might undergo bad fitness (relatively low PI) from the other points, and subsequently died out. This is clearly seen in the diagram where paths generated for **D** and **E** provide successful solutions, however the remaining paths for **A**, **B** and **C** deviate undesirably. Alternatively, the proposed rule-scoring method discriminates among genes using the strength of the rules in the evolutionary process. This helps to eliminate defective rules and allows for evolution of superior rules more evenly spread across the entire input space. This is perfectly illustrated in figure 6(b).

## 5 Conclusion

In this paper, a new selection scheme was proposed which evaluated chromosomes on the gene level. The method scores rules on the same column of the chromosomes for selection and rearranges the rules for reproducing parents, on the contrary of conventional EA working on the chromosome level. The simulation and experiment were performed for path planning for shooting in robot soccer. The results of evolutions showed that

the proposed method performs better, with fast convergence and smaller standard deviation. It also maintains consistency that suggests evolved solutions are more reliable. The experiments also showed that the robot following paths evolved by the proposed method ran faster with lower variation of performance.

## Acknowledgement

This research is supported by the Ubiquitous Computing and Network (UCN) Project, the Ministry of Information and Communication (MIC) 21st Century Frontier R&D Program in Korea.

## References

1. T. J. Procyk and E. H. Mamdani, "A linguistic self-organizing process controller," *Automatica*, Vol.15, pp.15-30, 1979.
2. H.-P. Schwefel, *Numerical Optimization of Computer Models*. Chichester, UK:John Wiley, 1981.
3. D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley,1989.
4. C. C. Lee, "Fuzzy logic in control system: fuzzy logic controller-part I and II," *IEEE Trans. Syst. Man Cybern.*, Vol.20, 1990.
5. M.Mizumoto, "Fuzzy controls by fuzzy sington-type resoning method," *Proc. of the Fifth IFSA world congress*, Seoul, Korea, pp.945-948, 1993.
6. T.Bäck and H.-P.Schwefel, "An overview of evolutionary algorithms for parameter optimization," *Evol. Comput.*, Vol.1, pp.1-23, 1993.
7. J.-H. Kim, Ed., *Robotics and Autonomous Systems*, Special Issue: First Micro-Robot World Cup Soccer Tournament, MiroSot, vol. 21, no. 2, Sept. 1997.
8. Y.-J. Kim, J.-H. Kim and D.-S. Kim, "Evolutionary Programming-Based Uni-vector Field Navigation Method for Fast Mobile Robots", *IEEE Trans. on Systems Man and Cybernetics - Part B - Cybernetics*, vol. 31, no. 3, pp. 450-458, Jun. 2001.
9. J.-M. Yang and J-H. Kim, "Sliding Mode Control for Trajectory Tracking of Nonholonomic Wheeled Mobile Robots," *IEEE Transactions on Robotics and Automation*, Vol.15, No.3, Jun. 1999.
10. S.-J. Kang, C.-H. Woo, H.-S. Hwang, K.B. Woo, "Evolutionary design of fuzzy rule base for nonlinear system modeling and 5control," *IEEE Transactions on Fuzzy Systems*, Vol.8, No.1, pp37 - 45, 2000.
11. M.-S. Lee, M.-J. Jung, J.-H. Kim, "Evolutionary programming-based fuzzy logic path planner and follower for mobile robots," *Proceedings of the 2000 Congress on Evolutionary Computation*, Vol.1, pp.139 - 144., 2000.
12. W. Pedrycz and M. Reformat, "Evolutionary Fuzzy Modeling," *IEEE Trans. Fuzzy Syst.*, Vol.11, No.5, pp.652-665, Oct. 2003.
13. J.-H. Kim, D.-H. Kim, Y.-J. Kim, K.-T. Seow, *Soccer Robotics* (Springer Tracts in Advanced Robotics), Springer Verlag, 3540218599, 326, 326, Sep. 2004.

# An Improved Genetic-Based Particle Swarm Optimization for No-Idle Permutation Flow Shops with Fuzzy Processing Time

Qun Niu and Xingsheng Gu<sup>\*</sup>

Research Institution of Automation, East China University of Science & Technology  
200237, Shanghai, China  
aromacc@tom.com

**Abstract.** Due to the uncertainty of the processing time in the practical production, no idle flow shop scheduling problem with fuzzy processing time is introduced. The objective is to find a sequence that minimizes the mean makespan and the spread of the makespan by using a method for ranking fuzzy numbers. The particle swarm optimization (PSO) is a population-based optimization technique that has been applied to a wide range of problems, but there is little reported in respect of application to scheduling problems because of its unsuitability for them. In the paper, PSO is redefined and modified by introducing genetic operations such as crossover and mutation to update the particles, which is called GPSO and successfully employed to solve the formulated problem. Several benchmarks with fuzzy processing time are used to test GPSO. Through the comparative simulation results with genetic algorithm, the feasibility and effectiveness of the proposed method are demonstrated.

## 1 Introduction

In the practical production, it is sometimes required that the resources of a production system perform the operations consecutively. In fact, due to some technical or economical circumstances, it is necessary to set machine running continuously. The no-idle condition is typically imposed when the machine idle time is extremely costly. For example, the furnace in the fiber glass processing, in which glass batches are reduced to molten, stays on during the entire production season as it takes three days to heat it back to the required temperature of 2800 ° F. An important class of these scheduling problems is constrained by the no-idle production environment, where machines work continuously without any interruption from the start of the first job processing to the last job completion. In the paper, the no idle flow shop scheduling is studied.

The no-idle schedule has been first discussed by Adiri and Pohoryles [1]. Garey, Johnson, and Sethi [2] proved the NP-hardness of the  $F3/no-idle/Cmax$ . Baptiste

---

\* Supported by The National Natural Science Foundation of China (No.60474043) and The Key Technologies Program of Shanghai Municipal Science and Technology Commission (No. 04dz11008).

and Hguny[3] proposed a proof strongly similar to this one for the NP-hardness of the *F3/no-idle/Cmax*: The complexity of the *Fm/no-idle/Cmax* has only been mentioned by Tanary, Sotskov, and Strusevitch[4]. In Giaro [5], the complexity of some specified flow-shop or open-shop scheduling problems with no-idle or no-wait constraints were studied. Narasimhan [6], and Panwalkar have studied a manufacturing environment, which could be found in many industries: the hybrid flow-shop where the production system has continuous-process machinery on the first stage and a repetitive-batch equipment on the second one.

Most researchers who solve the scheduling problem assume the time parameters such as processing time and due dates are fixed values. However, in many real world applications, the processing times of jobs could be vague and imprecise due to incomplete information or uncertain environment. For example, the concept of processing time includes setup time and traveling between machines which will not be exactly the same from day to day. The earliest paper in fuzzy scheduling appeared in 1979 [7].

A few approaches to scheduling problems with fuzzy processing time have been developed and reported in literature. The first significant application that considers the uncertainty in time parameters is the one of Fortemps [8]. Majority of approaches consider fuzzy processing time and fuzzy due dates [9, 10]. Chanas considered minimization of maximum lateness of jobs in a single machine scheduling problem [11]. Litoiu and Tadei[12] presented some new models for real-time task scheduling with fuzzy deadlines and processing times. Hong[13]used triangular membership functions for flexible flow shops with two machine centers to examine processing-time uncertainties.

A limited amount of literature has been devoted to the no idle flow shop scheduling problems with fuzzy processing time(FNIFSP). In this paper, as a practical application, we focus on the FNIFSP with fuzzy processing times. Minimizing fuzzy makespan is frequently used in the production where time is costly, while the application is within a just-in-time environment, minimizing the uncertainty of the makespan is also important. In the paper, we adopt and modify a method of ranking fuzzy numbers to evaluate the objectives for FNIFSP with triangular fuzzy processing time. In order to solving the problem more effectively, Genetic-Based Particle Swarm Optimization(GPSO) is proposed, which is redefined and improved by utilizing the genetic operations to update the particles. In particularly, GPSO possesses better ability of finding the global optimum compared with GA algorithms.

The paper is organized as follows. Fuzzy no idle flow shop scheduling problem statement is reported in Section 2. Section 3 introduces related fuzzy sets arithmetic and the method for ranking fuzzy numbers is presented and modified to solve our problem. It is followed by a presentation of GPSO that will be used for FNIFSP in section 4. A lot of simulation results to illustrate how GPSO can perform better for the FNIFSP compared with genetic algorithm are shown in Section 5. Conclusions are given in sections 6.

## 2 No Idle Flow Shop Scheduling with Fuzzy Processing Time

The no idle flow shop problem can be formulated as follows.

There are a set of jobs  $J = \{1, 2, \dots, n\}$  and a set of machines  $M = \{1, 2, \dots, m\}$ . Job  $j$ ,  $j \in J$ , consists of a sequence of  $m$  operations  $O_{j1}, O_{j2}, \dots, O_{jm}$ ; Operation  $O_{ik}$

corresponds to the processing of job  $j$  on machine  $k$  during an uninterrupted processing time  $\tilde{P}_{i_k k}$  which is the fuzzy processing time.  $\tilde{I}_{i_k j}$  and  $\tilde{C}_{i_k j}$  denote the start time and the completed time respectively. Each of  $n$  jobs from the set  $J$  has to be processed on  $m$  machines in the order given by indexing of the machines. Each machine can process at most one operation at a time, and each job can have at most one of its operations undergoing processing at any point in time. In the paper, the objective function is to minimize the mean makespan and the spread of the makespan.

(1) When  $i_k = 1, j = 1$ ,

$$\tilde{I}_{i_k j} = 0, \quad \tilde{C}_{i_k j} = \tilde{P}_{i_k j} \quad (1)$$

(2) If  $i_k > 1, j = 1$ ,

$$\begin{aligned} \tilde{I}_{i_k j} &= \tilde{C}_{i_k (j-1)} \\ \tilde{C}_{i_k j} &= \tilde{I}_{i_k j} + \tilde{P}_{i_k j} = \tilde{C}_{i_k (j-1)} + \tilde{P}_{i_k j} \end{aligned} \quad (2)$$

(3) If  $i_k > 1, j > 1$ ,

$$\begin{aligned} \tilde{I}_{i_k j} &= \max(\tilde{C}_{i_k (j-1)}, \tilde{C}_{(i_k - 1) j - 1}) \\ \tilde{C}_{i_k j} &= \tilde{I}_{i_k j} + \tilde{P}_{i_k j} \end{aligned} \quad (3)$$

Subject to:

$$\begin{aligned} \tilde{I}_{i_k j} &= \tilde{C}_{(i_k - 1) j} \\ \tilde{C}_{i_k j} &= \tilde{I}_{(i_k + 1) j} \\ \tilde{C}_{i_k j} &\geq \tilde{C}_{(i_k - 1) j} + \tilde{P}_{i_k j} \\ \tilde{C}_{i_k j} &\geq \tilde{C}_{i_k (j-1)} + \tilde{P}_{i_k j} \\ \tilde{C}_{i_k j} &\geq \tilde{C}_{(i_k - 3)(j+1)} + \tilde{P}_{i_k j} \end{aligned} \quad (4)$$

$$\text{Objective function: } F = \min(C_{i_k, j} + \beta \text{spread}) \quad (\beta \in [0, 1]) \quad (5)$$

### 3 Related Fuzzy Set Operations

#### 3.1 Fuzzy Addition and Fuzzy Maximum

Fuzzy Set Theory was formalized by Professor Lofti Zadeh in 1965[14], and its successful application has ensured its adoption around the world.

Denote the triangular fuzzy number  $\tilde{M}$  by a triplet  $(a, b, c)$  and  $\tilde{N}$  by a triplet  $(d, e, f)$ . The addition of the two triangular fuzzy numbers  $\tilde{M}$  and  $\tilde{N}$  is shown by the

following formula, which return the sum of the fuzzy numbers represented by the two lists.

$$\tilde{M} + \tilde{N} = (a, b, c) + (d, e, f) = (a+d, b+e, c+f) \quad (6)$$

The maximum ( $\vee$ ) of two triangular fuzzy numbers is approximated by a triangular fuzzy number using the following equation:

$$(a, b, c) \vee (d, e, f) \equiv (a \vee d, b \vee e, c \vee f) \quad (7)$$

We can use this approximation because we work with the fuzzy interval rather than the membership function. Also, degree of the approximation depends on the overlap of the two triangular fuzzy numbers.

### 3.2 The Method of Ranking the Fuzzy Numbers for FNIFSP

Ranking two numbers is quite important in the scheduling. Ranking is easily performed since the operands are all crisp numbers. In the paper, fuzzy ranking methods have to be applied since each operand is a fuzzy set. Several fuzzy ranking methods have been proposed in the literature, such as Yager [15](1981), Dubois[16] and Prade [17](1983), Chen[18](1985), Lee and Li [19](1988).

As the makespan is not the only objective and the uncertainty of the makespan is also considered in the real production, Lee and Li fuzzy ranking method is adopted and modified to solve the problem [19]. The mean and the variance are used to replace the fuzzy makespan and the spread of the makespan. In the case of the uniform distribution, the mean of a fuzzy event A with membership function  $\mu_A$  is defined as and the variance of A is defined as

$$M_u(\tilde{A}) = \frac{\int_A x \mu_A(x) dx}{\int_A \mu_A(x) dx}, \quad \sigma_u^2(\tilde{A}) = \frac{\int_A x^2 \mu_A(x) dx}{\int_A \mu_A(x) dx} - M_u^2(\tilde{A}). \quad (8)$$

In the case of the proportional distribution, the mean and variance of A are, respectively, given by the integrals

$$M_u(\tilde{A}) = \frac{\int_A x \mu_A^2(x) dx}{\int_A \mu_A^2(x) dx}, \quad \sigma_u^2(\tilde{A}) = \frac{\int_A x^2 \mu_A^2(x) dx}{\int_A \mu_A^2(x) dx} - M_u^2(\tilde{A}) \quad (9)$$

For the normal triangular fuzzy numbers  $\tilde{M} = (a, b, c)$ , the mean and the variance can be simplified to the follows

$$M_u(\tilde{A}) = \frac{a+b+c}{3}, \quad \sigma_u^2(\tilde{A}) = \frac{a^2+b^2+c^2-ab-ac-bc}{18} \quad (10)$$

$$M_u(\tilde{A}) = \frac{a+b+2c}{4}, \quad \sigma_u^2(\tilde{A}) = \frac{(a-b)^2 + 2(a-c)^2 + 2(b-c)^2}{80} \quad (11)$$

The  $\sigma(\tilde{A})$  can be denoted as the spread of the fuzzy makespan which shows the uncertainty of the makespan. If  $\sigma(\tilde{A})$  turns more larger, the fuzzy makespan is more uncertain. We wanted the mean makespan is minimal while the uncertainty of the makespan is also minimal. So the ranking index is given as follows:

$$F(\tilde{A}) = m(\tilde{A}) + \omega\sigma(\tilde{A}), \omega \in [0,1]. \quad (12)$$

## 4 PSO for Flow Shop with Fuzzy Processing Time

### 4.1 Proposed GPSO for FNIFSP

PSO is invented and developed by Eberhart and Kennedy [20], which was inspired by the social behaviour of animals such as bird flocking. PSO has been used to solve a variety of difficult optimization problems, including engineering applications and function minimization [21,22]. At present, most existing PSO applications are resorting to continuous function value optimization, but only few studies considered discrete optimization problems especially TSP or scheduling problems. Clerc [23] adopted PSO to solve the well known Traveling Salesman Problem.

We describe the formulations of a GPSO algorithm for FNIFSP.

- 1) Problem definition: Each feasible solution of FNIFSP is a point in the search space and may be regarded as a bird. It is not same as continuous optimization problems, the solution is not a certain value but a sequence such as [3 2 5 1 4]. So, the original PSO is not suitable, and the updating method should be redefined to fit the problems.
- 2) Updating of the particles: Each particle keeps track of its coordinates in the problem space, which are associated with the best solution (fitness) it has achieved so far. This value is called pbest. Another “best” value that is tracked by the global version of the particle swarm optimization is the overall best value, and its location obtained so far by any particle in the population. This location is called gbest. The purpose of updating the particle is to obtain a new feasible solution  $X_{i+1}^k$ . In our GPSO, the new individuals will be generated by the pbest, gbest and the current individual, which is formulated as follows:

$$X_{i+1}^k = (P_i^k - X_i^k) + (P_g^k - X_i^k) + \overline{X}_i^k \quad (13)$$

As  $P_i^k$ ,  $P_g^k$  and  $X_i^k$  is not a value, the subtraction operation and addition operation is not the same as solving continuous optimization problems, and need to be redefined using new methods. So, genetic operations such as crossover and mutation are introduced to update the particle in (13).

Various operations performed for updating particle positions are explained below:

The symbol “—” of subtraction operator: The subtracting represents the crossover operation of the two individuals.

The symbol “+” of addition operator: the addition operation means the optimal solution is selected from the offspring of  $P_i^k - X_i^k$ ,  $P_g^k - X_i^k$ , and  $\bar{X}_i^k$ .

For example, subtracting two individuals i.e. (Pbest – Pcurrent) results in the crossover over.  $\bar{X}_i^k$  means the mutation operation of  $X_i^k$ .

The process of generating new particles is illustrated in Fig.1.  $P_i^k - X_i^k$  means  $P_i^k$  and  $X_i^k$  is taken as two parents and generate the offspring by crossover operation. The two point crossover method is adopted in our GPSO. The two-point crossover sets two crossover points at random, and takes a section between the points from one parent and other sections outside the points from the other parent and recombines them.  $P_g^k - X_i^k$  is the same as  $P_i^k - X_i^k$ .  $\bar{X}_i^k$  is mutation offspring of  $X_i^k$ . The new particle  $X_{i+1}^k$  is the best individual of the offspring generated by  $P_i^k - X_i^k$ ,  $P_i^k - X_i^k$  and  $\bar{X}_i^k$ .

$$\begin{aligned} P_i^k - X_i^k &\Rightarrow \begin{cases} \text{parent1 } P_i^k \\ \text{parent2 } X_i^k \end{cases} \Rightarrow \text{crossover} \Rightarrow \text{offspring} \\ P_g^k - X_i^k &\Rightarrow \begin{cases} \text{parent1 } P_g^k \\ \text{parent2 } X_i^k \end{cases} \Rightarrow \text{crossover} \Rightarrow \text{offspring} \\ X_i^k &\Rightarrow \text{mutation} \Rightarrow \bar{X}_i^k \end{aligned}$$

**Fig. 1.** The generation of new particles

## 4.2 Implement of GPSO for FNIFSP

1) *Representation:* The first task is to represent a solution of a problem in solving JSSP. The particle is represented by a feasible solution. Integer coding is used to solve no idle Flow shop scheduling. We assume that 6 jobs will be processed, which numbered as [1 2 3 4 5 6]. A random sequence is be yielded such as [2 6 1 3 4 5], and in terms of the order the jobs will be processed in turn. The purpose is to insure that all the individuals generated randomly are feasible.

2) *Initialization:* The PSO starts with a randomly generated set of particles called the initial population. The size of the population depends on the solution space.

3) *Updating the particles:* the new particle is generated from.

4) Crossover operator and mutation operator are used in the update, which are described as following:

I ) *Crossover:* The two-point crossover is used, which sets two crossover points at random. After two offspring are evaluated, the better one is used as the offspring. Examples of crossover are given in Fig. 2, where index denotes the operation number of each job.

II ) *Mutation:* The mutation followed in this paper is the order-based mutation. Two genes are picked in the particle at random and exchange their genes. Suppose the particle is [4 1 3 6 2 5], after performing mutation the particle is [4 1 5 6 2 3].

The new particle  $X_{i+1}^k$  can be obtained from the offspring of  $P_i^k - X_i^k$ ,  $P_g^k - X_i^k$ , and  $\bar{X}_i^k$ .

<i>parent1</i>	4	<u>2</u>	6	3	1	5	
<i>parent2</i>	5	<u>1</u>	3	2	4	6	
<i>crossover offspring 1</i>		4	1	3	2	5	6
<i>crossover offspring 2</i>		5	2	6	3	1	4

**Fig. 2.** Example of crossover operation

## 5 Computational Results

To test the effectiveness of our approach, some instances are needed. A method [24] is used to fuzzify some of the famous crisp benchmarks. For each crisp duration  $x$ , a three-point triangular fuzzy number is built. The first point is drawn randomly from the interval  $[\delta_1 x, x]$ , where  $\delta_1 < 1$ . The center point is set equal to  $x$ , and the third point is drawn randomly from the interval  $[x, \delta_2 x]$ , where  $\delta_2 > 1$ . The famous benchmarks of ta001, ta003, ta031, ta033, rec1, rec3, and car1 are selected to fuzzify the processing time and to investigate the performance of GPSO.

To measure the effectiveness and viability of GPSO, results are compared with GA. To perform the comparison experiment fairly, the population size and the generation are set to 20 and 1000. We coded in Java and run on Pentium 2.0 G and every scale needed to be run ten times. The probability of crossover is 0.8 and the probability of mutation is 0.15. The comparison result is listed in Table 1. In Table 1, BFV denotes the best value of the ten times' run; WVF means the worst value of the ten times' run; AFV denotes the average value of the best value of the ten times' run; Time is the CPU time of every run and records by second; w denotes the weight of the makspan spread in the objective function.; For every w, the results of the upper row show the objective values of uniform distribution, and the results of the lower row show the objective values of proportional distribution.

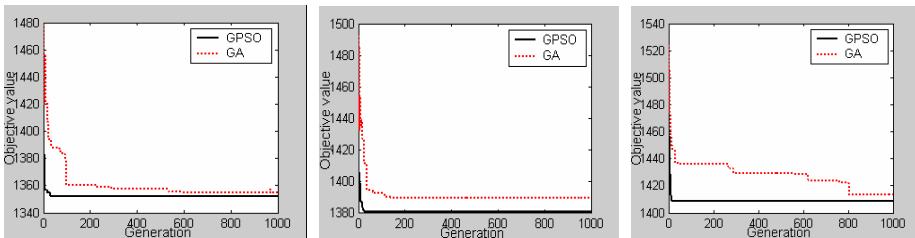
As we can see from Table 1, if we increase the weight of the spread in the objective function, we get higher values of the makespan. As we give more weight to the spread in the objective function, the spread decreases at the cost of the fuzzy makespan. The application environment may be the factor that decides how much weight to give to the spread in the objective function. Again, if the application is within an environment where time is costly, we may prefer to give less weight to the spread. On the hand, if the application is within a just-in-time environment, we may prefer to give more weight to the spread.

From Table 1, we can see that the results of GPSO are superior to GA in term of searching quality and derivation of the results. If the size of the problem is small such as the instance of Car1, the results almost have no difference. But when the size of the problem is large, the results of GPSO are much better than GA.

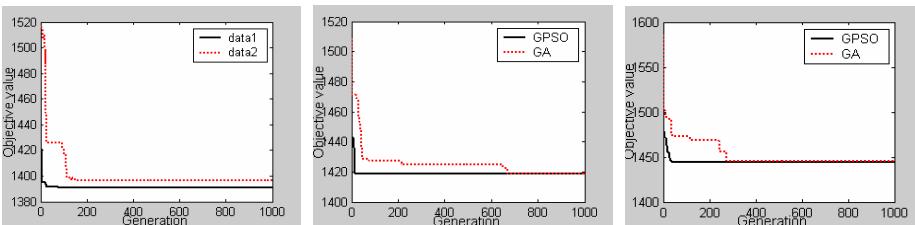
From Fig. 3 and Fig. 4, it can be seen that the varying curves of objective values using GPSO descend much faster than using GA, and the objective values descent to lower level by using GPSO than by GA. So, it is concluded that GPSO is more efficient than GA, and it is also concluded that the final searching quality of GPSO performs much better than GA.

**Table 1.** The comparison results of GPSO and GA

<i>prob- lem</i>	<i>w</i>	GPSO				GA			
		BFV	WFV	AFV	Time	BFV	WFV	AFV	Time
Ta001	0	1351.7	1351.8	1351.8	3.08	1354.8	1373.4	1357.2	3.20
		1390.9	1391.8	1391.3	3.10	1396.8	1401.9	1397.9	3.21
	0.5	1380.2	1380.2	1380.2	3.13	1389.1	1390.9	1389.4	3.23
		1417.0	1418.7	1417.5	3.11	1418.7	1421.0	1420.9	3.24
	1	1408.6	1408.6	1408.6	3.14	1413.7	1420.3	1417.5	3.22
		1443.2	1444.2	1443.7	3.13	1445.7	1466.5	1457.2	3.26
	0	1237.6	1239.7	1238.3	3.12	1270.0	1274.2	1271.5	3.24
		1281.4	1284.8	1282.5	3.17	1288.1	1318.3	1312.6	3.26
Ta003	0.5	1271.4	1272.4	1271.8	3.14	1279.9	1311.0	1296.9	3.27
		1311.2	1311.0	1311.1	3.20	1344.8	1355.4	1347.3	3.29
	1	1300.4	1303.1	1301.5	3.15	1311.9	1337.5	1326.8	3.25
		1340.6	1343.5	1342.7	3.18	1349.5	1373.5	1356.1	3.31
	0	2894.7	2900.6	2897.2	8.85	2913.1	2921.7	2919.5	9.07
		2984.6	2989.8	2986.3	8.82	3003.6	3018.1	3015.4	9.04
	0.5	2958.3	2959.4	2958.7	8.80	2967.8	2989.8	2975.3	9.05
		3045.0	3045.0	3045.0	8.82	3060.5	3063.7	3060.9	9.05
Ta031	1	3021.9	3026.6	3024.6	8.83	3035.7	3050.8	3042.1	9.04
		3105.3	3108.7	3106.8	8.89	3119.2	3136.8	3127.6	9.07
	0	2916.6	2918.3	2917.2	8.45	2941.8	2976.0	2968.4	8.62
		3030.8	3032.3	3031.4	8.43	3042.3	3092.6	3068.9	8.61
	0.5	3002.6	3003.7	3002.9	8.49	3049.3	3065.7	3054.8	8.64
		3110.8	3111.2	3110.9	8.45	3169.8	3185.1	3177.9	8.67
	1	3085.3	3087.9	3086.1	8.47	3136.4	3152.7	3146.4	8.69
		3190.0	3190.0	3190.0	8.49	3223.6	3223.6	3223.6	8.68
Ta033	0	7449.1	7449.1	7449.1	1.47	7449.1	7449.1	7449.1	1.58
		7691.7	7691.7	7691.7	1.43	7691.7	7691.7	7691.7	1.53
	0.5	7620.9	7620.9	7620.9	1.42	7620.9	7620.9	7620.9	1.54
		7854.6	7854.6	7854.6	1.45	7854.6	7854.6	7854.6	1.51
	1	7792.8	7792.8	7792.8	2.46	7792.8	7792.8	7792.8	1.55
		8017.5	8017.5	8017.5	1.47	8017.5	8017.5	8017.5	1.56
	0	1324.0	1329.9	1326.3	2.87	1342.1	1361.4	1355.8	3.04
		1368.6	1374.8	1370.6	2.84	1384.7	1388.1	1386.9	3.01
Car1	0.5	1351.7	1359.5	1354.3	2.85	1381.8	1389.2	1386.7	3.01
		1405.1	1406.1	1405.4	2.81	1410.5	1419.4	1415.7	3.03
	1	1393.8	1394.9	1394.5	2.89	1399.3	1417.3	1410.1	3.02
		1419.5	1425.9	1421.7	2.88	1439.4	1455.8	1448.3	3.07
	0	1215.6	1215.6	1215.6	2.86	1220.8	1234.3	1231.2	3.03
		1263.6	1263.6	1263.6	2.91	1280.9	1285.6	1282.7	3.08
	0.5	1251.3	1251.3	1251.3	2.94	1265.7	1273.0	1269.5	3.06
		1296.9	1296.9	1296.9	2.90	1309.3	1314.8	1310.3	3.06
Rec1	1	1286.9	1286.9	1286.9	2.92	1295.6	1298.8	1294.8	3.04
		1263.6	1263.6	1263.6	2.93	1264.1	1278.2	1275.9	3.05



**Fig. 3.** Convergence comparison of the algorithms for ta001 of uniform distribution



**Fig. 4.** Convergence comparison of the algorithms for ta001 of proportional distribution

## 6 Conclusion

In the paper, we investigated the FNIFSP with fuzzy processing times. The method of ranking fuzzy numbers is used to estimate the value of the fuzzy makespan and the spread of the makespan. It is worth mentioning that as we introduced the spread to the objective function, more computation was required to obtain an optimal solution. The application environment may be the factor that decides how much weight to give to the spread in the objective function. If the application is within an environment where time is costly, less weight to the spread is given. If the application is within a just-in-time environment, we prefer to give more weight to the spread. Finally, the better estimate we can have for the processing times, the less uncertainty we get for the fuzzy makespan. GPSO is proposed and improved to solve FNIFSP which is suitable to the scheduling problems. Through a lot of simulations, it was demonstrated that, compared with GA, GPSO found better results than GA. Therefore, GPSO can be considered as an efficient algorithm to FNIFSP.

## References

1. Adiri, I., & Pohoryles, D. Flow-shop/no-idle or no-wait scheduling to minimise the sum of completion times. *Naval Research Quarterly*, 29(1982) 495–504.
2. Garey, M. R., Johnson, D. S., & Sethi, R. The complexity of flow-shop and job-shop scheduling. *Mathematics of Operational Research*, 1(1976) 117–129.
3. Baptiste, P., & Hguny, L. K. A branch and bound algorithm for the  $F = \text{no-idle} = C$  max (Vol. 1). *Proceedings of the International Conference on Industrial Engineering and Production Management (IEPM'97)*, Lyon, (1997) pp. 429–438.

4. Tanarv, V. S., Sotskov, Y. N., & Strvsevitch, V. A. *Scheduling theory multi-stage system*. Dordrecht: Kluwer Academic Publishers. (1994)
5. Giaro, K. NP-hardness of compact scheduling in simplified open and flow shops. *European Journal of Operational Research*, 130 (2001) 90–98.
6. Narasimhan, S. L., & Mangiameli, P. M. A comparison of sequencing rules for a two-stage hybrid flow-shop. *DecisionSciences*, 18 (1987) 250–265.
7. Prade H.: Using fuzzy set theory in a scheduling problem: a case study. *Fuzzy Sets and Systems*. 2 (1979) 153–165.
8. Slowinski R., Hapke M.: *Scheduling Under Fuzziness*. Heidelberg. (2000)
9. Fortemps P.: Jobshop scheduling with imprecise durations: a fuzzy approach. *IEEE Trans. Fuzzy Syst.* 5 (1997) 557–569.
10. McCahon S., Lee E. S.: Job sequencing with fuzzy processing times. *Computers and Mathematics with Applications*. 19 (1990) 31-41.
11. Chanas S., Kasperski A.: Minimizing Maximum Lateness in a Single Machine Scheduling Problem with Fuzzy Processing times and Fuzzy Due Dates. *Engineering Applications of Artificial Intelligence*. 14 (2001) 377-386.
12. Litoiu M., Tadei R.: Real-time task scheduling with fuzzy deadlines and processing times. *Fuzzy Sets and Systems*. 117 (2001) 35–45.
13. Hong Tzungpei, Wang Tzutin.: Fuzzy flexible flow shops at two machine centers for continuous fuzzy domains. *Information Sciences*. 129 (2000) 227-237
14. Zadeh, L.A. Fuzzy sets. *Information and Control*. 8 (1965) 338-353.
15. Yager, R.R. A procedure for ordering fuzzy subsets of the unit interval, *Information Sciences* 24 (1981) 143-161.
16. Dubois, D. and Prade, H. Ranking fuzzy numbers in the setting of possibility theory, *Information Sciences* 30 (1983) 183-224.
17. Chen, S.H. Ranking fuzzy numbers with maximizing set and minimizing set, *Fuzzy Sets and Systems* 17 (1985) 113-129.
18. Lee E.S., Li R.J.: Comparison of fuzzy numbers based on the probability measure of fuzzy events. *Computers & Mathematics with Applications*. 15 (1988) 887-896.
19. Lee E.S., Li R.J.: Comparison of fuzzy numbers based on the probability measure of fuzzy events. *Computers & Mathematics with Applications*. 15 (1988) 887-896.
20. Eberhart R. C., Kennedy J.: A new optimizer using particle swarm theory. In Proc. 6th Int. Symp. Micro Machine and Human Science, Nagoya, Japan. (1995) 39-43.
21. Laskari E. C., Parsopoulos K. E., Vrahatis M. N.: Particle swarm optimization for minimax problems. In Proc. IEEE 2002 Congr. Evolutionary Computation Honolulu. (2002) 1582-1587.
22. Mostaghim S., Teich J.: Strategies for finding good local guides in multi-objective particle swarm optimization (MOPSO). In Proc. IEEE 2003 Swarm Intelligence Symp. (2002) 26-33.
23. Clerc M.: Discrete particle swarm optimization, illustrated by the Traveling Salesman Problem. *New Optimization Techniques in Engineering*. Springer, Heidelberg Germany. pp. (2004) 219-239.
24. Omar A. G.: A bi-criteria optimization: minimizing the integral value and spread of the fuzzy makespan of job shop scheduling problems. *Applied Soft Computing*. (2003) 197-210.

# Determinants of E-CRM in Influencing Customer Satisfaction

Yan Liu, Chang-Feng Zhou, and Ying-Wu Chen

School of Information System and Management,  
National University of Defense Technology,  
Changsha, 410073, Hunan, China

{yanliu, cfzhou, ywchen}@nudt.edu.cn

**Abstract.** The widespread use of the web technology presents an opportunity for business to use the Internet as a tool for electronic customer relationship management (e-CRM). Despite agreement that e-CRM has direct or indirect impact on customer satisfaction, the significance and the determinants of e-CRM in influencing customer satisfaction have not been well researched. This study empirically develops a temporal model explaining the relationship between e-CRM and online customer satisfaction. A theoretical framework that consists of e-CRM initiatives: system quality, information quality and service quality; intrinsic success: responsiveness and efficiency; and objective: customer satisfaction is further expanded. An electronic questionnaire is used to collect sample data. Then this model is tested by means of the statistical analysis method of Structure Equation Model (SEM). These findings should be of great interest to both researchers and practitioners. Discussion and implication are presented in the end.

**Keywords:** E-CRM, customer satisfaction, electronic questionnaire, SEM, PLS.

## 1 Introduction

Customer relationship management (CRM) is a comprehensive business and marketing strategy that integrates technology, process, and all business activities around the customer [1], [2]. It is mostly defined in terms of the acquisition and retention of customers and the resulting profitability [3], [4]. Effective CRM is assumed to lead to bottom-line benefits for the organization [5]. With the rapid growth of electronic business and proliferation of Internet-based services, a new concept is born: e-CRM. It encompasses all the processes needed to acquire, build and maintain customer relationship through e-business operations.

E-CRM [6] chiefly relies on Internet or web-based interaction of companies with their customers. Since the beginning of the commercial use of the web, e-CRM has received increasing attention from both practitioners [7]-[9] and researchers [10]. The main driver for e-CRM adoption seems to be a commonly shared belief that it improves customer loyalty and retention [11] through the enhancement of customer satisfaction. Researchers and practitioners alike are claiming positive effects of

e-CRM on customer satisfaction. However, no empirical evidence has been provided for these claims.

As the widespread use and dependency on e-CRM increase, so does the need to develop a better understanding of e-CRM success. Customer satisfaction is one of the more immediate objectives of e-CRM and as satisfaction is often used as a surrogate of success, a good way to study e-CRM success is to examine the relationship between e-CRM and customer satisfaction. Despite the great interest in elucidating the relationship between e-CRM and customer satisfaction [12], prior research has not shed sufficient light on this relationship. Most previous studies focused on the effects of specific features of e-CRM on satisfaction in isolation. The lack of a comprehensive model with a theoretical underpinning casts doubts about the interpretation of the empirical results of these studies. Therefore, the specific research goals are to further develop the e-CRM success model based on empirically evident instruments that (1) develop operationally and empirically a temporal model explaining the effects of e-CRM initiatives on customer satisfaction in the context of online service, (2) identify the scales of these factors, (3) test the relative importance of various factors, and (4) conclude with policy recommendations for promoting the online services in general.

## 2 Theoretical Framework

This paper formulates the model illuminating the effects of e-CRM on customer satisfaction into the causal phases, which comprise e-CRM initiatives, intrinsic success and e-CRM customer satisfaction. These factors are the basis for our research model and hypotheses.

### 2.1 E-CRM Initiatives

Although researchers and practitioners alike are claiming positive effects of e-CRM, we still lack appropriate conceptualization and empirical evidence of these effects. In fact, there is no consistent measurement for e-CRM success [13], making it difficult to manage and assess e-CRM activities. Since e-CRM is an IS application [14], the IS success literature can shed some lights on the conceptualization of e-CRM success. Researchers studying IS success have focused the main determinant success factors of IS on system quality, information quality and service quality.

**System Quality.** If the system has been implemented and adopted successfully, a firm is able to reap its benefits. The potential benefits to a firm are related to the impact dimension of system success. The determining criteria in assessment of system support are the performance characteristics of the systems under study. These concern resource utilization, reliability, response time, ease of terminal use, data accuracy, reliability, completeness, system flexibility, and ease of use. This paper considers favorable system invest, implementation level, integration of e-CRM system with legacy MIS systems, and open networking system for sales force, which will reinforce the relationship between system users and customers.

**Information Quality.** A function of the output value produced by the e-CRM system is perceived from the system users. Making effective use of customer information

resources is the critical issues facing IS executives. This reflects the high value of customer data resources and the importance of managing them effectively. Knowing customers is critical to overall e-CRM success; however, just gathering customer data is not enough. With customer information analytics, these organizations can begin to realize the value from their e-CRM implementation. Customer information analytics is more than just information about the facts. It builds insight into customer and market behaviors, enabling businesses to take the correct action necessary in ever changing market environments.

**Service Quality.** Prior studies have stressed the importance of providing high quality of service [15], [16]. The essence of service quality is the ability to deliver what the customer needs and expects. Service Quality is the customer's subjective assessment that the service they are receiving during the e-CRM is the service they expect. If the service quality of the customer's experience exceeds his expectations, he would be willing to come back and do more business with the vendor. Conversely, customers who experience low service quality will be more inclined to defect to other vendors because they are not getting what they expect. Thus, E-Commerce environments need to consider the 'intended Total Customer Experience': what do the customers want to experience when they interact with their organization and what are their expectations.

## 2.2 Intrinsic E-CRM Success

**Responsiveness.** The responsiveness describes how often an online store voluntarily provides services (e.g. customer inquiries, information retrieval and navigation speed) that are important to its customers. The "responsiveness" measures the ability of a company to provide the appropriate information to customers when a problem occurs, a mechanism for handling returns, and an arrangement for online guarantees. Therefore, customers expect online stores to respond to their inquiries promptly, and customers have identified a fast response as an element of high-quality services.

**Efficiency.** Information system implementation success is frequently defined in terms of the achievement of some predetermined goals, which normally include multiple efficiency parameters such as time, cost, and function [17], [18]. Efficiency is an important and useful measure of performance, which is closely related to, but different from, productivity. Unlike productivity, technical efficiency has been studied less frequently by IS researchers [19]. Efficiency, in this study, is different from the traditional IS success measure in that it is comprehensive internal achievement of a firm's e-CRM process. We measured internal efficiency as one of the intrinsic measures of e-CRM implementation success in terms of perceived improvements such as cost reduction, time saving, and alleviation of e-CRM load.

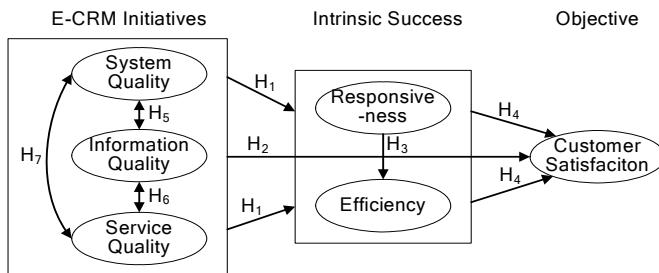
## 2.3 E-CRM Customer Satisfaction

CRM is a customer-driven concept; that is, it allows customers to be in control of the system. Customer satisfaction is the collective outcome of the customer's perception, evaluation, and psychological reaction to the consumption experience with product or service. The proliferation of electronic commerce has blurred the distinction between end-users and online customers, driving the need to account for the mix of

technological and marketing elements that become more salient in the Internet environment. The satisfaction of online customers is not solely determined by product and service attributes (as in the marketing literature), but also shaped by the customer's interaction with the system (as in the IS literature).

Integrating both technological and marketing elements, e-CRM covers all aspects of the customer's online experience throughout the entire transaction cycle. In the context of e-commerce, Sterne [20] proposes a framework to characterize online customer experience, consisting of three stages: pre-sale, sale, and after-sale interactions. Lu [21] uses this framework to study the effects of ecommerce functionality on satisfaction, demonstrating that e-CRM features contribute differently to the satisfaction associated with each transaction stage. Following the same line, Feinberg et al. [22] map the e-CRM features of retail websites into the pre-sale, sale and post-sale stages in investigating the relationship between e-CRM and satisfaction. The usage of the transaction cycle framework to classify satisfaction is also advocated by Khalifa et al. [23], who investigate the relative contribution of pre-sale, sale and post-sale satisfaction to the formation of overall satisfaction.

Based on such theoretical perspectives this paper focuses on the causal relationships among three e-CRM initiatives (system quality, information quality, and service quality), intrinsic e-CRM successes (responsiveness and efficiency), and objective of e-CRM (e-CRM customer satisfaction). We examine the relationships among these constructs and develop the research hypotheses. The model is presented in Fig. 1.



**Fig. 1.** The theoretical framework

This leads to the following hypothesis:

- $H_{1a}$ : System quality affects positively responsiveness.
- $H_{1b}$ : Information quality affects positively responsiveness.
- $H_{1c}$ : Service quality affects positively responsiveness.
- $H_{1d}$ : System quality affects positively efficiency.
- $H_{1e}$ : Information quality affects positively efficiency.
- $H_{1f}$ : Service quality affects positively efficiency.
- $H_{2a}$ : System quality affects positively customer satisfaction.
- $H_{2b}$ : Information quality affects positively customer satisfaction.
- $H_{2c}$ : Service quality affects positively customer satisfaction.
- $H_3$ : Responsiveness affects positively Efficiency.

- $H_{4a}$ : Responsiveness affects positively customer satisfaction.
- $H_{4b}$ : Efficiency affects positively customer satisfaction.
- $H_5$ : System quality and information quality are related to each other.
- $H_6$ : Information quality and service quality are related to each other.
- $H_7$ : System quality and service quality are related to each other.

### 3 Research Methodology

This research design incorporated both qualitative and quantitative approaches. In the first phase, the theoretical literatures from various sources are reviewed to design the research framework. Consequently, the quantitative phase is performed by a survey questionnaire and the obtained data are analyzed through the use of software PLS-graph.

#### 3.1 Survey Study

The research model is validated through an online survey study. Pitkow and Recker [24] present all the advantages of the online surveying method. The target population was shoppers from the customer base of an Internet retailer of computer hardware. Consequently, the data were collected by the use of self-administrated questionnaires. The survey electronic questionnaire consisted of two sections. The first section set out to capture the general profiles of online shoppers. Respondents were asked about their demographic backgrounds, including age, gender, and level of education. Consequently, the participants were asked to identify important e-CRM features that influenced their satisfaction with online shopping. These factors consisted of system quality, information quality, service quality, responsiveness, efficiency and customer satisfaction. A total of 20 questionnaire items were established to measure the extent to which participants gave to the level of agreement or perception toward e-Service provided in the Web portal. A five point Likert scale, ranging from 1 = strongly disagree to 5 = strongly agree, was employed.

#### 3.2 Data Analysis

The model was tested by means of the statistical analysis method of Structure Equation Model (SEM). Although the maximum likelihood (ML) approach is a frequently used estimation procedure for SEM, The proposed SEM in this article adopted the Partial Least Squares (PLS) approach. PLS is a regression-based technique that originates from path analysis [25]; however, it has emerged as a powerful approach to studying causal models involving multiple constructs with multiple indicators. This approach facilitates testing of the measurement model and the structural model simultaneously. The PLS approach is superior to other SEM approaches for this study because of its flexibility for distributional assumptions, its small sample size requirements, and its strength in handling complex predictive models [26]. The computer program used for this analysis was the PLS Graph version 3.0 [27].

## 4 Results and Discussion

### 4.1 The Respondent Profile

A total of 150 questionnaires were distributed and all of them were returned giving an overall response rate of 100 percent. The response rate was substantially high; mainly due to the fact that the questionnaires were collected right after the respondents completed the form. However, three of the returned questionnaires were incomplete and were discarded, producing a total of 147 usable questionnaires (98 percent of the total questionnaires distributed). The demographic profile of the respondents such as age, gender, and level of education in the university are shown in Table 1.

**Table 1.** Characteristics of respondents

	Characteristics	Number	Percentage (%)
Age	≤20	36	20.2
	21-30	61	34.3
	31-40	55	30.9
	≥40	26	14.6
Gender	Male	112	62.9
	Female	66	37.1
Level of Education	High School	48	27.0
Education	Bachelor	62	34.8
	Master	40	22.5
	Higher than Master	28	15.7

### 4.2 Testing the Measurement Model

The testing of the measurement model includes the examination of internal consistency, convergent validity, and discriminant validity. Internal consistency of the constructs is evaluated with composite reliability, as defined by Fornell and Larcker [28]. Nunnally's [29] recommended level of 0.7 for evaluating composite reliability can be used to assess internal consistency. Convergent validity indicates the degree to which multiple items measuring the same construct agree. Convergent validity is adequate when constructs have an Average Variance Extracted (AVE) of at least 0.5. Convergent validity is also demonstrated when items load highly (loading > 0.5) on their associated factors. The results of the tests for internal consistency and convergent validity are presented in Table 2.

From Table 2, the composite reliabilities of all constructs are at least 0.70, except for the factor of service quality (0.67), thus indicating adequate internal consistency. All factor loadings are greater than 0.55 except for one item of service quality. Most of the constructs have an AVE of at least 0.5 except for service quality (0.43). Convergent validity is satisfactory for the constructs in the measurement model.

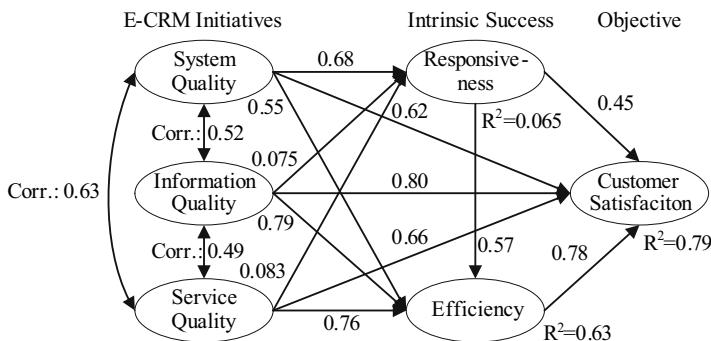
**Table 2.** Assessment of the measurement model

Construct	Mean	Std. Dev	Loading
System quality (Composite Reliability = 0.85 AVE = 0.66)			
SYSQ1	4.05	0.74	0.62
SYSQ 2	3.75	0.75	0.89
SYSQ 3	3.86	0.77	0.88
SYSQ 4	3.95	0.76	0.87
Information quality (Composite Reliability = 0.75 AVE = 0.50)			
INFQ1	3.78	0.75	0.78
INFQ 2	3.94	0.60	0.59
INFQ 3	3.91	0.67	0.73
Service quality (Composite Reliability = 0.67 AVE = 0.43)			
SERQ1	3.38	0.87	0.79
SERQ 2	3.73	0.85	0.74
SERQ 3	3.80	0.76	0.38
Responsiveness (Composite Reliability = 0.91 AVE = 0.77)			
RESP1	4.12	0.73	0.87
RESP 2	4.01	0.76	0.91
RESP 3	3.97	0.72	0.84
Efficiency (Composite Reliability = 0.81 AVE = 0.59)			
EFFI1	3.84	0.71	0.80
EFFI 2	3.90	0.69	0.88
EFFI 3	3.98	0.76	0.60
Customer satisfaction (Composite Reliability = 0.85 AVE = 0.53)			
SATI1	4.14	0.50	0.69
SATI 2	3.87	0.62	0.69
SATI 3	3.94	0.64	0.76
SATI 4	3.90	0.60	0.73

### 4.3 Testing the Structural Model

The test of the structural model includes estimating the path coefficients and the  $R^2$  values. The path coefficients, which indicate the strength and direction of the relationships among the variables, should be significant and directionally consistent with expectations. The  $R^2$ , which represents the proportion of variance in the endogenous variables that can be explained by the antecedents, demonstrates the predictive power of the model. Collectively,  $R^2$  and path coefficients indicate how well the model fits the empirical data. The path coefficients and  $R^2$  values of structural model are presented in Fig. 2.

As hypothesized, system quality has a significant effect on responsiveness (path coefficient = 0.68), so  $H_{1a}$  is supported. The effects of information quality and service quality on responsiveness are insignificant (path coefficient = 0.075, 0.083 respectively), suggesting that  $H_{1b}$  and  $H_{1c}$  are rejected. The  $R^2$  for responsiveness is 0.065, suggesting that other important determinants for responsiveness has been omitted in this model.



**Fig. 2.** Path coefficients and  $R^2$  values of structural model

System quality, information quality, service quality all contribute to efficiency, thus supporting  $H_{1d}$ ,  $H_{1e}$  and  $H_{1f}$ . These factors have path coefficients of 0.55, 0.79 and 0.76 respectively, with a total  $R^2$  of 0.63.

The empirical results show that about 79% of variances in customer satisfaction are explained by system quality, information quality, service quality, responsiveness and efficiency. Hypotheses  $H_{2a}$ ,  $H_{2b}$ ,  $H_{2c}$ ,  $H_{4a}$ , and  $H_{4b}$  are verified. These factors have path coefficients of 0.62, 0.80, 0.66, 0.45 and 0.78 respectively.

Responsiveness has a significant effect on efficiency (path coefficient = 0.57), so  $H_3$  is supported.

The system quality, information quality and service quality are related to each other with the correlations coefficient of 0.52, 0.49 and 0.63 respectively, so  $H_5$ ,  $H_6$  and  $H_7$  are supported.

## 5 Conclusion and Policy Implications

This study investigates determinants of electronic customer relationship management in influencing customer satisfaction. A theoretical framework is developed, through the integration of previous research. The model is then tested empirically through a longitudinal survey study. This exploratory work demonstrates that it is possible to develop preliminary scales for research the relationship between e-CRM and customer satisfaction. Factors identified by this research may be of interest to other researchers and providers interested in the evaluation of e-CRM. The policy implication gives us from this research as follows:

First, investment in e-CRM system, information, and service, in general, is expected to enhance an organization's performance as measured by efficiency. This study substantiates the positive correlation between IT investment and a firm's internal efficiency.

Information quality is the strongest satisfaction factors from any e-CRM implementation. By utilizing it, businesses can take the correct action necessary in ever changing market environments. The real value of information lies in the insight it offers and, ultimately, in the positive, customer-oriented action it triggers. The insight to measure effectiveness, cut costs, reduce churn, understand relationships, anticipate

trends, predict demand, optimize promotions, or segment the market must be used as a catalyst for action.

The result also suggests that efficiency may be an important determinant of customer satisfaction. That is, the result is consistent with a chain of causality from efficiency to customer satisfaction. The causal influence of internal efficiency on external customer satisfaction makes sense conceptually, too. This intriguing interpretation is preliminary and should be subjected to more wide-ranging experimentation.

In future research, other proxies of e-CRM success (e.g., continued usage: repurchase, firm profitability) should be studied. Of particular interest is the mediating role of satisfaction. Customer retention and customer loyalty, for example, have received inadequate attention in MIS and marketing theories. These complex relationships require further exploration. Research could also investigate the relative importance of the factors impacting each stage of the e-CRM process.

## References

1. Anton, J.: *Customer Relationship Management*, Prentice-Hall, New York, NY (1996)
2. Anton, J., Hoeck, M.: *e-Business Customer Service*, The Anton Press, Santa Monica, CA (2002)
3. Menconi, P.: CRM 101: building a great customer relationship management strategy, in Renner, D.H.(Ed.), *Defying the limit: Reaching New Heights in Customer Relationship Management*, Montgomery Research, San Francisco, CA (2000) 31-33
4. Nykamp, M.: *The Customer Differential: The Complete Guide to Implementing Customer Relationship Management*, American Management Association, Chicago, IL (2001)
5. Connelly, P.J. and Yoger, T. "Can CRM win and retain loyal repeat customers?", *Info World*, (2001) 58-59
6. Kundisch, D., Wolfersberger, P., Calaminus, D., Kloepfer, E.: Enabling eCCRM: Content model and management for financial services. *Proceedings of the 34th Hawaii International Conference on System Sciences*, Hawaii (2001)
7. Adams, J.: The hidden opportunity of e-commerce: e-CRM. *Future Banker*, 4 (11) (2000) 41-53
8. Holden, J.: CRM shouldn't hurt. *Sales & Marketing Management*, 153 (2) (2001) 29-30
9. Orr, J.: Strategic options for CRM: Which way off the round about? *Vital Speeches of the Day*, 67 (20) (2001) 615-618
10. Romano, N. C., Fjermestad, J.: Electronic commerce customer relationship management: An assessment of research. *International Journal of Electronic Commerce*, 6 (2) (2001b) 61-113
11. Rosenbaum, H., Huang, B.Y.: A framework for web-based ecommerce customer relationship management. in *Proceedings of the 8th AMCIS Conference*. (2002)
12. McKinney, Yoon, K., Zahedi, F.: The measurement of web-customer satisfaction: an expectation and disconfirmation approach. *Information Systems Research* 13(3) (2002) 296-315
13. Goodhue, D.L. Wixom, B.H., Watson, H.J.: Realizing business benefits through CRM: hitting the right target in the right way. *MIS Quarterly Executive* 1(2) (2002) 79-94
14. Romano, N.C., Fjermestad, J.: Electronic commerce customer relationship management: An assessment of research. *International Journal of Electronic Commerce* 6(2) (2001) 61-113

15. Teas, R.K.: Expectations as a comparison standard in measuring service quality: an assessment of reassessment, *Journal of Marketing* 58 (1994) 132-139
16. Zeuthaml, V.A., Berry, L.B., Parasuraman, A.: The behavior consequences of service quality, *Journal of Marketing* 60 (1996) 31-46
17. Hong, K. K., Kim, Y. G.: The critical success factors for ERP implementation: An organizational fit perspective. *Information and Management* 40(1) (2002) 25-40
18. Markus, M. L., Tanis, C.: The enterprise systems experience from adoption to success. Cincinnati, OH: Pinnaflex Educational Resources Inc. (2000)
19. Shao, B. B. M., Lin, W. T.: Technical efficiency analysis of information technology investments: A two-stage empirical investigation. *Information and Management* 39(5) (2002) 391-401
20. Sterne, J.: Customer service on the Internet. New York: John Wiley and Sons. (1996)
21. Lu, J.: A model for evaluating e-commerce based on cost/benefit and customer satisfaction. *Information Systems Frontiers* 5(3) (2003) 265-277
22. Feinberg, R.A., Kadam, R., Hokama, L., Kim, I.: The state of electronic customer relationship management in retailing. *International Journal of Retail & Distribution Management* 30(10) (2002) 470 - 481
23. Khalifa, M., Limayem, M., Liu, V.: Online consumer stickiness: a longitudinal study. *Journal of Global Information Management* 10(3) (2002) 1-15
24. Pitkow, J.E., Recker, M.M.: Using the web as a survey tool: Results from the second WWW user survey, *Computer Networks & ISDN Systems* 27(6) (1995) 809-822
25. Wold, H., Leitner, H.: Systems analysis by partial least squares. In: Nijkamp P Wrigley N, (editors). *Measuring the Unmeasurable*, Boston, MA: Martinus Nijhoff Publishers (1985) 221-251
26. Chin,W.W., Newsted, P.R.: Structural equation modeling analysis with small samples using partial least squares. In: Hoyle RH, editor. *Statistical Strategies for Small Sample Research*. Thousand Oaks, CA: SAGE Publications (1999) 307-341
27. Chin W.W.: PLS-Graph user' guide, version 3.0. Unpublished manuscript 2001 edition
28. Fornell, C., Larcker, D. F.: Structural equation models with unobservable variables and measurement error. *Journal of Marketing Research* 18(1) (1981) 39-50
29. Nunnally, J.: *Psychometric Theory*, McGraw-Hill, New York. (1978)

# Penalty Guided PSO for Reliability Design Problems

Ta-Cheng Chen

Department of Information Management, National Formosa University  
Yulin, 632, Taiwan  
tchen@nfu.edu.tw

**Abstract.** This paper considers nonlinearly mixed integer reliability design problems in which both the number of redundancy components and the corresponding reliability of each component in each subsystem are to be decided simultaneously so as to maximize the system reliability. The reliability design problems have been studied in the literature for decades, usually using mathematical programming or heuristic/metaheuristic optimization approaches. The difficulties encountered for both methodologies are to maintain feasibility with respect to three nonlinear constraints, namely, cost and weight constraints, and constraints on the products of volume and weight. A penalty-guided particle swarm optimization approach is presented for solving the mixed integer reliability design problems. It can efficiently and effectively search over promising feasible and infeasible regions to find the feasible optimal or near optimal solution. Numerical examples indicate that the proposed approach performs better than other approaches for four reliability-redundant allocation design problems considered in this paper.

**Keywords:** Mixed integer, Particle swarm optimization, Reliability-redundant allocation.

## 1 Introduction

The system reliability optimization is very important in the real world applications and the various kinds of systems have been studied in the literature for decades. Typically there are two ways to achieve higher system reliability: (i) by increasing the reliability of system components, and (ii) by using redundant components in various subsystems in the system. In the first way, the system reliability is able to be enhanced to some degree, but the required reliability enhancement may be never attainable even though the most currently reliable elements are used. Use of the second way is to choose the optimal combination of elements and redundancy-levels; the system reliability can be also enhanced, but the cost, weight, volume, etc. will be increased as well. It perhaps the most common problem in design-for-reliability and has been called as a “*redundancy allocation problem*” [12]. In addition to the above two ways, the combination of the two approaches and reassignment of interchangeable elements are also feasible ways for increasing the system reliability [7] [12]. Such problem of maximizing system reliability through redundancy and component reliability choices is called “*reliability-redundancy allocation problem*” [12]. However, a fundamental trade-off problem between reliability enhanced and resources consumed has to be encountered while a high reliability system is to be designed.

According to the above ways for enhancing system reliability, the reliability design problems include the integer and mixed-integer reliability problems. For the integer reliability problems (redundancy allocation problems), the component redundancy allocation is to be decided while the component reliabilities are given. The major focus of recent work in the redundancy allocation problems is on the development of heuristic/metaheuristic algorithms for redundancy allocation [2] [3] [8]. For the mixed-integer reliability problems (reliability-redundancy allocation problems), the number of redundant components and the corresponding component reliabilities are to be decided simultaneously so as to maximize the system reliability. Only a few approaches were applied to optimize the redundancy level and component reliability contemporaneously in literature. Moreover, those optimization problems are either subjected to the linear constraints [6] [13] [14] or to the nonlinear constraints [9] [7] [16] [18].

In this paper, four mixed-integer reliability problems with multiple nonlinear constraints are considered and solved by using a novel metaheuristic approach. These problems include series, series-parallel, complex (bridge) systems [7] [11] [18] [9] and a gas turbine overspeed protection system [4] [19]. Above problems can be stated as following nonlinearly mixed-integer programming model.

$$\text{Max } R_s = f(\mathbf{r}, \mathbf{n}) \quad (1)$$

$$\text{Subject to } g(\mathbf{r}, \mathbf{n}) \leq \mathbf{b} \quad (2)$$

$$0 \leq r_i \leq 1, n_i \in \text{positive integer}, 1 \leq i \leq m$$

where  $r_i$  and  $n_i$  are the reliability and the number of components in the  $i^{\text{th}}$  subsystem respectively;  $f(\bullet)$  is the objective function for the overall system reliability;  $g(\bullet)$  is the constraint function and  $\mathbf{b}$  is the upper limit on the resource;  $m$  is number of subsystems. The goal is to determine the number of component and the components' reliability in each subsystem to achieve maximum overall system reliability.

Some approaches have been considered to deal with the above problem. For example, mathematic programing relative approaches are applied [7] [15] [18]. However, most of those require derivatives for all nonlinear constraint functions. It makes the exact optimal solutions to this problem hard to be derived easily because of the highly computational complexity. For overcome this difficulty, [9] and [19] applied genetic algorithms (GAs) and [1] applied immune algorithms (IAs) to solve these problems more effectively.

Recently, particle swarm optimization (PSO), which was originally proposed by [10], have been widely studied and applied to a variety of optimization problems. The main concept of PSO is based on the food-searching behavior of birds flocking or fish schooling. When PSO is adopted to solve problems, each particle has its own location and velocity, which determine flying direction and distance respectively. The fitness value is evaluated by the optimization function. Compared with other meta-heuristic evolutionary approaches such as GAs and IAs, the PSO has the following advantages (i) less parameters (ii) easy implementation (iii) fast constringency. The advantages are good for solving the mixed-integer programming problems because a population of particles in PSO can operate simultaneously so that the possibility of paralysis in the whole process can be reduced. In this article a penalty guided PSO is applied to solve the reliability-redundancy allocation problems.

## 2 Particle Swarm Optimization

In PSO, individuals (or particles) work by cooperation and competition between themselves. Each individual adjusts its flying by its own experience with the information from companions. Each individual represents a potential solution to the problem. The particles fly through the search space by updating the position of the  $i^{\text{th}}$  particle at time  $t$  according the following equation [10]:

$$x_i(k+1) = x_i(k) + v_i(k) \cdot \Delta t \quad (3)$$

Where  $k$  is the iterative number, the  $x_i(t)$  and  $v_i(t)$  are vectors representing the current position and velocity respectively, i.e.,  $x_i = (x_{i1}, x_{i2}, \dots, x_{iN})$  and  $v_i = (v_{i1}, v_{i2}, \dots, v_{iN})$  in which  $N$  is the number of decision variables of the problem to be solved. Also,  $\Delta t$  is the time step value. Denote the best location this particle has experienced is depicted as  $p_i = (p_{i1}, p_{i2}, \dots, p_{iN})$ , and the best position of the colony as  $p_g(p_{g1}, p_{g2}, \dots, p_{gN})$  respectively. The velocity updates are governed by the following modified equations [17]:

$$v_{in}(k+1) = w \cdot v_{in}(k) \cdot \Delta t + c_1 \cdot rand_{ln}() \cdot (p_{in} - x_{in}(k)) / \Delta t + c_2 \cdot rand_{2n}() \cdot (p_{gn} - x_{in}(k)) / \Delta t \quad (4)$$

where  $0 \leq w \leq 1$  is an inertia weight determining how much of the particle's previous velocity is preserved,  $c_1$  and  $c_2$  are positive acceleration constants,  $rand_{ln}()$  and  $rand_{2n}()$  are uniform random sequences sampled from  $U(0,1)$ . A unit inertia weight ( $w = 1$ ) and a unit time step ( $\Delta t = 1$ ) are employed in our work. The termination criterion for the iterations is determined according to whether the maximum generation is reached. In our implementation, each string consisting of substring includes the type of component and redundant levels for each subsystem. The details is described in Section 2.1.

### 2.1 Solution Representation Mechanism

The solution representation for PSO can be used in the similar manner to that of genetic algorithms. In our implementation, the particle is represented by a string, each string consisting of a substring for each subsystem. Each subsystem in turn consists of a real number substring representing the level of redundancy (integer) and the corresponding reliability of redundant components (real) type of component. It is illustrated in Fig. 1. For instance, the real number in the subsystem 2 is 4.8718, which represents there are four components with reliability = 0.8718.

### 2.2 Constrained Optimization

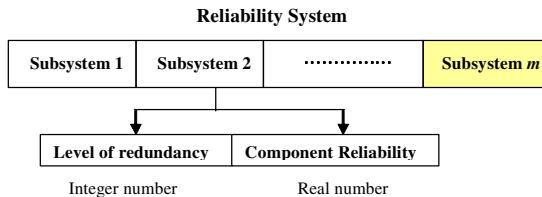
For breeding the superior particles for the next generation (iteration), to evaluate the antibody is necessary step for the PSO. The goal of PSO is to adapt the unfeasible particles to the feasible particles, so as to reduce the constraint violations of the search for obtaining the optimal or near optimal solutions. For handling these constraint violations, a penalty function has to be defined. The penalty function increases the penalty for infeasible solutions based on the distance away from the feasible region. According to the Equation (2) in the problem formulation, for individual  $i$ , the constraint violation value for the  $i^{\text{th}}$  individual is defined as,

$$\text{Penalty}_i = \begin{cases} g(r, n) - b, & \text{if } g(r, n) > b \\ 0, & \text{Otherwise} \end{cases}$$

After defining the penalty function as above, the fitness of each particle to the fitness function can be obtained. The fitness function is described below:

$$\text{Fitness} = \frac{R_s}{1 + \sum_{i=1}^M \text{Penalty}_i}$$

where,  $M$  is the number of constraints in the problem. The above fitness value is to be maximized when the penalty is minimized.



**Fig. 1.** Solution representation

### 3 Numerical Examples and Discussion

To evaluate the performance of the proposed approach for the mixed-integer nonlinear reliability design problems, four test problems ( $P1 \sim P4$ ) are solved. The nonlinear constraints of the first three examples used by [7], [11] and [18] are identical. These examples are the series system, series-parallel system and complex (bridge) system. The fourth example is an overspeed protection system which was investigated by [4] and [19]. All the above problems of maximizing the systems reliability subject to multiple nonlinear constraints can be stated as the mixed-integer nonlinear programming problem. For each problem, both the component reliabilities and redundancy allocations are to be decided simultaneously. The four reliability-redundancy problems are formulated below.

#### Nomenclature

- $m$  the number of subsystems in the system
- $n_i$  the number of components in subsystem  $i$ ,  $1 \leq i \leq m$
- $\mathbf{n} \equiv (n_1, n_2, \dots, n_m)$ , the vector of the redundancy allocation for the system
- $r_i$  the reliability of each component in subsystem  $i$ ,  $1 \leq i \leq m$
- $\mathbf{r} \equiv (r_1, r_2, \dots, r_m)$ , the vector of the component reliabilities for the system
- $q_i = 1 - r_i$ , the failure probability of each component in subsystem,  $1 \leq i \leq m$
- $R_i(n_i) = 1 - q_i^{n_i}$ , the reliability of subsystem  $i$ ,  $1 \leq i \leq m$
- $R_s$  the system reliability
- $g_i$  the  $i^{\text{th}}$  constraint function
- $w_i$  the weight of each component in subsystem  $i$ ,  $1 \leq i \leq m$

- $v_i$  the volume of each component in subsystem  $i$ ,  $1 \leq i \leq m$   
 $c_i$  the cost of each component in subsystem  $i$ ,  $1 \leq i \leq m$   
 $V$  the upper limit on the sum of the subsystems' products of volume & weight  
 $C$  the upper limit on the cost of the system  
 $W$  the upper limit on the weight of the system

**P1. Series system** (Fig. 2(a) [7])

$$\begin{aligned}
 \text{Max } f(\mathbf{r}, \mathbf{n}) &= \prod_{i=1}^m R_i(n_i) \\
 \text{Subject to } g_1(\mathbf{r}, \mathbf{n}) &= \sum_{i=1}^m w_i v_i n_i^2 \leq V, \\
 g_2(\mathbf{r}, \mathbf{n}) &= \sum_{i=1}^m \alpha_i (-1000 / \ln r_i)^{\beta_i} (n_i + \exp(n_i / 4)) \leq C, \\
 g_3(\mathbf{r}, \mathbf{n}) &= \sum_{i=1}^m w_i n_i \exp(n_i / 4) \leq W, \\
 0 \leq r_i &\leq 1, n_i \in \text{positive integer}, 1 \leq i \leq m
 \end{aligned}$$

**P2. Series-parallel system** (Fig. 2(b) [7])

$$\begin{aligned}
 \text{Max } f(\mathbf{r}, \mathbf{n}) &= 1 - (1 - R_1 R_2) (1 - (1 - R_3)(1 - R_4) R_5) \\
 \text{Subject to } g_1(\mathbf{r}, \mathbf{n}), g_2(\mathbf{r}, \mathbf{n}) \text{ and } g_3(\mathbf{r}, \mathbf{n}) \\
 0 \leq r_i &\leq 1, n_i \in \text{positive integer}, 1 \leq i \leq m
 \end{aligned}$$

**P3. Complex (bridge) system** (Fig. 2(c) [7])

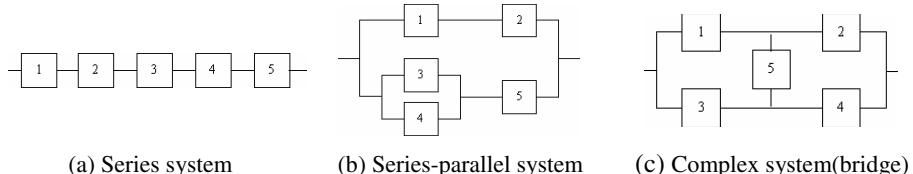
$$\begin{aligned}
 \text{Max } f(\mathbf{r}, \mathbf{n}) &= R_1 R_2 + R_3 R_4 + R_1 R_4 R_5 \\
 &\quad + R_2 R_3 R_5 - R_1 R_2 R_3 R_4 - R_1 R_2 R_3 R_5 - R_1 R_2 R_4 R_5 \\
 &\quad - R_1 R_3 R_4 R_5 - R_2 R_3 R_4 R_5 + 2 R_1 R_2 R_3 R_4 R_5 \\
 \text{Subject to } g_1(\mathbf{r}, \mathbf{n}), g_2(\mathbf{r}, \mathbf{n}) \text{ and } g_3(\mathbf{r}, \mathbf{n}) \\
 0 \leq r_i &\leq 1, n_i \in \text{positive integer}, 1 \leq i \leq m
 \end{aligned}$$

**P4. Overspeed Protection system** [4] [19]

$$\begin{aligned}
 \text{Max } f(\mathbf{r}, \mathbf{n}) &= \prod_{i=1}^m [1 - (1 - r_i)^{n_i}] \\
 \text{Subject to } h_1(\mathbf{r}, \mathbf{n}) &= \sum_{i=1}^m v_i n_i^2 \leq V \\
 h_2(\mathbf{r}, \mathbf{n}) &= \sum_{i=1}^m C(r_i) \cdot [n_i + \exp(n_i / 4)] \leq C \\
 h_3(\mathbf{r}, \mathbf{n}) &= \sum_{i=1}^m w_i n_i \exp(n_i / 4) \leq W \\
 1.0 \leq n_i &\leq 10, \quad n_i \in \text{positive integer}, \\
 0.5 \leq r_i &\leq 1 - 10^{-6}, \quad r_j \in \text{real number},
 \end{aligned}$$

where,  $C(r_i) = \alpha_i [-T/\ln(r_i)]^{\beta_i}$  is the cost of each component with reliability  $r_i$  at subsystem  $i$ ;  $T$  is the operating time during which the component must not fail.

The input parameters defining the specific instances of the four problems have the same values as [4], [7], [9], [11], [18], and [19], and are show in Tables 1, 2 and 3.



**Fig. 2.** Three types of reliability problems

**Table 1.** Data used in series system ( $P1$ ) and complex system ( $P3$ ) [7]

$i$	$10^3\alpha_i$	$\beta_i$	$w_i v_i^2$	$w_i$	$V$	$C$	$W$
1	2.33	1.5	1	7			
2	1.450	1.5	2	8			
3	0.541	1.5	3	8	110	175	200
4	8.050	1.5	4	6			
5	1.950	1.5	2	9			

**Table 2.** Data used in series-parallel system ( $P2$ ) [7]

$i$	$10^3\alpha_i$	$\beta_i$	$w_i v_i^2$	$w_i$	$V$	$C$	$W$
1	2.500	1.5	2	3.5			
2	1.450	1.5	4	4.0			
3	0.541	1.5	5	4.0	180	175	100
4	0.541	1.5	8	3.5			
5	2.100	1.5	4	4.5			

**Table 3.** Data used in overspeed protection system ( $P4$ ) [4] [19]

Stage	$10^3\alpha_j$	$\beta_j$	$v_i$	$w_i$	$V$	$C$	$W$	$T$
1	1	1.5	1	6				
2	2.3	1.5	2	6				
3	0.3	1.5	3	8	250.0	400.0	500.0	1000 hrs.
4	2.3	1.5	2	7				

The proposed penalty guided PSO is implemented in MATLAB® on the Pentium-4 3.2 GHz PC. The determination of all parameters is a significant problem for the PSO implementation. However, there is no any formal methodology to solve the problem because different value-combinations of the parameters result to different characteristics as well as different performance of penalty guided PSO. Therefore, one should note that the best values for the PSO parameters may be case-dependent and based upon the experience from preliminary runs.

The numerical results are shown in Table 4 through 7, in which the best solutions of each problem are reported and compared with solutions reported previously in the literature. Table 4 indicates that the solution of the series problem found by penalty

guided PSO is just slightly better than the solution found by [1] and [18], but is comparatively better than the solutions found by [7], [9], and [11]. Table 5 shows that the solution of series-parallel problem found by PSO is better than those reported in literature [1] [7] [9]. Compared with the solutions found by [4] and [19] in the Table 6, the solutions found by proposed method are relatively with more significant improvement. In Table 7, the solution found by the proposed approach is much better than the previous best know solution by [1], [4] and [19]. Although the penalty guided PSO found better solutions for all the four problems, some of the improvements look extremely tiny (*ex.*, on the order of  $10^{-6}$  or  $10^{-5}$ ). However, it has to be emphasized that even tiny improvement in reliability is often hard to be achieved in high reliability applications.

For measuring the improvement, *MPI* (maximum possible improvement) can be used to measure the amount improvement of the solutions found by the proposed approach to the previous best know solutions [3]. *MPI* is the fraction that the best feasible solution achieved of the maximum possible improvement, it is described as:

$$MPI (\%) = (R_{s\_PSO} - R_{s\_other}) / (1 - R_{s\_other})$$

Where  $R_{s\_PSO}$  represents the system reliability obtained by the proposed approach and  $R_{s\_other}$  represents the system reliability obtained by other approaches in literature. By using the index, it shows the proposed approach made more improvements in *P2~P4*. Especially in Table 7 (*P4*), the solution found by the proposed approach is much superior to those in previous literature. As above comparisons, the solutions found by the proposed approach can dominate any other methods for the four problems.

According to the comparison results from Table 4-7, it indicates the solutions obtained by IAs and PSO are very close so that additional analysis is done and described in Table 8. In Table 8, it shows the solutions obtained by penalty guided PSO are more effective and efficient than by IA in terms of solution quality and CPU time. In this experiment, all test problems are solved by the proposed penalty guided PSO and IA for twenty runs with identical population size and maximum generations. The solutions found by PSO generally are much better than by IA but with much fewer CPU time. It can conclude that the penalty guided PSO can find the near optimal solutions more effectively and efficiently. As the above observations, it shows that the performance of the penalty guided PSO is superior to other methods.

**Table 4.** Comparison of the best PSO solution with other approaches for *P1*

	[7]	[11]	[18]	[9]	[1]	<b>PSO</b>
<b>n</b>	(3,2,2,3,3)	(3, 3,2,3,2)	(3,2,2,3,3)	(3,2,2,3,3)	(3,2,2,3,3)	(3,2,2,3,3)
	0.777143	0.77960	0.77939	0.779427	0.779266	0.779435
	0.867514	0.80065	0.87183	0.869482	0.872513	0.871805
<b>r</b>	0.896696	0.90227	0.90288	0.902674	0.902634	0.902824
	0.717739	0.71044	0.71139	0.714038	0.710648	0.711503
	0.793889	0.85947	0.78779	0.786896	0.788406	0.787720
$R_s$	0.931363	0.92975	0.931677	0.931578	0.931678	<b>0.931682</b>
<i>MPI (%)</i>	0.4648%	2.7502%	0.0073%	0.1520%	0.0059%	
Slacks of ( $g_i \sim g_j$ )	27	27	27	27	27	27
	0.000000	0.000010	0.013773	0.121454	0.001559	0.625102
	7.518918	10.57248	7.518918	7.518918	7.518918	7.518918

*Note:* Slack is the unused resources;  $MPI (\%) = (R_{s\_PSO} - R_{s\_other}) / (1 - R_{s\_other})$

**Table 5.** Comparison of the best PSO solution with other approaches for  $P2$ 

	[7]	[9]	[1]	<b>PSO</b>
<b>n</b>	(3,3,1,2,3)	(2,2,2,2,4)	(2,2,2,2,4)	(2,2,2,2,4)
	0.838193	0.785452	0.812485	0.819178
	0.855065	0.842998	0.843155	0.844602
<b>r</b>	0.878859	0.885333	0.897385	0.895837
	0.911402	0.917958	0.894516	0.895151
	0.850355	0.870318	0.870590	0.868685
$R_s$	0.99996875	0.99997418	0.99997658	<b>0.99997665</b>
<b>MPI (%)</b>	25.280%	9.566%	0.299%	
Slacks of ( $g_1 \sim g_3$ )	53 0.000000 7.110849	40 1.194440 1.609289	40 0.002627 1.609289	40 0 1.609289

Note: Slack is the unused resources; MPI (%) =  $(R_{s\_PSO} - R_{s\_other}) / (1 - R_{s\_other})$

**Table 6.** Comparison of the best PSO solution with other approaches for  $P3$ 

	[7]	[9]	[1]	<b>PSO</b>
<b>n</b>	(3,3,2,32)	(3,3,3,3,1)	(3,3,3,3,1)	(3,3,3,3,1)
	0.814483	0.814090	0.812485	0.815878
	0.821383	0.864614	0.867661	0.868265
<b>r</b>	0.896151	0.890291	0.861221	0.859217
	0.713091	0.701190	0.713852	0.711529
	0.814091	0.734731	0.756699	0.752922
$R_s$	0.99978937	0.99987916	0.99988921	<b>0.99988934</b>
<b>MPI (%)</b>	47.462%	8.424%	0.117%	
Slacks of ( $g_1 \sim g_3$ )	18 1.854075 4.264770	18 0.376347 4.264770	18 0.001494 4.264770	18 0 4.264770

Note: Slack is the unused resources; MPI (%) =  $(R_{s\_PSO} - R_{s\_other}) / (1 - R_{s\_other})$

**Table 7.** Comparison of the best PSO solution with other approaches for  $P4$ 

	[4]	[19]	[1]	<b>PSO</b>
<b>k</b>	(6,6,3,5)	(3,6,3,5)	(5,5,5,5)	(5,6,4,5)
	0.81604	0.965593	0.903800	0.901899
	0.80309	0.760592	0.874992	0.849636
<b>r</b>	0.98364	0.972646	0.919898	0.948071
	0.80373	0.804660	0.890609	0.888268
	$R_s$	0.99961	0.999468	0.999942
<b>MPI(%)</b>	88.46%	91.54%	22.41%	
Slacks of ( $h_1 \sim h_2$ )	65 0.064 4.348	92 -70.733576 127.583189	50 0.002152 28.803701	50 0 28.803701

Note: Slack is the unused resources; MPI (%) =  $(R_{s\_PSO} - R_{s\_other}) / (1 - R_{s\_other})$

**Table 8.** Performance Comparison with IA to Penalty Guided PSO (20 Runs)

Items	<b>PSO</b>				<b>IA</b>			
	P1	P2	P3	P4	P1	P2	P3	P4
Population	80	80	80	80	80	80	80	80
Max iter.	500	200	200	200	500	200	200	200
<i>Best R<sub>s</sub></i>	0.93168215	0.99997665	0.99988934	0.99995467	0.931678	0.99997658	0.99988921	0.99994218
<i>Avg. R<sub>s</sub></i>	0.92781729	0.99996734	0.99985940	0.99994815	0.92737864	0.99996169	0.99985484	0.99993137
<i>Avg. CPU</i> (sec.)	3.77	1.54	1.81	1.47	250.53	100.28	102.04	102.93

## 4 Conclusions

This paper presents penalty guided PSO for solving various reliability design problem, which include series systems, series-parallel system, complex (bridge) system and overspeed protection system. In these optimization problems, both the redundancy (number of redundant components) and the corresponding reliability of each component in each subsystem under multiple constraints are to be decided simultaneously. Compared with other traditional evolutionary computation based approaches, penalty guided PSO are able to provide better solutions with much fewer CPU time. As demonstrated in the previous section, the best solutions found by penalty guided PSO are all better than or tie the well-known best solutions by other heuristic methods for mixed-integer reliability problems. While the improvement may be too small to be insignificant, our limited experience suggests that the penalty guided PSO find solutions which are of a quality and are comparable to that of other heuristic algorithms. The proposed method achieves the optimal/near optimal solution for all problems in this work more effectively and efficiently.

## Acknowledgment

The research is supported by grants from National Science Council, Taiwan under contract NSC 93-2213-E-150-012.

## References

- Chen, T.-C.: An immune algorithm based approach for reliability design problems. *in Proceeding of International Conference on Operations and Supply Management*. Bali, Indonesia (2005)
- Chen, T.-C., You, P.-S.: Immune Algorithm based Approach for Redundant Reliability Problems. *Computers in Industry*, Vol. 56. (2005) 195-205.
- Coit, D.W., Smith, A.E.: Reliability optimization of series-parallel systems using a genetic algorithm. *IEEE Transaction on Reliability*, Vol. 45 (1996) 254-260.
- Dhingra, A.K.: Optimal apportionment of reliability & redundancy in series systems under multiple objectives. *IEEE Transactions on Reliability*, Vol. 41(4) (1992) 576-582.
- Gopal, K., Aggarwal, K.K.: A new method for solving reliability optimization problem. *IEEE Transaction on Reliability*, Vol. R-28 (1978) 36-38.
- Hillier, F.S., Lieberman, G.J.: Operations research. McGraw-Hill, New York (1995)
- Hikita, M., Nakagawa, Y., Harihisa, H.: Reliability optimization of systems by a surrogate-constraints algorithm. *IEEE Transactions on Reliability*, Vol. R-41(3) (1992) 473-480.
- Hsieh, Y.-C.: A linear approximation for redundant reliability problems with multiple component choices. *Computers and Industrial Engineering*, Vol. 44 (2003) 91-103.
- Hsieh, Y.-C., Chen, T.-C., Bricker, D.L.: Genetic algorithms for reliability design problems. *Microelectronic Reliability*. Vol. 38 (1998) 1599-1605.
- Kennedy, J., Eberhart, R.C.: Particle swarm optimization. *in Proceedings of IEEE International Conference on Neural Networks*. Perth, Australia, Vol. IV (1995) 1942-1948.
- Kuo, W., Hwang, C.L., Tillman, F.A.: A note on heuristic methods in optimal system reliability. *IEEE Transactions on Reliability*, R-27, (1978) 320-324.

12. Kuo, W., Prasad, V.R.: An annotated overview of system-reliability optimization. *IEEE Transaction on Reliability*, Vol. 49 (2) (2000) 176-187.
13. Misra, K.B.: A simple approach for constrained redundancy optimization problem. *IEEE Transactions on Reliability*, Vol. R-21 (1972) 30-34.
14. Misra, K.B., Sharma, J.: A new geometric programming formulation for a reliability problem. *International Journal of Control*, Vol. 18 (1973) 497-503.
15. Nakagawa, Y., Miyazaki, S.: Surrogate constraints algorithm for reliability optimization problems with two constraints, *IEEE Transaction on Reliability*, R-30, (1981) 175-180.
16. Park, K.S.: Fuzzy apportionment of system reliability. *IEEE Transactions on Reliability*, Vol. R-36 (1987) 129-132.
17. Shi, Y., Eberhart, R.C.: A modified particle swarm optimizer. *in Proceeding of IEEE Congress on Evolutionary Computation*. Anchorage, Alaska, (1998) 69-73.
18. Xu, Z., Kuo, W., Lin, H.H.: Optimization limits in improving system reliability. *IEEE Transactions on Reliability*, Vol. R-39 (1990) 51-60.
19. Yokota, T., Gen, M., Li, Y.-X.: Genetic algorithm for non-linear mixed integer programming problems and its application. *Computers and Industrial Engineering*, Vol. 30(4) (1996) 905-917.

# Developing Methodologies of Knowledge Discovery and Data Mining to Investigate Metropolitan Land Use Evolution

Yongliang Shi<sup>1,\*</sup>, Jin Liu<sup>2</sup>, Rusong Wang<sup>1</sup>, and Min Chen<sup>2</sup>

<sup>1</sup> Research Center for Eco-Environment Sciences, Chinese Academy of Science  
18 Shuangqing Road, Beijing, China 100085  
vectorise@hotmail.com,  
wangrs@mail.rcees.ac.cn

<sup>2</sup> State Key Laboratory of Software Engineering, Wuhan University  
Luojiashan, Wuhan, China 430072  
typhoon2008@hotmail.com,  
lineararrow@163.com

**Abstract.** In the urban/territorial planning process, the quality of the evaluation procedure is crucial. It is necessary to select and implement innovative tools able to handle the huge amount of available data concerning territorial systems in order to extract useful information from them to enhance the quality of evaluation procedure for urban/territorial planning. This paper selects some tools derived from Artificial Intelligence, and incorporated GIS through the elaboration of various types of available data, to extract and build knowledge directly from experimental data and also to represent the extracted knowledge very effectively and communicatively, in the form of sets of spatial transformation rules. It describes the structure of the data mining tools which are most suitable for applications in the field of urban planning, aimed at discovering the transformation rules driving the evolution of cities in special of metropolitan in analysis.

**Keywords:** Knowledge discovery, Data mining, Machine learning, Spatial and temporal reasoning.

## 1 Introduction

The Date Mining and Knowledge Discovery in Database (KDD) field is nowadays receiving growing attention and interest from the business world and research community. In fact, over the past few years, research and development work in data mining has made great progress and a large number of application papers have appeared in the literature [1,2]. Many successful applications have been reported in various sectors such as marketing, finance, manufacturing, and telecommunications. In the last few years, some research has also emerged for applying data mining techniques to discover implicit knowledge in spatial databases [3,4]. In fact, with the present wide application of remote sensing technology and automatic data collection

---

\* Corresponding author.

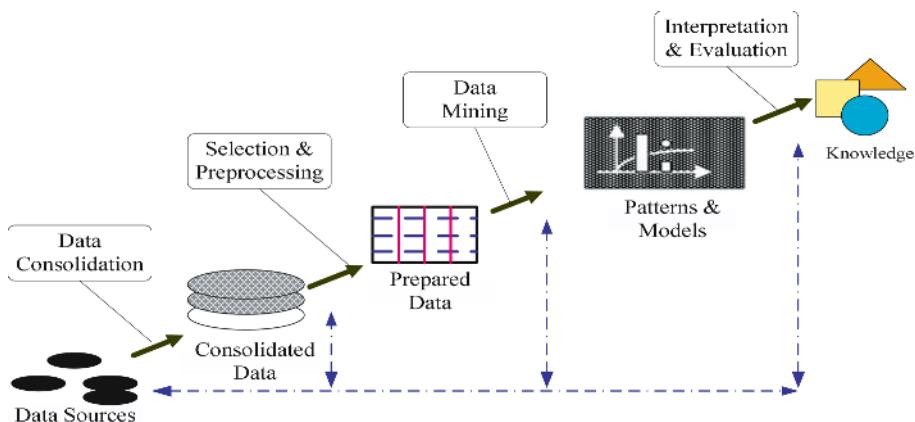
tools, a tremendous amount of spatial data has been collected and stored in databases. It is necessary to select and implement innovative tools able to handle the huge amount of available data concerning territorial systems (statistical, economic, cartographic data and so on) in order to extract useful information from them to enhance the quality of evaluation procedure for urban/territorial planning. In the context, we propose building cognitive systems (intelligent systems) which, on observing a given reality, are able to extract knowledge regarding the role played by urban/territorial factors and their relations in urban/territorial evolution and change [4]. To perform this task, innovative and efficient tools and methodologies can be found in the field of Data Mining and Knowledge Discovery in Database Process, a new technology which merges concepts and techniques from many different research areas, such as statistics, machine learning, etc.

We describe the goals and tools of the “knowledge builder and manager”, and illustrate the aims in applying some of these tools to the investigation of the causes of land use evolution in a metropolitan context. On the basis of a survey on the state of the art in presently available tools, we select some tools derived from Artificial Intelligence, and incorporate GIS through the elaboration of various types of available data (thematic maps, population density, transport infrastructures, maps of ecology and historical heritage, etc.), to extract and build knowledge directly from experimental data and also to represent the extracted knowledge very effectively and communicatively, in the form of sets of spatial transformation rules [5]. In this paper we describe the structure of the data mining tools which are most suitable for applications in the field of urban planning, aimed at discovering the transformation rules driving the evolution of cities in special of metropolitan analysis. In particular, one of the tools most suitable for the aims of the work is classification: the classification-based methodology is described in detail, paying particular attention to the problems of database structure, parameter setting, pruning level, etc. We carry out experiment connected to case study investigating the evolution of land use in the metropolitan area of Beijing between 1992 and 2002. The experimental applications of knowledge discovery learning tools, coupled and integrated with GIS. It must be stressed that the GIS was used not only as a georeferenced archive and a quick information retrieval tool, but it was programmed in order to be integrated with the AI model and to produce additional geographic information [5]. The remainder of the paper is organized as follows. Section 2 introduces the methodologies and tools and section 3 methodologies implementation: evolution of land use in the metropolitan of Beijing. Section 4 draws some conclusions from this work.

## 2 Methodologies and Tools

### 2.1 Data Mining and Knowledge Discovery in Databases

Extracting the knowledge which is hidden in the databases and exploiting this knowledge in support of the decision-making process is the main goal of KDD. In our research, KDD is a process consisting of many steps, which enables the data analyst to progress from raw, inconsistent and noise data to interesting and actionable knowledge [6,7]. Fig.1 shows a summary of the steps of KDD process [7].



**Fig. 1.** Steps in the KDD process

## 2.2 Data Resource and Data Mining Algorithms

The initial data we adopted for our application are very heterogeneous, consisting in economic, statistical and cartographical data; in particular we mainly used thematic maps (land use maps, maps of geological and historical heritage, transport networks, etc.) and various kinds of census data (population density, etc.).

The general assumption of our experimental work is that the factors underlying urban change are: 1) the land use and the presence of various kinds of infrastructure in each elementary territorial cell. 2) the spatial relations of each cell with other cells, measured in terms of “strict adjacency” and “enlarged adjacency”.

For each spatial entity there is a description (defined by means of alphanumeric attributes) of the state of the entity itself and a description of the states of the neighboring entities. All the results were presented in the form of relational tables, using the previously quoted automated procedures within a GIS.

Fig.2 and Fig.3 show the way in which the territory was modeled and the correspondence between cartographic and relational data (Tab.1).

5031	6042	6124	
<b>6</b>	<b>6</b>	<b>6</b>	
5030	6041	6123	
<b>6</b>	<b>6</b>	<b>6</b>	
5029	6040	6122	
<b>4</b>	<b>4</b>	<b>7</b>	

**Fig. 2.** Adjacent

6	6	6	6	6	6	7	6
6	6	6	6	6	6	7	7
6	6	6	6	6	6	7	7
6	6	6	6	6	6	4	6
4	4	4	4	4	7	4	6
6	4	4	4	4	7	4	4
6	6	4	6	7	4	4	4

**Fig. 3.** Neighboring

**Table 1.** Report Table Corresponding to a Single Territorial Entity (cell identified as 6041 in Fig.2)

Description of the "core" cell		Description of the land use of the adjacent cell							Description of the land use of the neighboring cell							Other information
ID number	Land use	Land use 1	Land use 2	Land use 3	Land use 4	Land use 5	Land use 6	Land use 7	Land use 1	Land use 2	Land use 3	Land use 4	Land use 5	Land use 6	Land use 7	...
6041	6	0	0	0	2	0	5	0	0	0	0	11	0	27	8	...

In this table each line represents an elementary territorial entity and each column an attribute useful for describing a characteristic of this entity. The values that each single attribute may assume can be continuous or discrete. In our applications, it was in many cases preferable to express each attribute by means of discrete. In our applications, it was in many cases preferable to express each attribute by means of discrete quantitative values in order to more easily understand the information stored within the data set. In this case the conversion can be performed by defining a range of numerical values corresponding to a qualitative one (Table 2). For instance, if we define: the information contained in Table 1 can be put into the form reported in Table 3.

Both of these relational tables (numerical and discrete) are in a form suitable for insertion into a mining algorithm.

**Table 2.** Definition to a range of numerical values

<b>For adjacent description</b>		0=nothing(N) 1-2=low(L) 3-4=medium(M) >4=high(H)
<b>For neighboring description</b>		0=nothing(N) 1-10=low(L) 11-20=medium(M) >20=high(H)

**Table 3.** Report Table Corresponding to a Single Territorial Entity (cell identified as 6041 in Fig.2) with Discrete Values of Attributes

Description of the "core" cell		Description of the land use of the adjacent cell							Description of the land use of the neighboring cell							Other information
ID number	Land use	Land use 1	Land use 2	Land use 3	Land use 4	Land use 5	Land use 6	Land use 7	Land use 1	Land use 2	Land use 3	Land use 4	Land use 5	Land use 6	Land use 7	...
6041	6	N	N	N	L	N	H	L	N	N	N	M	N	H	L	...

### 2.3 Classification and Decision Trees

Classification is a daily human activity. In our context, classification is the process of finding a set of models (or functions) that describe and distinguish data classes or concepts, for the purpose of being able to use the model to predict the class of objects whose class label is unknown.[8]

The derived model may be represented by a set of classification (IF-THEN) rules, decision trees or neural networks [9, 10].The decision tree method is the classification technique used in this paper. Neural networks can provide accurate predictions in the discrete and continuous domain: the problem is that the results are sets of mathematical formulas, that is, they are incomprehensible black boxes. Many decision tree induction algorithms have been developed by the machine-learning community. In our experiments, we used C5.0, the most recently available version of the decision tree algorithm that J.Ross Quinlan has been evolving and refining for many years [11].

With regard to the data set we adopted for our applications, many relational tables have been built, with a different number of attributes used to describe each single cell. This is because various classes of distance (and consequently of criteria for neighboring cells) have been defined and tested, ranging from 500 to 2000 meters corresponding to the "importance" of each attribute [12]. For instance, after various trials preformed using different values of neighboring radiiuses, we realized that the presence of a metro station influenced cells within a distance of about 500 meters, while the presence of a park or a railway access was felt at a greater distance(1000 or 2000 meters, Fig.4).

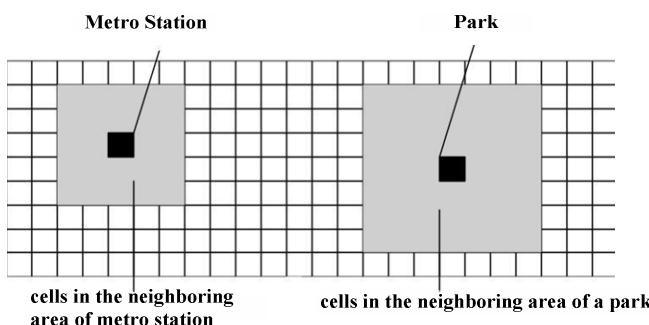
For all the applications, the attribute corresponding to land use value at, the second time slice was chosen as class label. In other words, a class value was assigned to each cell by adding a column to the report table, as in Table 4.

The extracted rules are in the form:

IF→Description of cell and cell's neighboring areas at the first date

THEN→Land use at the second date

In our applications we used various values of pruning for each data set, in order to obtain sets of rules characterized by various degrees of precision that is more or less detailed [13].



**Fig. 4.** Neighboring Radiiuses Assigned to Various Attributes

**Table 4.** Report Table Corresponding to a Single Territorial Entity Including Class Label

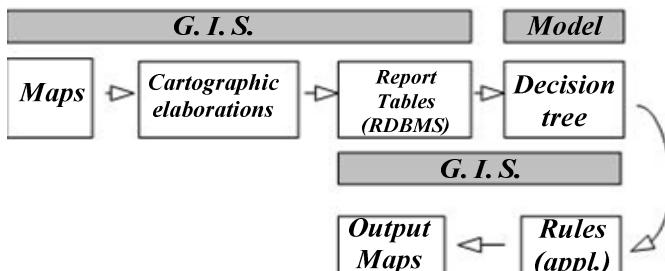
ID number	Description of the “core” cell		Description of the land use of the adjacent cell							Description of the land use of the neighboring cell							Class label
	Land use (time 1)	Land use 1	Land use 2	Land use 3	Land use 4	Land use 5	Land use 6	Land use 7	Land use 1	Land use 2	Land use 3	Land use 4	Land use 5	Land use 6	Land use 7	Land use (time 2)	
6041	6	N	N	N	L	N	H	L	N	N	N	M	N	H	L	...	

### 3 Methodologies Implementation: Evolution of Land Use in the Metropolitan of Beijing

#### 3.1 System Structure

We describe the results of experimental application of a knowledge discovery learning tool (decision trees) aimed at discovering the relations and rules driving the evolution of some specific urban systems. We carried out experiment connected case-studies of the evolution of land use in the metropolitan area of Beijing between 1992 and 2002. For this application, the territory was divided into elementary units whose size and shape depend on the disaggregating level and on the nature of available information[12].The input databases were prepared using both statistical(census data on population and economic activities)and geographical (land use, road network maps, etc.) data, each referring to two different time sections. The information includes the initial and final state (land use) of each territorial unit and the quantitative and qualitative identification of the state of its neighboring areas.

The experiments produced some sets of IF-THEN rules revealing some of the main transformation processes which took place in the studied areas in the considered time period. The study of the described phenomena was performed by using and integrating two different tools: GISs and automatic learning tools. Fig.5 shows the system structure.

**Fig. 5.** System structure

### 3.2 Spatial Data Elaboration

By means of a GIS tool the primary data was converted into a format permitting subdivision of the territory into homogenous elementary units (cells  $100 \times 100\text{m}$ ): this representation enables conceptually correct and very quick spatial elaboration to be performed. After defining the shape and characteristics of each entity, it is then possible to select the local elements that are in the neighboring area of every elementary cell [5,12].

The development and the building of a tool able to achieve the previous aim covers different phases[12]:

- building of maps referred to different time slices;
- building of automated procedures within GIS in order to perform various kinds analysis such as map overlays, neighboring analysis, etc.;
- building of automated report tables referred to cartographic analysis;
- implementation of an algorithm able to extract land transformation rules;
- building of thematic maps referred to the output data obtained in the previous phases.

The basic map for the analysis were for the year 1992 and 2002 was provided by the Land Administrate Bureau of Beijing .They are two of a series of maps at scale 1:15000 indicating land uses of the whole of Beijing Municipality. We georeferenced these maps and then we coded all the different land use values.

### 3.3 Results: Evolution of “Functional” Land Use

The initial data base of Beijing metropolitan was made by 960 land use variables referring to 6300 census areas, derived from the 1992 and 2002 censuses. The variables were selected and grouped in land use classes following the criterion of homogeneity with respect to location behavior. The land uses adopted are described in Table 5.

**Table 5.** Classes in the Metropolitan Area of Beijing

<b>Land use classes</b>	
<b>Prevailing land uses</b>	<b>Mixed land uses</b>
1) Empty(V)	9)Residential+ education (R-Ce)
2) High-density residential (R)	10)Residential+ commerce (R-Cc)
3) Low-density residential (R1)	11)Residential+industry(R-IN)
4) Education (Ce)	12)Residential+health services(R-Ch)
5) Commerce (Cc)	13) Commerce + education (Cc-Ce)
6) Industry (IN)	14) Education +industry(Ce-IN)
7) Health services (Ch)	15) Commerce +industry(Cc-IN)
8) Green(G)	

Moreover there are fixed characteristics (attributes) of the cells: they do not change, but influence the evolution of the system. Therefore, they do not belong to the target map. They correspond to the presence in any cell of elements relevant for

accessibility or characterizing the whole surface of the cell. They are: 1) the presence of railway or underground stations; 2) the presence of access points to the ring road; 3) parks and not transformable areas (by the Master Plan); 4) green-land; 5) other municipalities; 6) adjacency to a cell which presents one of the 5 attributes listed above; 7) historic center. In order to study the same area by means of decision tree induction, we used the above described set of data, but we adopted a large set of information for each cell, including the definition of various kinds of enlarged neighboring areas (cells in a radius of 500, 1000 and 2000 meters).

The report table we adopted included the information described in Table 6.

**Table 6.** Report Table ( parts)

Attributes (IF)	
<b>Land use 1992</b>	
1	Presence of Metro station:
2	Metro stations in the adjacent cells
3	Metro stations in the neighboring cells (radius 500 meters) in adjacent cells
4	Metro station in the neighboring cells (radius 1000meters)
5	Presence of Highway access:
6	Highway access in the adjacent cells
7	Highway access in the neighboring cells (radius 1 000 meters)
...	...
38	(R-IN) in the neighboring cells (radius 900 meters)
39	(R-Ch) in the neighboring cells (radius 900 meters)
40	(Ce-Cc) in the neighboring cells (radius 900 meters)
41	(Ce-IN) in the neighboring cells (radius 900 meters)
42	(Cc-IN) in the neighboring cells (radius 900 meters)
Class label (THEN)	
<b>Land use 2002</b>	

From the first experiment results we obtained some realistic information about the evolution processes taking place within the metropolitan area of Beijing between 1992 and 2002. The most significant results are represented in the following tables.

We can observe that in the case high-density residential appear in proximity to already dense areas: it indicates a diffusion process driven by density, rather than the birth of new clusters. In the case of new expansion, there is the need for empty areas, with some access to services. Densification of course appears in low-density areas and is driven by accessibility to transport infrastructures. Dense residential urbanization also appears in areas with good access to services and with good environmental quality (Table 7).

The evolution of education activities is a significant issue in a large metropolitan area which is the capital city. It has often been seen as “invading” the most valuable areas of the city. Indeed, we can see that education activities drove away other activities (2 Substitution) in areas with the best and more prestigious urban environment (historic center and its surroundings), endowed with services and with good accessibility (Table 8).

**Table 7.** 1992-2002 Residential Urbanization

In the rules found, three types of evolution can be identified:

<b>1 EXPANSION</b> In presence of:	<b>2 DENSIFICATION OF AREAS WITH GOOD ACCESSIBILITY</b> In presence of:	<b>3 URBANIZATION OF AREAS WITH A GOOD ENVIRONMENTAL QUALITY</b> In presence of:
-Empty cells in the enlarged neighboring cells	-Low-density residential inside a given radius	-High-density residential in the neighboring cells
-High-density residential in the enlarged neighboring cells	-High-density residential in the neighboring cells	-Some services located inside a given radius
-Services in the enlarged neighboring cells	-Rail stations and/or metro stations in the radius	-Parks(adjacent in the radius)

**Table 8.** Location Choices of Education Activities

In the rules found, three main types of evolution can be identified:

<b>1 GROWTH( from R-Ce, Ce-Cc)</b>	<b>2 SUBSTITUTION</b>		<b>3 NEWLOCATIONS (from V)</b>
	<b>2a:Centralisation and image effect</b>	<b>2b:Dragging and “quality” effect</b>	<b>Image effect and city effect</b>
-Available areas in the enlarged neighboring cells -Demand from firms in the enlarged neighboring cells	-Nearness to the historic center -Adjacent to historic center -presence of metro stations in the neighboring cells	-Education in the enlarged neighboring cells -Greenland in the enlarged neighboring cells -Nearness to areas with good environment context -Areas with good accessibility	-Presence of high-density residential -Nearness to historic center -Nearness to parks

## 4 Conclusion

Knowledge Discovery is a bottom-up approach, which starts from data and tries to induce some relevant features that are not known in advance. In this analysis, the data themselves suggest conjectures on their semantics. In the literature, knowledge discovery is further characterized as either directed or undirected [14]. In directed knowledge discovery; the goal of the process is to explain the value of some particular attribute in terms of others. A typical data mining tool for directed knowledge discovery is classification, whereas typical tools for undirected knowledge discovery are clustering and association rules [8, 10]. In this paper, we refer to a process of directed knowledge discovery, based on classification techniques incorporated GIS. The follow-on paper applies the techniques to investigate the evolution of urban land use of a real city--the metropolitan Beijing, which demonstrates the realistic nature of these methodologies and tools' results. Compared to other methodology to analysis urban land use evolution, methodologies based on knowledge discovery and data mining possesses several advantageous features for investigating urban development processes, the main one being using decision tree to handle complex context and rule of development. With the application of decision tree and classification, the state of a cell is associated with a grade of membership representing the stage a cell is in its urban development process [12]. The grade of membership represents urban development as a continuous process in space and over time, rather than as a binary non-urban to urban conversion process [15]. In addition,

the use of decision tree provides transition rules more closely linked to human decision-making behavior. Advantages for investigating urban land use evolution also come from combining these AI methodologies and tools in the GIS. With this combination we can easily control the performance of the system, visualize its output and calibrate it. Experimental application of the methodologies to a city produced report table of land use evolution supporting the approach.

**Acknowledgements.** Financial support for this work is provided by the Capital Urban Planning Commission, the National Natural Science Foundation of China (NSFC) under Grant No.70433001 and the Projects of Natural Science Foundation of Hubei Province, No.2005ABA240 and No.2005ABA123. The authors are indebted to Professor LI Manchun of Nanjing University for comments on this paper. The authors wish to express their gratitude to the interviewees and particularly to Ms. HE Yong, the technician in the Institution of Beijing Urban Planning and Design.

## References

1. Holsheimer, M., Kersten, M., Mannila, H., Toivinen, H., A perspective on databases and data mining, *Proceedings of the First International Conference on Knowledge Discover and Data Mining*, KDD 1995, 150-155
2. Han,J.,Kamber,M. Data Mining:Concepts and Techniques, Morgan Kaufmann Publishers (2000)
3. Silva, E. A., & Clarke, K. C.. Calibration of the SLEUTH Urban Growth Model for Lisbon and Porto, Portugal. *Computers, Environment and Urban Systems* 5(2002)525-552
4. Batty, M.. GeoComputation using cellular automata. In *GeoComputation* ,S.Openshaw, & R. J.Abrahart (Eds.), Taylor & Francis , London UK (2000) 95-126
5. Bolliger, J., Simulating complex landscapes with a generic model:sensitivity to qualitative and quantitative classifications. *Ecological Complexity* 2(2005) 131-149
6. Shi Zhongzhi, Advanced Artificial Intelligence, Science Press (in Chinese), Beijing, China(1998)
7. Mitchell,T.M., Machine Learning, McGraw Hill (1997)
8. Breiman, L., Friedman,J., Olshen R., Stone C., Classification and Regression Trees Wadsworth International Group (1984)
9. Liu,B.,Hsu,W.,Ma,Y, Integrating classification and association rule mining, *Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining*,KDD1998, 80-86
10. Kamber,M.,Winstone,L.,Gong,W.,Cheng,S.,Han,J., Generalisation and decision tree induction: Efficient classification in data mining, *Proceedings of International Workshop Research issues on Data Engineering*, RIDE1997,111-120
11. Quinlan,J.R.,C4.5:program for Machine Learning, Morgan Kauffmann Publishers(1993)
12. Batty, M., Couclelis, H., & Eichen, M. Urban systems as cellular automata. *Environment and Planning B-Planning & Design* 2(1997)159-164
13. Liu, B., Hsu, W., Mun, L., Lee, H, Finding interesting patterns using user expectations. *IEEE Transactions on Knowledge and Data Engineering* 6(1999) 817-832
14. Mannila,H.. Local and global methods in Data Mining:basic techniques and open problems, *Proceedings of the 29th International Colloquium on Automata, Languages and Programming*, ICALP 2002, LNCS 2380, 57-68
15. Lidia,D.,Evolving Cities: Geocomputation in Territorial Planning. Ashgate Press, Chippenham,Wiltshire(2004)

# Vibration Control of Suspension System Based on a Hybrid Intelligent Control Algorithm

Ke Zhang<sup>1</sup> and Shiming Wang<sup>2</sup>

<sup>1</sup> School of Mechanical and Automation Engineering  
Shanghai Institute of Technology, 200235 Shanghai, China  
zkwy@hotmail.com

<sup>2</sup> College of Engineering Science & Technology  
Shanghai Fisheries University, 200090 Shanghai, China  
xjtuwangshiming@163.com

**Abstract.** Vibration control is a pivotal study subject for vehicle suspension system. In this paper, the physical model of semi-active quarter-coach suspension was established by the base of the theories of Buckingham's Pi theorem. According to the characteristics of the semi-active suspension of diesel-truck, a hybrid intelligent control algorithm-Fuzzy cerebellar model articulation control combined the Fuzzy logic control with cerebellar model articulation control techniques was presented and used to perform online control of semi-active suspension for the first time, novel weight-update laws were derived that guarantee the stability of closed-loop system, both information retrieval and learning rules were described by algebraic equations in matrix form. The results of experiment of closed -loop excited by three typical vibration signals showed that the control strategy proposed here can obviously reduce the value of mean square root of vertical acceleration for semi-active suspension system, compared with the traditional control strategy.

## 1 Introduction

In order to improve the performance of conventional suspension system, many scholars have been researching automatic control techniques for years. The commercial and scientific activity are mainly two motivations, the mainspring of commercial activity has been the vehicle manufacturers and components suppliers who wish to improve the performance and quality of their products. Form the scientific community, control system researchers have perceived the automatic control of vehicle suspensions as an obvious and desirable goal that is made achievable by new developments in actuators, sensors and low-cost electronics [1],[2],[3]. In recent years, various active and semi-active strategies have been proposed to improve performance and remove the restrictions of passive suspensions. The active suspension requires a power source to generate suspension forces according to some prescribed criterion. Such a system suffers from high cost, complexity and considerable external power requirement. As a result, semi-active suspension has appeared which uses basically dissipative elements for which the force can be actively modulated. Thus research on semi-active suspension system and control strategy has become the international

advanced subject in the field of theory and engineering of automobile in recent years. Although the theoretical research has been developed, the problems on physical realization and online control strategy remain somewhat difficult. In this paper, a physical model of a two-degrees-of-freedom of semi-active suspension of diesel-truck is developed based on the theories of resemble and model testing. In terms of the model, the online control of semi-active suspension is performed by using Fuzzy cerebellar model articulation control algorithm (FCMAC). The potential performance improvements offered by FCMAC-semi-active suspension system are evaluated and analyzed in comparison with PID strategies.

## 2 Physical Model of Semi-active Suspension

### Nomenclature

$r$ — road disturbance input	$\xi$ — ratio of damp of servo valve
$m_1, m_2$ — unsprung mass, sprung mass	$B$ — damping coefficient
$k_1, k_2$ — tyre stiffness, spring stiffness	$x_v$ — displacement of servo valve
$\omega_v$ — inherent frequency of servo valve	$q$ — load flow of servo valve
$p_l$ — pressure difference across the actuator piston	$K_q$ — discharge gain of servo valve
$\beta_e$ — bulk modulus of the hydraulic fluid	$K_v$ — gain of servo amplifier
$C_p$ — total leakage coefficient	$\rho$ — density of hydraulic fluid
$V_t$ — total volume of the actuator cylinder	$u_v$ — input of servo valve
$K_c$ — discharge coefficient of servo valve	$u$ — control input
$x$ — state	$y$ — measured output

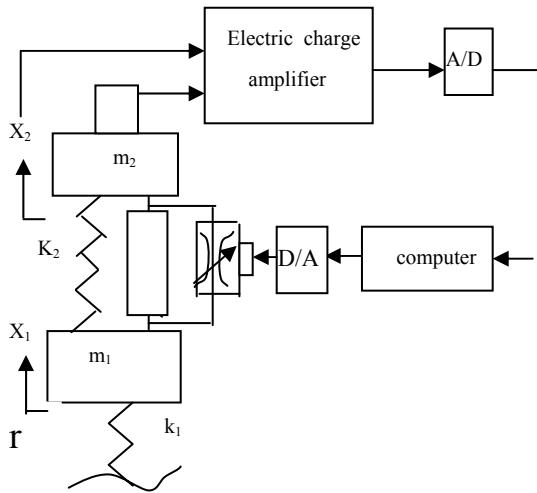
### 2.1 Dynamic Model

Quarter-coach model is very often used for suspension analysis and design. Because it is simple yet capture many important characteristics of the full model. Fig. 1 shows a Quarter-coach model of semi-active suspension system, the spring, shock absorber and active force-generating element placed between the sprung and unsprung masses constitute the suspension. A realistic representation of the entire system is simply replaced by a damper with a rate that could be varied continuously between maximum and minimum limits.

The motion of the semi-active suspension system can be expressed by the differential equations:

$$\begin{aligned} m_1 \ddot{x}_1 + k_1(x_1 - r) + k_2(x_1 - x_2) + B(\dot{x}_1 - \dot{x}_2) + A_s(p_1 - p_2) &= 0 \\ m_2 \ddot{x}_2 - k_2(x_1 - x_2) - B(\dot{x}_1 - \dot{x}_2) - A_s(p_1 - p_2) &= 0 \end{aligned} \quad (1)$$

$$p_l = p_1 - p_2$$



**Fig. 1.** Quarter-coach model with semi-active suspension

where  $A_s(p_1 - p_2)$  is the control force from the hydraulic system. The system model is completed by the electromechanical servo system which drive the servo piston, it is approximated describe by the relation between the control voltage  $u_v$  and displacement of servo valve  $x_v$  by

$$\frac{x_v}{u_v} = \frac{K_v}{\left| \frac{s}{\omega_v} \right|^2 + 2\xi \frac{s}{\omega_v} + 1} \quad (2)$$

The oil flow is described by a differential equation of the form

$$q = A_s(\dot{x}_2 - \dot{x}_1) + C_{ip}p_l + \frac{V_t}{4\beta_e} \dot{p}_l \quad (3)$$

Above equation linearized

$$q = A_s(\dot{x}_2 - \dot{x}_1) + C_{ip}p_l + \frac{V_t}{4\beta_e} \dot{p}_l \quad (4)$$

The steady-state gain is taken to be one:

$$\dot{x}_v = \frac{1}{\tau}(-x_v + u) \quad (5)$$

State variables are  $x_1 = x_1$ ,  $x_2 = \dot{x}_1$ ,  $x_3 = x_2$ ,  $x_4 = \dot{x}_2$ ,  $x_5 = p_l$ ,  $x_6 = x_v$ , we can rewrite the above equations as follows:

$$\begin{aligned}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= \frac{1}{m_1} [-k_1(x_1 - r) - k_2(x_1 - x_3) - B(x_2 - x_4) - A_s x_5] \\
\dot{x}_3 &= x_4 \\
\dot{x}_4 &= \frac{1}{m_2} [k_2(x_1 - x_3) + B(x_2 - x_4) + A_s x_5] \\
\dot{x}_5 &= \frac{4\beta_e}{V_t} [-(K_c + C_{ip})x_5 + A_s(x_2 - x_4) + K_q x_6] \\
\dot{x}_6 &= \frac{1}{\tau} (-x_6 + u)
\end{aligned} \tag{6}$$

The output equation is  $y = \ddot{x}_2$

## 2.2 Physical Model

Based on Buckingham's Pi theorem, we choose the resemble variables:

$$m_1, \ddot{x}_1, k_1, x_1, r, k_2, x_2, B, \dot{x}_1, \dot{x}_2, A_s, p_l, m_2, \ddot{x}_2$$

The power of each variable is  $a_i (i=1,2,\dots,14)$ . We choose the  $L$  (length, unite: m),  $T$  (time, unite: s) and  $F$  (force, unite: N) as the basal variable, can represent the relation of dimension of each variable of semi-active suspension system and the basal variable in Table 1.

**Table 1.** Dimension of variables

	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$	$a_6$	$a_7$	$a_8$	$a_9$	$a_{10}$	$a_{11}$	$a_{12}$	$a_{13}$	$a_{14}$
L	-1	1	-1	1	1	-1	1	-1	1	1	2	-2	-1	1
T	2	-1	0	0	0	0	0	1	1	-1	0	0	2	-2
F	1	0	1	0	0	1	0	1	0	0	0	1	1	0

We define the  $\prod$  as relation of each variable of semi-active suspension system

$$\prod = m_1^{a_1} \ddot{x}_1^{a_2} k_1^{a_3} x_1^{a_4} r^{a_5} k_2^{a_6} x_2^{a_7} B^{a_8} \dot{x}_1^{a_9} \dot{x}_2^{a_{10}} A_s^{a_{11}} p_l^{a_{12}} m_2^{a_{13}} \ddot{x}_2^{a_{14}} \tag{7}$$

Based Buckingham's Pi theorem  $\prod'_i = \prod_i (i=1,2,\dots,14)$ , we choose  $m'_1, k'_1, x'_1, k'_2, x'_2, A'_s, m'_2, \ddot{x}'_2$  as the important variables, and consider  $r = r'$ ,  $B = B'$  being true, so the value of each variable of the system is made certain, and a physical model is established.

Automotive manufacturers have had success in using heuristic rule-based controllers for discretely variable damping systems in which the damping rate can be switched between two or three discrete settings. However, an extension of the rule-based controller to the continuously variable damping case was shown to be prohibitively complex and costly.

### 3 FCMAC Neural Networks

Automotive manufacturers have had success in using heuristic rule-based controllers for discretely variable damping systems in which the damping rate can be switched between two or three discrete settings [4], [5], [6], [7]. However, an extension of the rule-based controller to the continuously variable damping case was shown to be prohibitively complex and costly [8], [9]. In cerebellar model articulation control (CMAC) [10], each state variable is quantized and the problem space is divided into discrete states. A vector of quantized input value specifies state and is used to generate addresses for retrieving information from memory for this state. Information is distributively stored. Ability to learn nonlinear input-output mapping is the main advantage of CMAC. However, for the divisionary mode of input space, the relation between input state and the effect of the contents of memory locations for output, their online adjustment can not be performed by using CMAC. By integrating the fuzzy concept into CMAC, we propose FCMAC Neural Networks to improve the disadvantage, the mapping algorithm based in detail as follows: Consider now the multi-input multi-output system. Each input vector is divided into C sub-space, the address of input vectors in receptive field  $A_c$  is:

$$x_n = i_n \sum_{t=1}^{n-1} \frac{x_t}{C} + \sum_{p=2}^{n-1} \left( \frac{i_p - f_{\text{mod}}(i_n, C)}{C} \sum_{s=1}^{p-1} \frac{x_s}{C} \right) + \frac{i_1 - f_{\text{mod}}(i_n, C)}{C} \quad (8)$$

$$\begin{aligned} k &= 0, 1, 2, \dots, C-1, n = 1, 2, \dots, N \\ i_{no} + C > i_{nk} &= nC + k \geq i_{no} \\ n &= 1, 2, \dots, N, k = 0, 1, \dots, C-1 \\ \frac{i_{no} + C - k}{C} > n &\geq \frac{i_{no} - k}{C}, \quad \text{if } \{f_{\text{mod}}(i_{no} - k, C) = 0\} \end{aligned} \quad (9)$$

$$\begin{aligned} i_{nk} - k &= nC \geq i_{no} - k \\ \frac{i_{no} + C - k}{C} &\geq n > \frac{i_{nk} - k}{C}, \quad \text{if } \{f_{\text{mod}}(i_{no} - k, C) = 0\} \\ i_{nk} &= nC + k \end{aligned} \quad (10)$$

Then the multidimensional receptive field function can be defined as:

$$X_{nk} = i_{nk} \sum_{t=1}^{n-1} \frac{x_t}{C} + \sum_{p=2}^{n-1} \left( \frac{i_{pk} - f_{\text{mod}}(i_{nk}, C)}{C} \sum_{s=1}^{p-1} \frac{x_s}{C} \right) + \frac{i_{1k} - f_{\text{mod}}(i_{nk}, C)}{C} \quad (11)$$

Output is:

$$F(S) = \sum_{k=0}^{C-1} W(X_k) \mu(k) \quad (12)$$

The FCMAC weight be turned on by:

$$\Delta W(X) = \beta[d - F(s)]\mu(k)/C \quad (13)$$

In equation (13),  $d$  denotes target value of the output,  $\beta$  denotes the rate of learning, and  $\mu(k)$  is membership of fuzzy set of variables. We consider the case that  $s$  set of training data is repeatedly presented to learning structure, the updating at the presentation of the  $s$  th sample in the  $i$  th iteration is expressed as

$$W_s^i = W_{s-1}^i + \Delta W_{s-1}^i = W_{s-1}^i + \frac{\beta}{C}[d_{s-1} - \mu_{s-1}^T(k)W_{s-1}^i]\mu_{s-1}(k) \quad (14)$$

where subscripts  $s-1$  and  $s$  indicate the sample numbers, the superscript  $i$  indicates the iteration number and  $d_{s-1}$  is the target value of sample  $s-1$ . By using equation (14), the difference of memory contents between two consecutive iterations is calculated as:

$$\begin{aligned} DW_s^i &= W_s^{(i+1)} - W_s^{(i)} \\ &= DW_{s-1}^{(i)} + \frac{\beta}{C}\mu_{s-1}(k)\sum_{k=0}^{C-1}[d_{s-1} - \mu_{s-1}^T(k)W_{s-1}^{(i+1)}] \\ &\quad - \frac{\beta}{C}\mu_{s-1}(k)\sum_{k=0}^{C-1}[d_{s-1} - \mu_{s-1}^T(k)W_{s-1}^{(i)}] \\ &= DW_{s-1}^{(i)} - \frac{\beta}{C}\mu_{s-1}(k)\mu_{s-1}^T(k)DW_{s-1}^{(i)} \end{aligned} \quad (15)$$

Let us further define that

$$\begin{aligned} G_s &= E_{N_s} \cdots E_s E_{s-1} E_{s-2} \cdots E_1 \\ DW^{(i)} &= G_s [DW_1^{(i-1)}, DW_2^{(i-1)}, \dots, DW_{N_s}^{(i-1)}] \\ &= G_s^2 [DW_1^{(i-2)}, DW_2^{(i-2)}, \dots, DW_{N_s}^{(i-2)}] \\ &= G_s^i [DW_1^{(0)}, DW_2^{(0)}, \dots, DW_{N_s}^{(0)}] \\ &= G_s^i DW^{(0)} \end{aligned} \quad (16)$$

With the definition of  $DW_s^{(0)}$

$$\begin{aligned} DW_s^{(0)} &= W_s^{(1)} - W_s^{(0)} = W_{s-1}^{(1)} + \Delta W_{s-1}^{(1)} - W_s^{(0)} \\ &= W_1^{(1)} + \Delta W_1^{(1)} + \dots + \Delta W_{s-1}^{(1)} - W_s^{(0)} \\ &= W_{N_s}^{(0)} + \Delta W_{N_s}^{(0)} + \Delta W_1^{(1)} + \dots + \Delta W_{s-1}^{(1)} - W_s^{(0)} \\ &= W_s^{(0)} + \Delta W_s^{(0)} + \Delta W_{s+1}^{(0)} + \dots + \Delta W_{N_s}^{(0)} + \Delta W_1^{(1)} + \dots + \Delta W_{s-1}^{(1)} - W_s^{(0)} \\ &= \Delta W_s^{(0)} + \Delta W_{s+1}^{(0)} + \dots + \Delta W_{N_s}^{(0)} + \Delta W_1^{(1)} + \dots + \Delta W_{s-1}^{(1)} \end{aligned} \quad (17)$$

We should notice:  $\Delta W_s^{(i)}$  is different from  $DW_s^{(i)}$ ,  $\Delta W_s^{(i)}$  is the updating at the time when  $S$  is presented in the  $i$  th iteration,  $DW_s^{(i)} = W_s^{(i)} - W_s^{(i-1)}$  is change of weight in the  $i$  th iteration and in the  $i-1$ , from equation (13) we can have:

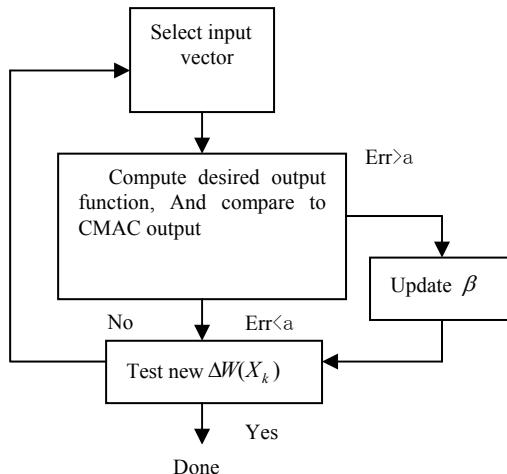
$$\Delta W_s^{(i)} = \frac{\beta}{C}\mu_s(k)[d_s - \mu_s^T(k)W_s^{(i)}] \quad (18)$$

We define scalar  $d_s = \mu_s^T(k) w_s^{(i)}$  as  $u_s^{(i)}$ , thus:

$$\Delta W_s^{(i)} = \frac{\beta}{C} \mu_s(k) u_s^{(i)} \quad (19)$$

Fig. 2 shows Flow diagram of control synthesis process. The relation between the input-output properties of system and its stability has been extensively studied using the theory of dissipative system.

The relation between the input-output properties of system and its stability has been extensively studied using the theory of dissipative system [11]. The relevance is that the FCMAC neural network used for control purposes here will be constructed to have an important dissipation property that makes them robust disturbances and unmodeled dynamics.



**Fig. 2.** Flow diagram of control synthesis process

## 4 Experiments

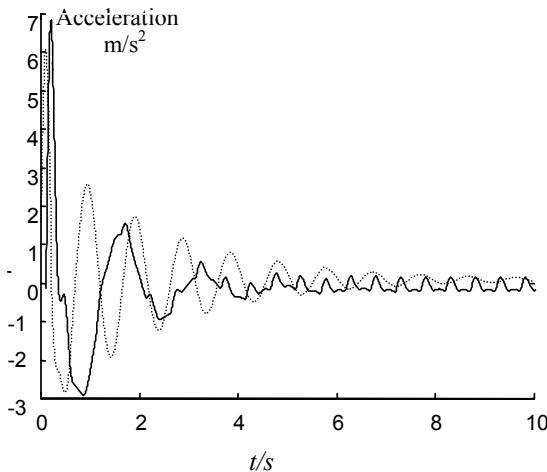
Physical model can be established based on the theories of Buckingham's Pi theorem of section 2, and variables of model are shown in Table 2.

As can be seen easily, the universes of discourse of the controller inputs are closely related to road disturbances, thus in the FCMAC semi-active system, we determine Fuzzy inputs and output as follows.

Controller inputs: vertical sprung mass acceleration, the change of vertical sprung mass acceleration and suspension deflection, controller output is voltage signal which is used to drive the servo valve. We divide each variable into 7 fuzzy subsets. The results of experiment of closed-loop excited by three typical vibration signals: the first is approximately a step input, representing a large-amplitude isolated obstacle, and

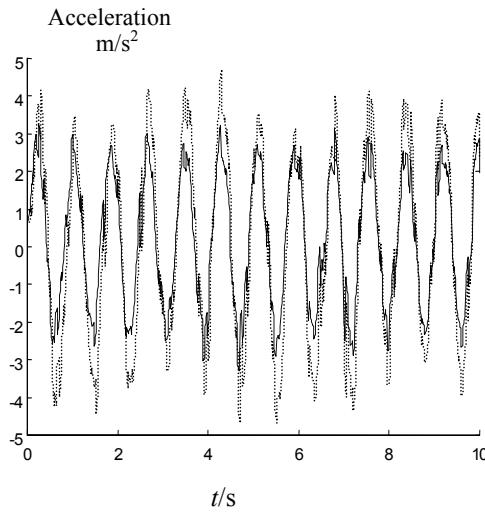
**Table 2.** Variable of physical model

variable	value	variable	value
$m_1$	7 (kg)	$K_r$	$4 \times 10^4$ (m/v)
$m_2$	62 (kg)	$K_q$	0.67 (m <sup>2</sup> /s)
$k_l$	110 (kN/m)	$K_c$	$1.9 \times 10^{10}$ (m <sup>3</sup> /Pas)
$k_2$	35 (kN/m)	$\beta_e$	700 (MPa)
$A_s$	0.0031(m <sup>2</sup> )	$V_t$	0.005 (m <sup>3</sup> )
$u$	0~10 (V)	$C_{sp}$	$6 \times 10^{12}$ (m <sup>3</sup> /Pas)
$\rho$	900 (kg/m <sup>3</sup> )	$x_{vmax}$	0.002 (m)

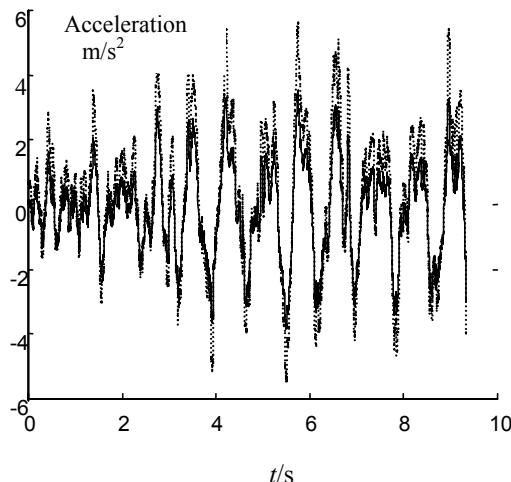
**Fig. 3.** Experiment result of vibration control for a step input

illustrating primary ride control. The second type of vertical road disturbance is a sinusoidal input, the effect of small-amplitude, high frequency inputs can be seen from the sinusoidal road input. The third type of vertical road disturbance is random disturbance input. In order to show that the performance of designed FCMAC semi-active suspension system is good, a comparative study with another scheme is necessary. In this experiment, another semi-active suspension system is designed using PID approach, which has suspension system is designed using PID approach, which has been widely used and been proven good control results in designing semi-active suspension systems. The results for the experiments described are given in Figs. 3-5. In each case, the dashed line shows the response of conventional PID semi-active suspension, the real line shows the FCMAC semi-active suspension system. The result from the step input, Fig. 3 shows that FCMAC semi-active suspension gives an improved heave response in terms of overshoot, although small. Figs. 3-5 show the

sprung mass vertical acceleration with road disturbance of step input, sinusoidal input, and random input in the FCMAC Neural Networks semi-active suspension system. It can be easily noted that the FCMAC Neural Networks semi-active suspension system show better performance in reducing the sprung mass vertical acceleration. Figs. 3-5 show the sprung mass vertical acceleration assuming 25%, 27% and 20% change, greatly improving the ride quality for road disturbance input.



**Fig. 4.** Experiment result of vibration control for a sinusoidal input



**Fig. 5.** Experiment result of vibration control for a random disturbance input

## 5 Conclusions

Extensions to the studies on the vibration absorption technique in semi-active suspension have been presented. Based on a physical model established by a strict technique, online control of semi-active suspension is performed by using a hybrid intelligent control integrated Fuzzy logic technique with CMAC Neural Network. Experiment results show that FCMAC controller described in this paper produced significant coach performance improvements in FCMAC semi-active suspension system. System model is used to offline assistant learning. However, the controller described is only an initial feasibility study with minimal tuning and it nevertheless produced improvements. There is scope for further improvements by extending the controller to the full range of coach motion, including longitudinal. The physical model described is only included as a realistic representation of the suspension system and allowed road inputs via the vertical tyre force. The model should then be extended to the full-car model by including the steering system, with driver steering wheel inputs via the rack motion, and the representation of the lateral tyre force.

## References

1. Mo Jamshidi: Fuzzy Control of Complex Systems. Soft Computing, Springer-Verlag (1997) 42-56
2. Chian-Shyong Tseng: Integrating Fuzzy Knowledge by Genetic Algorithms. IEEE Transactions on Evolutionary Computation 2(4) (1998) 138-149
3. Takao Sato, Akira Inoue & Yoichi Hirashima: Self-Tuning Two-Degree-of-Freedom PID Controller Reducing the Effect of Disturbance, Proc. American Control Conf., American (2002) 3997-4002
4. Daniel Neagu: Modular Neuro-fuzzy Networks Used in Explicit and Implicit Knowledge Integration, Proc. 15th International Conf. On Florida Artificial Intelligence Society, Florida, (2002) 277-281
5. Buswell, R., Angelo, P., Wright J.: Transparency and Simplification of Fuzzy Rule-based Models for On-line Adaptation, Proc. 2nd EUSFLAT Conf. Leicester (2001) 234-237
6. Yaochu Jin: A Framework for Evolutionary Optimization with Approximate Fitness Functions. IEEE Transactions on Evolutionary Computation 6(5) (2002) 481-494.
7. Markus Olhofer, Toshiyuki Arima, Yaochu Jin, Toyotaka Sonoda and Bernhard Sendhoff: Optimization of Transonic Gas Turbine Blades with Evolution Strategies, Honda R&D Technical Review 14(1) (2002) 203-216
8. Kim, M. Y., Cho H. S., Kim J. H.: Neural Network Based Recognition of Navigation Environment for Intelligent Shipyard Welding Robots, IROS, Maui, Hawaii, USA (2001)
9. Jeong, I. S., Cho, H.S.: Self-localization for Mobile Robots by Matching of Two Consecutive Environmental Range Data, Proc. of the 2001 IEEE International Conf. on Robotics and Automation (2001) 1603-1608
10. Chun-ShinLin: Learning Convergence of CMAC Technique. IEEE Transactions on Neural Networks 8(6) (1997) 1281-1291
11. Jong H. Park: H-Infinity Direct Yaw Moment Control with Brakes for Robust Performance and Stability of Vehicles. JSME International Journal, Series C 44(2) (2001) 403-413

# An Intelligent Conversational Agent as the Web Virtual Representative Using Semantic Bayesian Networks

Kyoung-Min Kim, Jin-Hyuk Hong, and Sung-Bae Cho

Dept. of Computer Science, Yonsei University  
134 Shinchon-dong, Seodaemun-ku, Seoul 120-749, Korea  
{kminkim, hinh}@sclab.yonsei.ac.kr, sbcho@cs.yonsei.ac.kr

**Abstract.** In this paper, we propose semantic Bayesian networks that infer the user's intention based on Bayesian networks and their semantic information. Since conversation often contains ambiguous expressions, managing the context or the uncertainty is necessary to support flexible conversational agents. The proposed method drives the mixed-initiative interaction (MII) that prompts for missing concepts and clarifies for spurious concepts to understand the user's intention correctly. We have applied it to an information retrieval service for Web sites so as to verify the usefulness.

## 1 Introduction

Conversational agents are representative intelligent agents that provide information for users by using the natural language dialogue. They understand the user's intention through conversation and offer an appropriate service [1]. Pattern matching, one of the popular methods for constructing the conversational agent, works well at a sentence, but it is not feasible to understand a dialogue in which context should be considered. Moreover, it is likely to fail to understand a complex sentence which requires a deep analysis. Recently, researchers have investigated on flexible dialogue models using Bayesian networks (BN) [2], where Bayesian networks have been also used in information retrieval (IR) [3,4].

When many variables in the application domain are related to each other, the inference of the user's intention becomes very difficult. In this paper, we propose semantic Bayesian networks (SeBN) not only to reduce the complexity of construction, but also to infer the user's intention in more detail.

## 2 Intelligent Conversational Agent

For the efficient inference of conversational agents, we design semantic Bayesian networks composed of the probabilistic inference and the semantic inference. The stepwise modeling helps to understand the user's intention through conversation.

It is constructed with three levels according to the function: keywords, concepts, and targets. The keyword layer consists of words related to the user's query, while the concept layer is composed of entities of the domain and their semantic relationship.

The target layer represents target information (products) whose attributes are defined. The concept layer is divided to three components: objects, attributes, and values. Each object is a set of attribute-value pairs, where node  $a_i$  is an attribute and node  $v_k$  is a value in the domain. A solid line represents the probabilistic relationship between nodes, while a dotted line signifies the semantic relationship between them.

The probabilistic relationship in semantic Bayesian networks is similar to that in the traditional IR model. First, it infers probabilistically between the keyword layer and the concept layer. The user's query  $U = \{k_1, k_2, \dots, k_t\}$ , where keyword  $k_i$  is interpreted as an elementary word in the keyword layer. It sets a keyword node as 1 when the word of the keyword layer is observed in query  $Q$  and otherwise 0.

It infers the probability of each node in the concept layer when all evidence variables associated with keywords are set. The probability  $P(c|W)$ , using with keyword  $W$  in the keyword layer as evidence, is defined as follows:

$$\begin{aligned} P(c|W) &= P(c|w_1, w_2, \dots, w_N) \\ &= \frac{P(c) \times P(w_1, w_2, \dots, w_N | c)}{P(w_1, w_2, \dots, w_N)} \\ &\approx P(c) \times P(w_1, w_2, \dots, w_N | c) \\ &= P(c) \times P(w_1 | c) \times P(w_2 | c) \times \dots \times P(w_N | c) \\ &= P(c) \prod_{i=1}^n P(w_i | c) \end{aligned}$$

After computing the probability of all nodes in the concept layer, it infers the probability  $P(p|C)$  of product  $p$  in the target layer using them as evidence similar to that of inferring the probability  $P(c|W)$ .

**Table 1.** Semantic inference in SeBN

<p>[Concept] Object : <math>O = \{o_1, o_2, o_3, \dots, o_n\}</math></p> <p>Attribute: <math>A = \{a_1, a_2, a_3, \dots, a_n\}</math></p> <p>Value : <math>V = \{v_1, v_2, v_3, \dots, v_n\}</math></p> <pre> <i>i</i> = find_high_probability_object(); // Search an object over the threshold. if ( object(<i>o</i>) &gt; <math>\alpha</math> ) {     <i>j</i> = find_OA_attribute(<i>o</i>); // Search attribute '<i>a</i>' whose probability is below the threshold which has <i>O-A</i> relationship with node <i>o</i>.     if ( attribute(<i>a</i>) &lt; <math>\beta</math> )         response (<i>a</i>, <i>v</i>);     else         reject; } </pre>	<pre> else {     <i>j</i> = find_high_probability_attribute(); // Search attribute '<i>a</i>' over the threshold     if ( attribute(<i>a</i>) &gt; <math>\alpha</math> ) { // Search attribute '<i>a</i>' whose probability is below the threshold which has <i>O-A</i> relationship with node <i>o</i>.         <i>i</i> = find_OA_object(<i>o</i>);         response(<i>o</i>);     }     else reject; } </pre>
---	---

It selects a node in the target layer whose probability is higher than the threshold after the inference. It provides the information of the target product to the user when a proper number of nodes are selected. In this paper, we define it as successful execution

when one product is selected. When there is no product selected, it executes the semantic inference of semantic Bayesian networks in the concept layer. There are two major relationships ('Has-a', 'Is-a') between nodes while 'Is-a' has two different types ('O-A', 'A-V'). Table 1 shows the semantic inference executed when the probabilistic inference fails to infer the user's intention. At first, it searches an object node whose probability is higher than the threshold. Then, it looks up an attribute whose probability is below the threshold, which has 'O-A' relationship with the object node. It collects supplementary information on the attribute selected and carries out the inference again with the information gathered from the user. It repeats the procedure until a target product is selected.

In order to search out what the user wants, it should gather enough information to infer target products. Traditional information retrieval systems work well only when the user's query includes enough for inference. When there is not enough information, however, the proposed method provides a suitable response to the user based on the mixed-initiative interaction. Finally, the proposed method is able to show good performance in diverse dialogue situations.

### 3 Experimental Results

#### 3.1 Experimental Environments

In order to verify the usefulness of the proposed method, we have developed a flexible conversational agent for virtual representative of web sites. It consists of a main window for displaying information, an input text box, and the avatar system with a speech generation engine. When the user types a query, the avatar responds in speech with a corresponding action. Q-avatar ([www.qavatar.com](http://www.qavatar.com)) is employed as the avatar system, while Voiceware ([voiceware.co.kr](http://voiceware.co.kr)), a solution for speech generation, is used to provide the user with a realistic and convenient interface.

**Table 2.** The attributes of objects

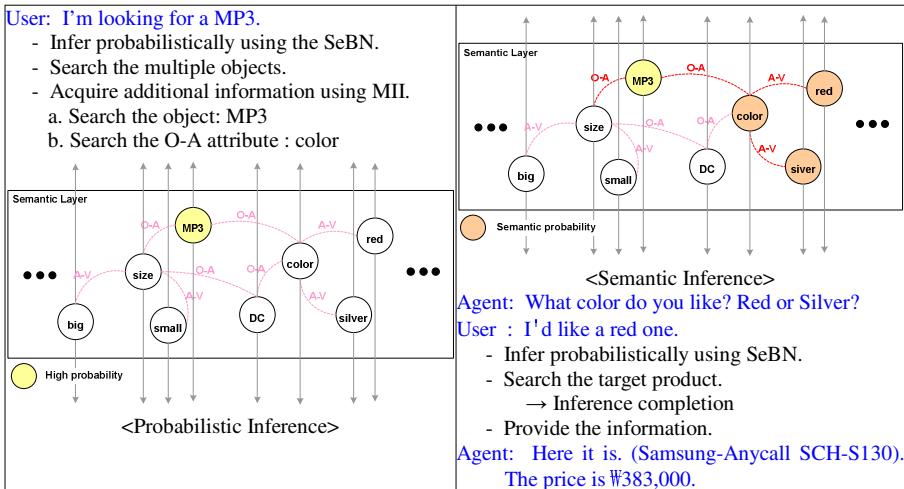
Object	Attributes
Cellular-phone (240 products)	Brand, Product, Model, Image, Bell, Camera, Pixel, Size, Weight, Color, Price, Year
Digital camera (688 products)	Brand, Product, Model, Image, Memory, Run-time, Color, Size, Feature, Weight, Price, Year
MP3 (488 products)	Brand, Product, Model, Image, Pixel, Memory, Weight, Feature, Size, Zoom, Color, Price, Year

The target domain is mobile Web sites introducing of cellular-phone, digital camera, and MP3. Table 2 describes the attributes of each object in the target database. The database is built by extracting information from 5 Web sites: Naver ([www.nshopping.naver.com](http://www.nshopping.naver.com)), Samsung-mall ([www.samsung-mall.co.kr](http://www.samsung-mall.co.kr)), LG-eshop ([www.gseshop.co.kr](http://www.gseshop.co.kr)), Enuri ([www.enuri.com](http://www.enuri.com)) and DCinside ([www.dcinside.com](http://www.dcinside.com)).

### 3.2 Qualitative Analysis: Illustration

In many cases, the user has background knowledge in addition to the content of conversation, so a query may not include all information required to infer the user's intention. The proposed conversational agent uses a mixed-initiative dialogue as shown in Fig. 1, by requesting additional information to the user. Finally, information on the target product is provided to the user after inference with the information.

As shown in Fig. 1, it searches the plural objects from the initial query. Since the agent needs additional information for the correct intention inference, it outputs a supplementary query, "What color do you like? Red or Silver?" to the user as the mixed-initiative interaction. The user responds that "I'd like a red one," and then it executes the probabilistic inference again using semantic Bayesian networks based on this response. Until it detects the plural products as the result of prior inference, the agent keeps up the conversation by the mixed-initiative interaction. If a product is selected, the agent finishes the inference and provides the information of the target product to the user.



**Fig. 1.** The target retrieval using MII

### 3.3 Quantitative Analysis

In order to evaluate the efficacy and satisfaction of the agent by younger adults, we have compared three conversational agents: Script-based, BN-based and the SeBN-based agents. Thirty subjects in age from 22 to 33 living in Korea have evaluated them. Participants have performed ten tasks to search for information on several products. Satisfaction scores have been measured by single item on five-point Likert scales (1.0= "not at all", 5.0="very much").

The result (see Table 3) shows that the proposed method ( $M=94.42$ ) is superior to the others ( $M=92.15, 87.51$ ). It is able to manage various types of dialogues while the

Script-based and BN-based agents fail to respond for them. It also shows good performance in providing suitable responses for the user with few interactions ( $M=2.96$ ).

As shown in Table 4, satisfaction with the overall intervention has been very high in the proposed method. The effect of the proposed method on easy, friendly, informative, repetitive and interesting was statistically measured by a one way ANOVA with the variant of the SeBN as the among-systems factor. Post-hoc tests were also conducted, whenever one or more of the significant factors entail more than two categories or levels. At most cases, the proposed method shows better results than the others.

**Table 3.** Comparative results in efficiency

Retrieval Rate (PR) Average Interactions (AI)	Script		BN		SeBN	
	PR(%)	AI	PR(%)	AI	PR(%)	AI
Average	87.51	3.53	92.15	3.18	94.42	2.96

**Table 4.** Comparative results in the user satisfaction

User Satisfaction	Script		BN		SeBN	
	Mean	SD	Mean	SD	Mean	SD
Easy	2.9	.7379	4.0	.4714	4.6	.5164
Friendly	2.7	.6749	3.8	.4216	4.7	.4830
Informative	3.1	.5676	3.7	.6749	4.4	.5164
Repetitive	3.9	.8756	2.3	.4830	1.6	.5164
Interesting	3.1	.5676	3.8	.7888	4.5	.5270

## 4 Conclusion and Future Works

We have proposed a conversational agent using semantic Bayesian networks to be more flexible and considerable in the inference of the intention. If information in the query is insufficient, the agent asks the user to give more information to infer the user's intention correctly. Finally, it improves the answering performance. While, designing networks becomes easier and more comprehensible by one's intuition. The research on the automatic construction of semantic networks is remained for the future study.

## Acknowledgements

The work was supported by the Korea Research Foundation Grant. (KRF-2004-005-H00005)

## References

1. S. Macskassy, "A conversational agent," Master Essay, Rutgers University, 1996.
2. J.-H. Hong and S.-B. Cho, "A two-stage Bayesian network for effective development of conversational agent," LNCS 2690, pp. 1-8, 2003.
3. S. Acid, et al., "An information retrieval model based on simple Bayesian networks," Int. Journal of Intelligent Systems, 18(2), pp. 251-265, 2003.
4. P. Calado, et al., "A Bayesian network approach to searching Web databases through keyword-based queries," Information Processing and Management, 40, pp. 773-790, 2004.

# Three-Tier Multi-agent Approach for Solving Traveling Salesman Problem

Shi-Liang Yan<sup>1</sup> and Ke-Feng Zhou<sup>2</sup>

<sup>1</sup> Engineering & Technology Center, Southwest University of Science and Technology,  
Mianyang, 621010, P.R. China  
[bmwbenz@163.com](mailto:bmwbenz@163.com)

<sup>2</sup> Department of computer science, Qiu-Ai Experimental Elementary School,  
Ningbo, 315153, P.R. China  
[zhoukf991@163.com](mailto:zhoukf991@163.com)

**Abstract.** The Traveling Salesman Problem (TSP) is a very hard optimization problem in the field of operations research. It has been shown to be NP-hard, and is an often-used benchmark for new optimization techniques. This paper will bring up a three-tier multi-agent approach for solving the TSP. This proposed approach supports the distributed solving to the TSP. It divides into three-tier (layer), the first tier is ant colony optimization agent and its function is generating the new solution continuously; the second-tier is genetic algorithm agent, its function is optimizing the current solutions group; and the third tier is fast local searching agent and its function is optimizing the best solution from the beginning of the trial. Ultimately, the experimental results have shown that the proposed hybrid approach has good performance with respect to the quality of solution and the speed of computation.

## 1 Introduction

The usual ways of solving the traveling salesman problem (TSP) are based either on integer linear programming techniques or on heuristic algorithms [1]. The former approach pursues the solution of the problem up to optimality. For example, highly optimized exact algorithms based on the branch-and-cut method [2] have been proposed that enable even large TSP instances to be solved. Unfortunately, this is not always possible because of the increase in computational work with problem size. Some heuristic approaches, however, have been proved to be very effective both in terms of execution times and quality of the solutions achieved. Domain-specific heuristics, such as 2-Opt [3], 3-Opt [4], and Lin-Kernighan (LK) [5], are surprisingly very effective for the TSP. On the other hand, general problem-independent heuristics like simulated annealing (SA) [6], genetic algorithms (GA) [7], ant system (AS) [8] and Neural Network (NN) [9] perform quite poorly on large TSP instances.

Several published results demonstrate that combining a problem-independent heuristic with a local search method is a viable and effective approach for finding high-quality solutions of large TSPs [10]. The problem-independent part of the hybrid algorithm drives the exploration of the search space, thus, focusing on the global

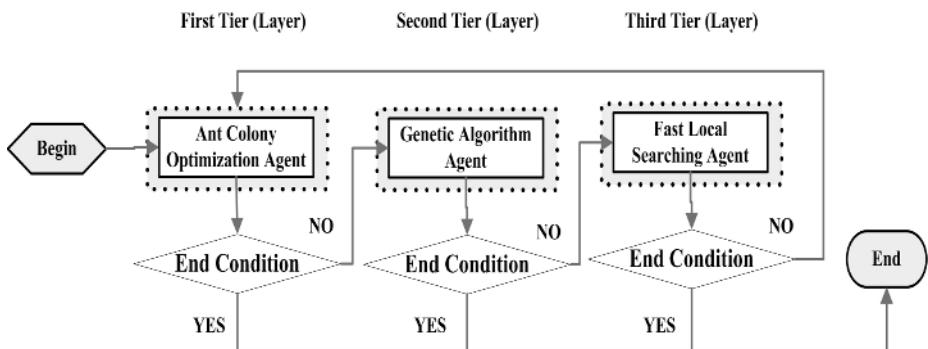
optimization task, while the local search algorithm visits the promising sub-regions of the solution space. Reference [11] proposed the chained local optimization algorithm, where a special type of 4-Opt move is used under the control of a SA schema to escape from the local optima found with LK. Reference [12] combines an original compact genetic algorithm with an efficient implementation of LK.

Since the TSP has proved to belong to the class of NP-hard problem, heuristics and meta-heuristics occupy an important place in the methods so far developed to provide practical solutions for large instances. Accordingly, this paper proposes a three-tier multi-agent approach for solving the TSP. This proposed approach supports the distributed solving to the TSP. The experimental results have shown that the proposed hybrid approach has good performance with respect to the quality of solution and the speed of computation.

## 2 Three-Tier Multi-agent Approach

### 2.1 Three-Tier Multi-agent Framework

Figure 1 depicts the framework of the proposed three-tier multi-agent approach. From figure 1, we can see that the proposed approach includes three-tier (layer), the first tier is ant colony optimization agent and its function is generating the new solution continuously; the second-tier is genetic algorithm agent, its function is optimizing the current solutions group; and the third tier is fast local searching agent and its function is optimizing the best solution from the beginning of the trial. Obviously, this proposed approach supports the distributed solving to the TSP. The proposed approach is terminated when one of the following criteria (end condition of whole approach) is satisfied: (1) maximum preset search time is exhausted; (2) the known optimal solution was achieved by this proposed approach.



**Fig. 1.** The framework of the proposed three-tier multi-agent approach

### 2.2 Ant Colony Optimization Agent

The first tier of this proposed approach is ant colony optimization (ACS) agent and its function is generating the new solution continuously. The ACS agent is terminated

when one of the following criteria is satisfied: (1) the known optimal solution was achieved by ACS agent; (2) all of the solutions generated in five continuous iterations are worse than the globally best tour from the beginning of the trial. When the current optimization in the first tier met the end condition of ACS agent but not satisfied the end condition of whole approach, it will use genetic algorithm (GA) agent to improve the quality of those solutions achieved by ACS agent. That is, if the final solutions achieved by ACS agent aren't satisfied the end condition of whole approach, then these solutions will be selected as the initial chromosomes population of GA agent.

### **2.3 Genetic Algorithm Agent**

The second-tier of this proposed approach is genetic algorithm agent and its function is optimizing the current solutions group. The GA agent is terminated when one of the following criteria is satisfied: (1) all individuals of a population are identical; (2) all of the solutions generated in five continuous iterations are worse than the globally best tour from the beginning of the trial; (3) maximum preset generation is reached; (4) the known optimal solution was achieved by GA agent. When the current optimization in the second tier met the end condition of GA agent but not satisfied the end condition of whole approach, it will use fast local searching agent to refine these improved solutions achieved by GA agent. That is, if the final solutions achieved by GA agent aren't satisfied the end condition of whole approach, then these solutions (or part of them, such as top ten) will be selected as the initial optimization individuals of fast local searching agent.

### **2.4 Fast Local Searching Agent**

The third tier of this proposed approach is fast local searching (FLS) agent and its function is optimizing the best solution from the beginning of the trial. The FLS agent is terminated when one of the following criteria is satisfied: (1) maximum preset search time is exhausted; (2) the known optimal solution was achieved by FLS agent. If the final solutions achieved by FLS agent aren't satisfied the end condition of whole approach, then it will use ACS agent to generate some new more promising solutions. Please note, ACS agent should apply the global updating rule to update the pheromone level to the refined solutions achieved by FLS agent. In the whole optimization process of this proposed approach, the globally best solution from the beginning of the trial will be recorded because of the randomness of the ACS agent and GA agent.

## **3 Simulation Study**

In this section, the proposed method was tested using a set of the benchmark TSP. To avoid any misinterpretation of the optimization results, relating to the choice of a particular initial solution, we performed each test 20 times. The performance of the proposed technology is compared to the other four published versions optimization algorithm (see table 1). The experimental results obtained for 8 test problems, using these five different methods, are given in table 2. This proposed method was implemented on a Pentium IV 2.4 GHz personal computer with a single processor and

**Table 1.** The 5 different methods used in this section

Mark	Name of the optimization algorithm	Reference
M1	Annealing-Based Heuristics	Reference [13]
M2	Guided Local Search	Reference [14]
M3	Lin-Kernighan Heuristic	Reference [15]
M4	Evolutionary Algorithm	Reference [16]
M5	Three-tier Multi-agent Approach	This paper

**Table 2.** The experimental results obtained from 8 test problems using the 5 methods

TSP	Performance Index	M1	M2	M3	M4	M5
ATT532 (42029)	Average time (S)	20.93	17.01	17.65	16.68	15.07
RAT783 (8806)	Average error (%)	0.016	0.014	0.013	0.013	0.010
PR1002 (259045)	Average time (S)	55.06	44.74	46.43	43.89	39.65
VM1084 (239297)	Average error (%)	0.022	0.020	0.019	0.018	0.014
PCB1173 (56892)	Average time (S)	129.58	105.28	109.26	103.28	93.31
U1432 (152970)	Average error (%)	0.027	0.024	0.023	0.022	0.017
U2152 (64253)	Average time (S)	114.49	93.02	96.54	91.25	82.45
PR2392 (378032)	Average error (%)	0.033	0.029	0.027	0.027	0.021
U1432 (152970)	Average time (S)	119.68	97.24	100.92	95.39	86.19
U2152 (64253)	Average error (%)	0.025	0.022	0.021	0.020	0.016
U1432 (152970)	Average time (S)	150.03	121.89	126.51	119.58	108.04
U2152 (64253)	Average error (%)	0.049	0.044	0.042	0.040	0.031
U2152 (64253)	Average time (S)	299.20	243.09	252.30	238.49	215.47
PR2392 (378032)	Average time (S)	293.59	238.53	247.56	234.01	211.43
PR2392 (378032)	Average error (%)	0.056	0.050	0.047	0.046	0.035
PR2392 (378032)	Average time (S)	0.077	0.068	0.065	0.063	0.049

512M RAM. Here, the average optimization time and the average optimization error were used for evaluating these different methods. Average optimization time is the average time of the computation time of these 20 independent runs. In the same way, average optimization error is the average error of the computation error of these 20 independent runs. The computation error is the relative error between the optimal result achieved by giving method and the optimal result produced by the recent published heuristic optimization algorithm. From table 2, we can see that both the average optimization time and the average optimization error, the proposed method of this paper are better than the other four methods. Simulations have shown that the proposed hybrid approach for the TSP has excellent performance with respect to the quality of solutions and the speed of calculation.

## 4 Conclusions

The contribution of this paper is summarized as following. (1) This paper presents a new multi-agent architecture for the TSP; the future new agent can be integrated into this architecture. (2) This study designs three-tier multi-agent framework with different

function in the proposed approach, all of these agents cooperate each other to find an optimal solution.

## References

1. LAPORTE G.: The Traveling Salesman Problem: an Overview of Exact and Approximate Algorithms. *European Journal of Operational Research*. 59(3) (1992) 231-247
2. PADBERG M., RINALDI G.: Optimization of a 532-city Symmetric Genetic Traveling Salesman Problem by Branch and Cut. *Operational Research Letters*. 6(1) (1987) 1-7
3. CROES G.A.: A Method for Solving Traveling Salesman Problems. *Operational Research*. 6(6) (1958) 791-812
4. LIN S.: Computer Solution of the Traveling Salesman Problem. *Bell System Technology Journal*. 44(5) (1965) 2245-2269
5. LIN S., KERNIGHAN B.W.: An Effective Heuristic Algorithm for the Traveling Salesman Problem. *Operational Research*. 21(2) (1973) 498-516
6. MARTIN O., OTTO S.W.: Combining Simulated Annealing with Local Search Heuristic. *Annual Operational Research*. 63(2) (1996) 57-75
7. MICHALEWICZ Z.: *Genetic Algorithms + Data Structures = Evolution Programs*. 3rd edn. Springer-Verlag, Berlin Heidelberg New York (1996)
8. DORIGO M., GAMBARDELLA L.M.: Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem. *IEEE Transactions on Evolutionary Computation*. 1(1) (1997) 53-66
9. ARAS N., ALTINEL I.K., OOMMEN J.: A Kohonen-Like Decomposition Method for the Euclidean Traveling Salesman Problem-KNIES\_DECOMPOSE. *IEEE Transactions on Neural Network*. 14(4) (2003) 869-890
10. BURIOL L., FRANCA P.M.: A New Memetic Algorithm for the Asymmetric Traveling Salesman Problem. *Journal of Heuristics*. 10(3) (2004) 483-506
11. MARTIN O., OTTO S.W., FELTEN E.W.: Large Step Markov Chain for the Traveling Salesman. *Journal of Complex System*. 5(3) (1991) 299
12. BARAGLIA R., HIDALGO J.I., PEREGO R.: A Hybrid Heuristic for the Traveling Salesman Problem. *IEEE Transactions on Evolutionary Computation*. 5(6) (2001) 613-622
13. PEPPER J.W., GOLDEN B.L., WASIL E.A.: Solving the Traveling Salesman Problem with Annealing-Based Heuristics: A Computational Study. *IEEE Transactions on Systems, Man and Cybernetics - Part A: Systems and Humans*. 32(1) (2002) 72-77
14. VOUDOURIS C., TSANG E.: Guided Local Search and its Application to the Traveling Salesman Problem. *European Journal of Operational Research*. 113(2) (1999) 469-499
15. HELSGAUN K.: An Effective Implementation of the Lin-Kernighan Traveling Salesman Heuristic. *European Journal of Operational Research*, 126 (1) (2000) 106-130
16. TSAI H.K., YANG J.M., TSAI Y.F., et al.: An Evolutionary Algorithm for Large Traveling Salesman Problems. *IEEE Transactions on Systems, Man and Cybernetics*. 34(4) (2004) 1718-1729

# Adaptive Agent Selection in Large-Scale Multi-Agent Systems

Toshiharu Sugawara<sup>1</sup>, Kensuke Fukuda<sup>2</sup>, Toshio Hirotsu<sup>3</sup>,  
Shin-ya Sato<sup>4</sup>, and Satoshi Kurihara<sup>5</sup>

<sup>1</sup> NTT Communication Science Laboratories,

3-1 Wakamiya Morinosato, Atsugi,  
kanagawa 243-0198, Japan

<sup>2</sup> National Institute of Informatics

<sup>3</sup> Toyohashi University of Technology

<sup>4</sup> NTT Network Innovation Laboratories

<sup>5</sup> Osaka University

**Abstract.** An agent in a multi-agent system (MAS) has to select appropriate agents to assign tasks. Unfortunately no agent in an open environment can identify the states of all agents, so this selection must be done according to local information about the other known agents; however this information is limited and may contain uncertainty. In this paper we investigate how overall performance of MAS is affected by learning parameters for adaptive strategies to select partner agent for collaboration. We show experimental results using simulation and discuss why overall performance of MAS varies.

## 1 Introduction

A huge number of agents are deployed in the vast Internet. Some agent work together cooperatively or competitively to provide many services. In general an agent selects partner agents appropriate for collaboration based on their abilities. However, if multiple candidate agents still remain, a more efficient agent is preferable. Of course, the efficiency of these agents are determined from agents' states such as workload, communication bandwidth as well as their intrinsic capabilities, such as CPU power. In an open environment like the Internet, however, agents have to select efficient partners based only on locally available information, which contains uncertainties.

Our interest lies in the total performance of MAS, when all agents select partner agents based on their partner selection strategies (PSS), where 'total' means the average performance of all agents. For this issue, we already investigated how PSS with learning can gradually improve total performance. This improvement is achieved by load balancing when workloads are high and by concentration when low[4].

This paper investigates how such learning parameters as fluctuation related to exploitation-versus-exploration will affect total performance and load balancing. As a performance parameter we use response time from sending a task to returning the results. We also introduce a simple coordination strategy to improve the total performance of MAS.

## 2 Simulation and Its Model

Our simulation model consists of a set of agents  $A = \{a_i\}$  and server agents  $S = \{s_j\} (\subset A)$  that can execute a specific task  $T$ . When agent  $a_i$  has  $T$ , it assigns it to a server that it knows,  $S_i (\subset S)$ . A PSS corresponds to the method whereby  $a_i$  selects  $s_j$  from  $S_i$ . To clearly understand the relationship between total performance and PSSs, we simplify the other parameters and assume that all tasks are only of a single type.

On the Internet, one observable parameter concerning performance is response time ( $rt$ ), time between the request of a task and the return of the result. Other parameters such as CPU types are not usually available, they are not used in our experiments. Using the observed data, agents learn which server's  $rt$  will be the smallest. In our experiments,  $a_i$  calculates expected response time  $e_i^j$  of known server  $s_j$  and (usually) selects the best server,  $\arg \min_{s_j \in S_i} e_i^j$ . To evaluate the total performance of a MAS, we adopt the average value of response time of all agents. This value is denoted by  $RT$ . Our experiments show how  $RT$  changes when all agents try to make their  $rt$  smaller.

Let  $|A| = 10000$  and  $|S| = 120$ . All agents are randomly placed on points of a 150 x 150 grid plane with torus and Manhattan distance topology. For every tick  $tl$  tasks are generated and given to  $tl$  randomly selected agents.  $tl$  is called the *task load* denoted by  $tl$  task/tick or  $tl$  T/t. Agent  $a_i$  receiving a task selects  $s_j \in S_i$  using its PSS to send it to  $s_j$ . Server  $s_j$  processes it and returns the result. Agent  $a_i$  observes the response time and calculate expected response time  $e_i^j$ .

Servers are assumed to have their own CPU capabilities; each server processes a task in 10 to 50 ticks. These capabilities are randomly assigned. When a task arrives at  $s_j$ , it is immediately executed if  $s_j$  has no other tasks. If  $s_j$  has others, the received task is stored in its queue and queued tasks are processed in turn. An agent can store 20 tasks in its queue. If  $a_i$  already has 20 tasks in its queue, the new task is dropped.

Communication cost, the time to send a task, is assumed to be proportional to the distance between the agent and server and ranges from 10 to 120 msec. An agent has its scope based on distance (less than 14), so it can communicate with agents and servers in its scope. For all agents to compare server response times, they have to know at least two servers. If an agent knows less than one server, it asks all known agents for servers; then in the following experiments agents can initially know two to fifteen servers.

The results of all our simulations are the average value of three independent experiments from three series of random numbers using three seeds. In these experiments, the total sum of the capabilities expressed by all agents is that they can theoretically process 4.7 to 5.0 tasks every tick. The actual capabilities are influenced by communication cost, the deviation of task allocation, and server distribution in the grid plane. Note that response time is the sum of the durations for communication, queuing, and processing.

Agents select appropriate servers using expected response time  $e_i^j$ , which is calculated by *average values* of observed response time  $h_i^j$  or estimated values by *update function*  $w_i^j$  that are often used in reinforcement learning:

$$h_i^j[n] = h_i^j[n - 1] * (1 - 1/n) + rt_i^j[n] * 1/n \quad (1)$$

$$w_i^j[n] = \begin{cases} w_i^j[n - 1] * (1 - \lambda) + rt_i^j[n] * \lambda & (\text{if } n > 1) \\ rt_i^j[1] & (\text{if } n = 1), \end{cases} \quad (2)$$

where  $rt_i^j[n]$  is the  $n$ -th observed response time when  $a_i$  sent the task to server  $s_j$  and  $\lambda$  is the learning parameter ( $0 \leq \lambda \leq 1$ .  $\lambda = 0.2$  in our experiments). Value  $h_i^j[n]$  ( $w_i^j[n]$ ) is  $a_i$ 's expected response time about  $s_j$  by Eq. 1 (Eq. 2) after  $n$  data,  $r_i^j[1], \dots, r_i^j[n]$ , of  $s_j$  were observed. We describe  $h_i^j$  and  $w_i^j$  simply if value  $n$  is unnecessary. Note that,  $\lim_{n \rightarrow \infty} 1/n = 0$ , so  $h_i^j$  becomes stable, although  $w_i^j$  may change according to the server's performance. Hereafter, Eq. 1 is called the *average value function*.

### 3 Performance Improvement by Learning

In the first experiment (Exp. 1),  $a_i$  selects a server from  $S_i$  by the following PSS:

- P1.  $a_i$  selects server  $\arg \min_{s_j \in S_i} e_i^j$  with a probability of  $p$  ( $0 \leq p \leq 1$ ). If multiple servers have the best  $e_i^j$ , one is randomly selected.
- P2. Otherwise,  $a_i$  selects the server with probabilistic distribution  $Pr(s_j)$ ,

$$Pr(s_j) = (e_i^j)^{-l} / \sum_{s_k \in S_i} (e_i^k)^{-l}$$

where  $e_i^j = h_i^j$  or  $w_i^j$ . Agent  $a_i$  initially sets  $e_i^k = 0$  for known server  $s_k$  so it selects  $s_k$  with no observed data. Note that P2 induces *fluctuation* to the PSS, since an agent may select the best server. The larger  $l$  is, the smaller this fluctuation is.

First we assume  $t=4T/t$ . Overall response time  $RT$  is calculated every 20K ticks and the average value of these  $RT$  values during 600 K and 800 K are shown in Fig. 1; this illustrates how independent learning by each agent can uprate the total performance of the entire MAS, probably by balance load. Note that since the first observed values of  $RT$ , that is, during 0K to 20K ticks, are 272.9 ticks and 11180.0 tasks, respectively, they can be considerably improved.

When using update function ( $e_i^j = w_i^j$ ), Fig. 1(a) indicates that  $RT$  generally improves if  $l$  is larger ( $p = 0.9$  or  $0.8$ , fixed); this result is consistent with the results of the fixed load case in [2] (In [2]  $p = 0$ , so graphs (ii) in Fig. 1(a) are closer to their experiments). However, when using the average value function ( $e_i^j = h_i^j$ ),  $RT$  is the best

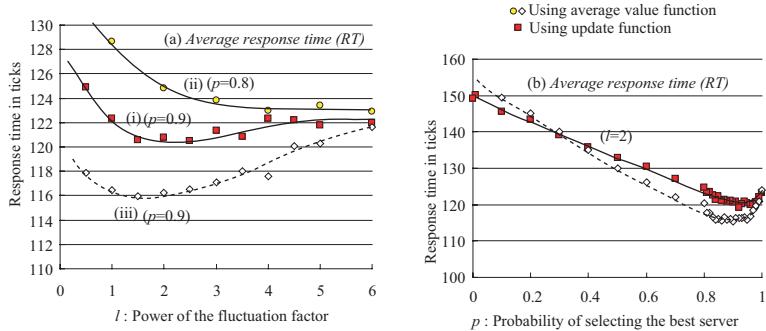


Fig. 1. Total performance values

around  $1.5 \leq l \leq 2$ . This means that some fluctuation can improve total performance. This suggests that using the average value function makes agent server selections stable and conservative over time, but the convergent state is not optimal.

We investigate how  $p$  affects total performance by fixing  $l$  to 2.0 since  $RT$  is the best around  $p = 2.0$ . This result is shown in Fig. 1 (b).  $RT$  values are minimal around  $p = 0.85$  to 0.9 in both cases. This graph shows that the observed tradeoff in the previous experiment is a special phenomenon; when  $p \leq 0.8$ , agents select the best servers by learning with information from their own viewpoints, so their performance value  $RT$  indicates improvement when  $p$  becomes larger. However if  $p > 0.9$ , this PSS induces task concentration to a few servers that have high CPU capabilities. Consequently, both  $RT$  degrades. This simulation shows that tradeoff appears only when  $0.8 \leq l \leq 0.9$ , which is a kind of intermediate state.

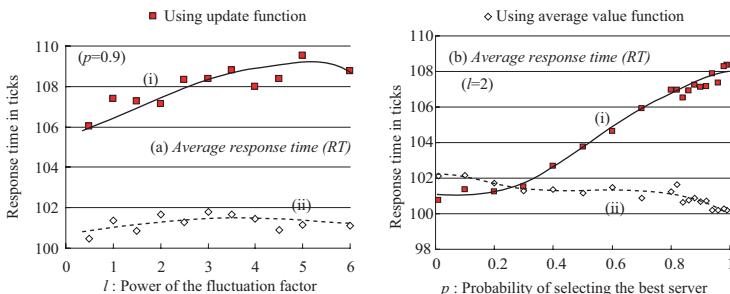
Finally, the most noticeable feature of these graphs is that using the average value function outperforms update functions in  $RT$ . When agents use update functions, they can act more adaptively to the environment including other agents. Hence this adaptability also continuously changes the environment. Agents can detect these changes only by observing worse response time and never be stable; so the total performance do not improve. However, using average values, agent selections become stable and conservative. This convergent state is not optimal but it is better than the unstable situation.

## 4 Communication with Local Agents

In [4], we suggested that a *collaborative strategy* (CS) can drastically improve total performance, but fluctuation degrades it, in contrast to the experiment in the previous section. In the next experiment (Exp. 2), we investigate in more detail how performance value  $RT$  changes when agents collaborate.

For this purpose, the following CS is inserted before P1.

- P0. With a probability of 0.01,  $a_i$  randomly selects a known agent, and asks it to recommend a ‘best server’  $s_n$ . If  $a_i$  already knows  $s_n$ , this is ignored. If not,  $a_i$  adds  $s_n$  and set  $e_i^n = 0$ .



**Fig. 2.** Total performance values ( $p = 0.9$ )

The requested agent recommends the server which is selected by P1 and P2. P0 suggests that a good server of neighboring agents may be good for itself. This CS requires one tick for communication; so their communication cost is relatively small because an agent only communicates with others within its scope, and this recommendation occurs fairly infrequently (1%). Additionally only “best” (often “better”) servers are recommended; so low performance servers are filtered out. Note that this CS is open, that is, a good server in the distance may be delivered by the agent-by-agent recommendation chain.

The results of Exp. 2 are illustrated in Fig. 2. Both figures show that the value of  $RT$  worsens if  $l$  and  $p$  becomes larger when agents use the update function; this indicates that if the agents select make rational decision with higher probability, the total performance of MAS degrades, although these variations by  $l$  and  $p$  are smaller than the ones in Exp. 1; this is contrary to our intuition.

In contrast, when the agents use average value function, we cannot observe clear relation among  $p$ ,  $l$ , and  $RT$ . This means that by using the average value function, the convergent state of MAS does not depend on parameters  $p$  and  $l$ ; fluctuation does not make total performance better nor worse. As in Exp. 1, using average value function outperforms update function except when  $p$  is small, as shown in Fig. 2 (b).

## 5 Discussion and Conclusion

We investigated how learning parameters for agent local strategies to select partner agents influence on the total performance of an entire MAS. Although many task allocation methods have been proposed[1,2,3], our experiments show that task allocation is not obvious when MAS is massive and each agent has sophisticated abilities such as decision-making by its own information and learning. All the figures in this paper indicate that adaptation by update function makes the environment fluid, so some degradation is observed. However, our experiments assume quite simple situations; tasks are given in a constant rate of  $4T/t$ . In actual Internet environments, some characteristics always change. Hence, we believe there is the tradeoff between unstable and stable: in other words, adaptive and inadaptive; which is better may depend on the speed of environmental changes versus the speed of adaptation. However, I believe that adaptability is vital because the Internet is a changing environment. The issue is the relationship between the speed of change and adaptation; this is our next research issue.

## References

1. Mirchandany, R., Stankovic, J.: Using stochastic learning automata for job scheduling in distributed procesing systems. *Journal of Parallel and Distributed Computing* **38**(11) (1986) 1513–1525
2. Schaefer, A., Shoham, Y., Tennenholtz, M.: Adaptive Load Balancing: A Study in Multi-Agent Learning. *Journal of Artificial Intelligence Research* **2** (1995) 475–500
3. Mehra, T., Wah, B.W.: Population-based learning of load balancing policies for a distributed computer system. *Proc. of Computing in Aerospace 9 Conference.* (1993) 1120–1130
4. Sugawara, T. *et. al.*: Total Performance by Local Agent Selection Strategies in Multi-Agent Systems. *Proc. of 5th Int. Joint Conf. on Autonomous Agents and Multiagent Systems.* (2006)

# A Mobile Agent Approach to Support Parallel Evolutionary Computation

Wei-Po Lee

Department of Information Management  
National University of Kaohsiung  
Kaohsiung, Taiwan  
wplee@nuk.edu.tw

**Abstract.** To enhance the performance of evolutionary algorithms, different parallel computation models have been proposed, and they have been implemented on parallel computers to speed up the computation. Instead of using expensive parallel computing facilities, in this paper we propose to implement parallel evolutionary computation models on easily available networked PCs, and present a multi-agent framework to support parallelism. To evaluate the proposed approach, different kinds of experiments have been conducted to assess the developed system and the preliminary results show the efficiency of our approach.

## 1 Introduction

Evolutionary Algorithms (EAs) have become increasingly popular to solve problems in different domains. Yet, in order to solve more difficult problems, two inherent features of EAs, *pre-mature convergence* and *computation time* must be improved. To overcome the two problems, the idea of parallelizing EAs has been proposed by different researchers, and it has been proven to be a promising method [1]. Conceptually, the parallelism is to divide a big population in a sequential EA into multiple smaller sub-populations that are distributed to separate processors and can then be evaluated simultaneously. According to the sub-population size, the parallel EAs are categorized into two types: *coarse-grain* and *fine-grain*, and they are usually implemented on MIMD and SIMD computers, respectively.

Though parallel computers can speed up EAs dramatically, they are not easily available and it is expensive to upgrade their processing power and memory. The expensive parallel computers cannot be expected especially in a campus-based computing environment. A promising alternative without expensive hardware facilities is to construct the parallel EC framework on a set of networked personal computers. There are some possible ways to manage the operation of such a distributed computational framework, for example a client-server technique or an agent-based approach. Client-server is the most common paradigm of distributed computing at present. But in this paradigm all components are stationary in respect to execution. Therefore, mobile agent-based design paradigm provides a better choice to support parallel computing [2][3]. Mobile agents are software agents that are capable of transmitting

themselves across a computer network and recommencing execution at a remote site. Many multi-agent platforms are publicly available. Application developers can use them as platforms and focus on the software development issues at the application level.

To make the parallel EAs more realistic, in this paper we propose a mobile agent-based methodology to support parallelism on networked computers in which mobile agents dynamically allocate available machines for the EA-code. To verify the proposed approach we have developed a prototype application system on a middleware platform JADE (Java Agent Development Framework, [3]). Different kinds of experiments have been conducted to assess the developed prototype system and the preliminary results show the promise and efficiency of our approach.

## 2 The Proposed Mobile Agent-Based Approach

To develop a parallel EC framework without using a powerful connection machine, we choose to implement a coarse-grain model on a set of networked PCs in our laboratory. Figure 1 shows the aspect of our computational framework for island model parallelism. In this figure, each grey block contains a sub-population of individuals in which EC continues for a certain number of generations before migration happens. During this period, computation for each sub-population is independent from others, so evolution for different sub-populations can proceed simultaneously. In this model, migration happens only between immediate neighbors along different dimensions of the hypercube, and the communication phase is to send a certain number of the best individuals of each sub-population to substitute the same number of worst individuals of its immediate neighbors at a regular interval.

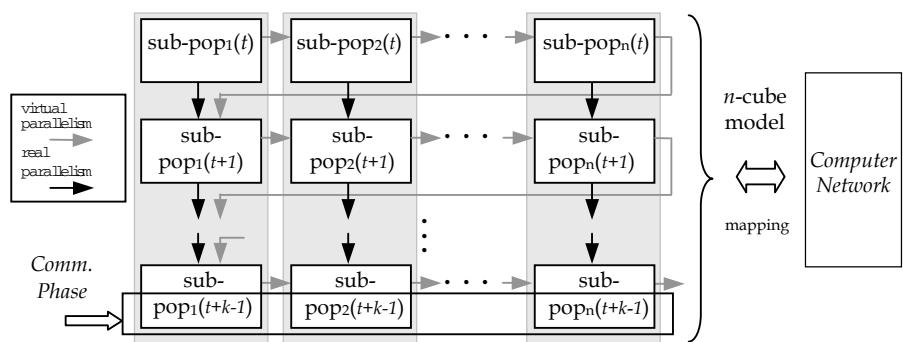
Though realizing parallelism in the above way is much cheaper than using a parallel computer, some machines have to be pre-specified to contribute their computation power for running sub-populations. This is in fact not practical for a computing laboratory environment in which the computational resources are shared by many end-users. Therefore, we develop an agent-based framework instead to manage the execution and communication for different sub-populations in an adaptive manner.

Our agent-based framework mainly includes three kinds of agents (in addition to the default agents provided by JADE for network services): the *mobile agent* to carry EC code, the *status agent* to report machine status, and the *synchronization agent* to record the evolving progress of sub-populations. Different agents communicate with each other in a pre-specified language through a common channel that is compliant with established FIPA standards to ensure the interoperability between agents. In the proposed framework, mobile agents play the major roles because they have the ability to migrate from machine to machine. In our strategy, each mobile agent takes the responsibility for running a sub-population. Initially, a status agent is created for each machine in the network framework, and the status agent SA in the main host is responsible for maintaining the information of individual machine status (e.g., the CPU utility) reported by other status agents. After the mobile agent MA starts the computation in the main host, it clones itself with the EC-code for each node of the pre-specified  $n$ -cube model. According to the information provided by SA, MA dispatches the duplication to each available machine. In the proposed framework, a mobile agent

can only execute its EC-code in an idle machine. Therefore, during the period of execution if a status agent in any machine (except the main host) detects the existence of a new end-user, it informs the mobile agent in the same machine to suspend the execution of its corresponding EC-code to release the computing resources for the end-user. The mobile agent then inquires machine availability from SA to find a free machine to resume the execution. If the mobile agent finds a machine with CPU utility lower than a certain threshold, it asks SA to reserve this machine, and then carries its code and related information to that machine to continue the computation from the point it is interrupted; otherwise it stays in the machine it has been in and waits for any available one.

A synchronized method is taken to exchange individuals for our island model EC, in which all sub-populations have to evolve for the same number of generation before the communication phase can happen. This is achieved by creating a synchronization agent in the main host to record evolving progress of different sub-populations. Once a sub-population has evolved for a pre-defined number of generations, the mobile agent responsible for it will send a message to the synchronization agent to indicate this situation, and the status agent in the same machine will inform SA to update the information of machine availability accordingly. Because sub-populations have to wait for each other and will release their computation resources while waiting, eventually each mobile agent can find a free machine for the sub-population not yet finished. In the worst case when all machines are taken by end-users during the execution, the mobile agents distributed in them can move back to the main host to continue their evolutionary computation.

To implement the above computing environment, we choose JADE as a platform and build our agents on it. JADE is compliant with FIPA standard specifications so that agents developed on it can thus interoperate with other agents built with the same standard. Also JADE allows each agent to dynamically discover other agents and to communicate with them in a peer-to-peer manner. By using such a platform, we can ignore the details of the middleware issues on a distributed architecture, but just concentrate on building agents to constitute our parallel evolutionary computing framework to solve application tasks.



**Fig. 1.** Control flows of the parallel evolutionary computation

### 3 Implementations and Experiments

Following our computational architecture and the mobile agent-based methodology, we conduct two series of experiments to compare the corresponding performance. The experiments are to evaluate the performance of using mobile agents to exploit the computational power of multiple machines. Two strategies, one static and one adaptive, are used to achieve the real parallelism. To verify our methodology, we use it to perform a time-consuming application task—evolving robot controllers [4]. Here, our previous genetic programming system ([5]) is extended with the agent-based parallel model to evolve robot controllers to achieve an obstacle avoidance task.

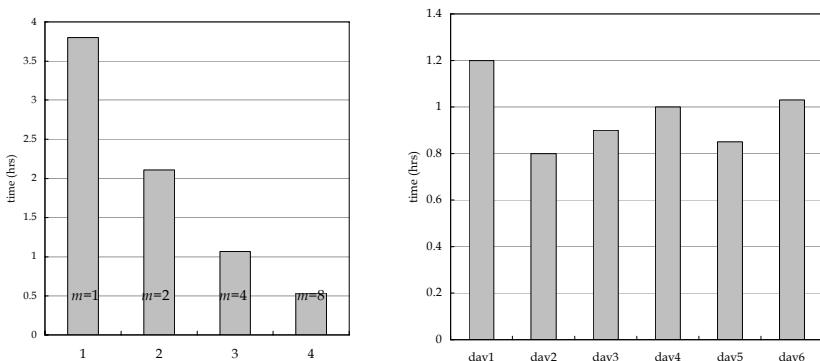
In the experiments of using agent-based strategies, we arranged eight networked computers running Windows NT operating system as a distributed computing environment to support the JADE platform. One of the computers played the role of “main host” where it was running the JADE Main-Container. Once the platform was activated, the JADE default agents AMS, DF, and RMA were instantiated, in which the DF agent provided other agents the information (e.g., the IP address) about the hosts connected to the JADE platform. Each machine in this framework had a status agent to report its corresponding status, and a mobile agent to take care of the computation for each sub-population. Also a synchronization agent was created in the main container to activate the communication phase for exchanging individuals between sub-populations.

The first phase is to examine whether the developed agent-based framework can be used to exploit the networked computing power to speed up the evolutionary computation. Hence, the agents remained static; they did not migrate between different machines. To compare the effect of using different numbers of computers, we have conducted experiments of one population of 400 individuals, two sub-populations of 200 individuals, four sub-populations of 100 individuals, and eight sub-populations of 50 individuals. In the experiments, the communication phase happened every ten generations and the migrants to be exchanged was 4% of a sub-population. These values were chosen because they were found to give the best performance in a small pilot study. Each sub-population was executed on one computer and end-users were not allowed to access the machines used for the experiments. Mobile agents here were created only to communicate with agents on other machines. Figure 2 (left) shows the average of the computational time spent for each strategy. As can be observed, the time for running a single experiment is reduced in an almost linear manner. Though in the above parallel model, agent communication for exchanging individuals between sub-populations needs extra computational effort, it is relatively small compared to the time for running a time-consuming evolutionary experiment.

Different from the above, the second phase is to show how the proposed mobile agent approach can support adaptive parallelism on network computers. In the experiments, eight hosts were connected to the JADE platform in which only one host was preserved for running EC experiments. The preserved host was initiated as the main container to enable the default agents AMS, DF, and RMA, and here end-users were allowed to use the other seven computers as they usually do. As in the first phase, each host was a container that included a mobile agent and a status agent. Also a synchronization agent was allocated in the main container. Initially, the EC code was executed on the main container, and then the mobile agent in this host checked

with the status agent to find other available hosts. For any free host, a duplicated mobile agent packed the EC-code and relevant information, moved to the target host, and started the execution for a new sub-population. After that, the mobile agents moved between different hosts according to the strategy described in section 2, to execute the corresponding EC-code for the application task.

As is mentioned, here the end-users were not restricted to use the machines connected to the mobile agent framework. Due to different user-accessing situations, the experiments of adaptive parallelism were conducted six times in different days. Figure 2 (right) shows the results. As can be seen, all runs have been sped up by the proposed approach. It shows the efficiency of our mobile agent-based approach.



**Fig. 2.** Computational cost for experiments with a static strategy ( $m$  is the number of computers used) and an adaptive strategy

## 4 Conclusions and Future Work

In this paper, we propose to implement parallel evolutionary computation models on easily available networked PCs and present a multi-agent framework for paralleling evolutionary computation in which mobile agents play the kernel roles to manage the execution and communication for the EC-code distributed in different computers. To evaluate our framework, two sets of experiments have been conducted for static and adaptive mobile agent strategies. The results show that both strategies can efficiently speed up the computation. Currently, we are investigating the issues of system reliability and fault tolerance by constructing a mechanism to ensure that our framework can recover from unexpected faults caused by the hardware defects or the inappropriate user-operations.

## References

1. Cantú-Paz, E.: Efficient and Accurate Parallel Genetic Algorithms. Kluwer Academic Publishers (2000)
2. Lange, D. B., Oshima, M.: Programming and Deploying Java Mobile Agents with Aglets. Addison-Wesley, Menlo Park, CA (1998)

3. Bellifemine, F., Poggi, A., Rimassa, G.: Developing Multi-Agent Systems with a FIPA-Compliant Agent Framework. *Software: Practice and Experience* 31 (2001) 103-128
4. Nolfi, S., Floreano, D.: *Evolutionary Robotics: the Biology, Intelligence, and Technology of Self-Organizing Machines*. MIT Press, MA (2000)
5. Lee, W.-P.: Evolving Complex Robot Behaviors. *Information Sciences* 121 (1999) 1-25

# The Design of Fuzzy Controller by Means of Genetic Algorithms and NFN-Based Estimation Technique

Sung-Kwun Oh<sup>1</sup>, Jeoung-Nae Choi<sup>2</sup>, and Seong-Whan Jang<sup>2</sup>

<sup>1</sup> Department of Electrical Engineering, The University of Suwon, San 2-2 Wau-ri, Bong-dam-eup, Hwaseong-si, Gyeonggi-do, 445-743, South Korea  
ohsk@suwon.ac.kr

<sup>2</sup> Department of Electronic and Information Engineering, Wonkwang University, 344-2, Shinyong-Dong, Iksan, Chon-Buk, 570-749, South Korea  
swhjang@wonkwang.ac.kr

**Abstract.** In this study, we introduce a neurogenetic approach to the design of fuzzy controllers. The design procedure exploits the technology of Computational Intelligence (CI) focusing on the use of genetic algorithms and neuro-fuzzy networks (NFN). The crux of the design concerns the selection and determination of optimal values of the scaling factors of the fuzzy controllers, which are essential to the entire optimization process. First, the tuning of the scaling factors of the fuzzy controller is carried out, and then the development of a nonlinear mapping for the scaling factors is realized by using GA-based NFN.

**Keywords:** Fuzzy Controller, Neurofuzzy Network (NFN), Genetic Algorithms, Estimation technique.

## 1 Introduction

In parallel to PID controllers that are regarded nowadays as the standard control constructs of numeric control [1], fuzzy controllers have positioned themselves in a similar dominant role at the knowledge-rich end of the entire spectrum of control algorithms. The intent of this study is to develop, optimize and experiment with the fuzzy controllers when developing a general design scheme of Computational Intelligence. One of the difficulties in the construction of the fuzzy controller is to derive a set of optimal control parameters of the controller such as linguistic control rules, scaling factors, and membership functions of the fuzzy controller. In the conventional design method being applied there, a control expert proposes some linguistic rules and decides upon the type and parameters of the associated membership functions. With an attempt to enhance the quality of the control knowledge conveyed by the expert (and this usually applies to the matter of calibration of such initial domain knowledge), genetic algorithms (GAs) have already started playing a pivotal role. The development process consists of two main phases. First, using genetic optimization we determine optimal parameters of the fuzzy controller for various initial states (conditions) of the dynamic system. Second, we build up a nonlinear model that captures a relationship between

the initial states of the system and the corresponding genetically optimized control parameters.

## 2 The Fuzzy Controller

In fuzzy PID controller, we confine ourselves to the following notation:  $e$  denotes the error between reference and response (output of the system under control),  $\Delta e$  is the first-order difference of error signal while  $\Delta^2 e$  is the second-order difference of the error. Note that the input variables to the fuzzy controller are transformed by the scaling factors ( $GE$ ,  $GD$ ,  $GH$ , and  $GC$ ) whose role is to allow the fuzzy controller to properly “perceive” the external world to be controlled.

The fuzzy PID controller consists of rules of the following form

$$R_j : \text{if } E \text{ is } A_{1j} \text{ and } \Delta E \text{ is } A_{2j} \text{ and } \Delta^2 E \text{ is } A_{3j} \text{ then } \Delta u_j \text{ is } D_j \quad (1)$$

The capital letters standing in the rule ( $R_j$ ) denote fuzzy variables (linguistic terms) whereas  $D$  is a numeric value (singleton) of the control action. An overall operation of a fuzzy PID controller can be described in the format so that the resulting control is formed incrementally based on the previous control

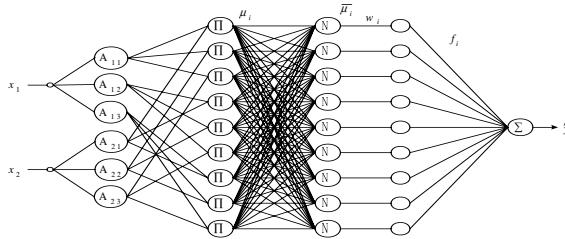
$$u(k) = u(k-1) + \Delta u(k) \quad (2)$$

## 3 Auto-tuning of the Fuzzy Controller Using GAs

In this study, the number of generations is set to 100, crossover rate is equal to 0.6, while the mutation rate is taken as 0.1. The number of bits used in the coding is equal to 10. Let us recall that this involves tuning of the scaling factors and a construction of the control rules. These are genetically optimized. We set the initial individuals of GAs using three types of parameter estimation modes such as a basic mode, contraction mode and expansion mode. In the case of a basic mode (BM), we use scaling parameters that normalize error between reference and output, one level error difference and two level error difference by  $[-1, 1]$  for the initial individuals in the GA. In a contraction mode (CM), we use scaling parameters reduced by 25% in relation to the basic mode. While in the expansion mode (EM), we use scaling parameters enlarged by 25% from a basic mode. The standard ITAE expressed for the reference and the output of the system under control is treated as a fitness function [2].

## 4 The Estimation Algorithm by Means of GA-Based Neurofuzzy Networks(NFN)

Let us consider an extension of the network with the fuzzy partition realized by fuzzy relations. Figure 1 visualizes an architecture of such NFN for two-input and one-output, where each input assumes three membership functions. The circles denote processing units of the NFN. The node indicated  $\Pi$  denotes a Cartesian product, whose output is the product of all the incoming signals. And  $N$  denotes the normalization of the membership grades.



**Fig. 1.** NFN structure by means of the fuzzy space partition realized by fuzzy relations

As far as learning is concerned, the connections change as follows

$$w(\text{new}) = w(\text{old}) + \Delta w \quad (3)$$

In this algorithm, to optimize the learning rate, momentum term and fuzzy membership function of the above NFN we use the genetic algorithm.

## 5 Experimental Studies

In this study, the dynamics of the inverted pendulum system are characterized by two state variables:  $\theta$  (angle of the pole with respect to the vertical axis),  $\dot{\theta}$  (angular velocity of the pole). The behavior of these two state variables is governed by the following second-order equation.

The dynamic equation of the inverted pendulum comes in the form

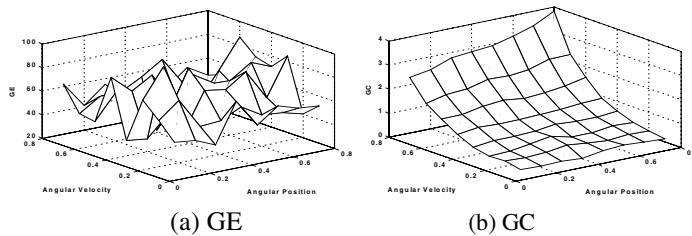
$$\ddot{\theta} = \frac{g \sin \theta + \cos \theta \left( \frac{-F - ml\dot{\theta}^2 \sin \theta}{m_c + m} \right)}{l \left( \frac{4}{3} - \frac{m \cos^2 \theta}{m_c + m} \right)} \quad (4)$$

Where  $g$  (acceleration due to gravity) is  $9.8 \text{m/s}^2$ ,  $m_c$  (mass of cart) is  $1.0 \text{kg}$ ,  $m$  (mass of pole) is  $0.1 \text{kg}$ ,  $l$  (length of pole) is  $0.5 \text{m}$  and  $F$  is the applied force expressed in Newtons.

Proceeding with the genetic optimization, we consider the ITAE (Integral of the Time multiplied by the Absolute value of Error), overshoot and rising time as the PI (Performance Index) of the controller as the three underlying criteria.

We selected  $0.1 \text{rad.}, 0.2 \text{rad.}, \dots, 0.7 \text{rad.},$  and  $0.8 \text{rad.}$  as a collection of initial angular positions and  $0.1 \text{rad/sec}, 0.2 \text{rad/sec}, \dots, 0.7 \text{rad/sec},$  and  $0.8 \text{rad/sec}$  as the corresponding family of values of the initial angular velocity. We tune (adjust) the control parameters of each controller (fuzzy PID controller, fuzzy PD controller and PID controller). Figure 2 visualizes the value of the scaling factors treated as a function of initial angular position and angular velocity of the inverted pendulum in the fuzzy PID controller. Evidently there are nonlinear characteristics.

In general, Fuzzy PD and fuzzy PID controllers are preferred architectures. But PID controller is also satisfactory in comparison to the fuzzy PID controller within a linear range of  $\theta < 0.4$ , while in case of a nonlinear range of  $\theta > 0.6$  Fuzzy PID controller architecture performs better than both Fuzzy PD and PID controller.



**Fig. 2.** Auto-tuned scaling factors according to the change of initial angles and angular velocity in the fuzzy PID controller (a) GE and (b) GC

The fuzzy PID controller and fuzzy PD controller are superior to the conventional PID controller from the viewpoint of ITAE, overshoot and rising time.

Now, we consider the case in which the initial angular positions and angular velocities of the inverted pendulum are selected arbitrarily within the given range. Here we show that the control parameters under the arbitrarily selected initial condition are not tuned by the GAs and the control parameters of each controller are estimated by using the estimation algorithm of GA-based NFN. We implement the optimal neuro-fuzzy networks for parameter estimation using GAs. In this algorithm, we adjust the learning rates, momentum coefficient, and apexes of membership function of neuro-fuzzy networks by using GAs. Table 1 shows the estimated scaling factors of the fuzzy controllers and control parameters of PID controller and describes performance index (ITAE, overshoot and rising time) in case of  $\theta = 0.22, 0.45(\text{rad})$  and  $\dot{\theta} = 0.22, 0.78(\text{rad/sec})$  respectively.

**Table 1.** The estimated parameters by means of the GA-based NFN and performance index(ITAE, overshoot and rising time) of controllers in the case of  $\theta = 0.22, 0.45(\text{rad})$  and  $\dot{\theta} = 0.22, 0.78(\text{rad/sec})$

Controller Type	Initial Angle	Initial angular velocity	GE	GD	GH	GC	ITAE	Overshoot (%)	Rising time (sec)
FPID	0.22	0.22	2.0328	61.546	237.3	3.706	0.419	0.000	0.261
	0.45	0.78	1.9079	61.082	236.4	2.411	0.855	0.000	0.167
FPD	0.22	0.22	7.4378	0.529		1.854	0.149	0.000	0.129
	0.45	0.78	4.1532	0.305		2.865	0.728	0.102	0.149
PID	K			Ti			Td		
	0.22	0.22	168.686	164.717	0.104		0.247	0.087	0.172
	0.45	0.78	168.507	165.861	0.103		0.953	0.147	0.181

In Table 1, we know that the fuzzy PD and fuzzy PID control effectively the inverted pendulum system. The proposed estimation algorithm such as GA-based NFN generates the preferred model architectures. The performance of the fuzzy controllers such as the fuzzy PD and the fuzzy PID controller with evidently nonlinear characteristics are superior to that of the PID controller especially in a nonlinear range of  $\theta > 0.45$  when using the nonlinear dynamic equation of the inverted pendulum, while in case of a linear range  $\theta < 0.45$ , PID controller is also satisfactory in comparison

fuzzy PID controller. Especially the fuzzy PD controller describes the most preferred one among the controllers when using NFN-based estimation technique.

## 6 Conclusions

In this paper, we have proposed a two-phase optimization scheme of the fuzzy controllers. The parameters under optimization concern scaling factors of the input and output variables of the controller that are known to exhibit an immense impact on its quality. The first phase of the design of the controller uses genetic computing that aims at the global optimization of its scaling factors where they are optimized with regard to a finite collection of initial conditions of the system under control. In the second phase, we construct a nonlinear mapping between the initial conditions of the system and the corresponding values of the scaling factors. Our simulation studies reveal that when using the genetic optimization for the scaling factor estimation mode and the estimation algorithm of the GA-based neurofuzzy networks model, the fuzzy PD/PID controller controls the inverted pendulum system in a highly effective manner especially for a nonlinear range of  $\theta$ .

**Acknowledgements.** This work was supported by Wonkwang University (2005).

## References

1. Oh, S.-K.: Fuzzy Model & Control System by C-Programming. Naeha Pub. Co. (2002).
2. Goldberg, D.E.: Genetic algorithms in Search, Optimization, and Machine Learning. Addison-Wiley Pub. Co. (1989)
3. Jang, J.R.: Self-Learning Fuzzy Controllers Based on Temporal Back Propagation. IEEE Trans. On Neural Networks. **3(5)** (1992) 714-723
4. Wang, L.: Stable Adaptive Fuzzy Controllers with Application to Inverted Pendulum Tracking. IEEE Trans. On Systems. Man and Cybernetics-Part B. **26(5)** (1996) 677-691
5. Yamakawa,T.: A New Effective Learning Algorithm for a Neo Fuzzy Neuron Model. 5th IFSA World Conference. (1993) 1017-1020
6. Oh, S.-K., Pedrycz,W., Lee, D.-K.: The Genetic Design of Hybrid Fuzzy Controllers. Cybernetics & Systems. **35** (2004) 333-361
7. Oh, S.-K.: Computational Intelligence by Programming(focused on Fuzzy, Neural Networks, and Genetic Algorithms). Naeha Pub. Co. (2002)
8. Oh, S.K., Park, B.J., Kim, H.K. : Genetically Optimized Hybrid Fuzzy Neural Networks Based on Linear Fuzzy Inference Rules. International Journal of Control Automation and Systems. **3(2)** (2005) 183-194

# GA-Based Polynomial Neural Networks Architecture and Its Application to Multi-variable Software Process

Sung-Kwun Oh<sup>1</sup>, Witold Pedrycz<sup>2</sup>, Wan-Su Kim<sup>1</sup>, and Hyun-Ki Kim<sup>1</sup>

<sup>1</sup> Department of Electrical Engineering, The University of Suwon, San 2-2 Wau-ri,  
Bongdam-eup, Hwaseong-si, Gyeonggi-do, 445-743, South Korea  
[ohsk@suwon.ac.kr](mailto:ohsk@suwon.ac.kr)

<sup>2</sup> Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB  
T6G 2G6, Canada and Systems Research Institute, Polish Academy of Sciences,  
Warsaw, Poland

**Abstract.** In this paper, we propose a architecture of Genetic Algorithms (GAs)-based Polynomial Neural Networks(PNN), discuss a comprehensive design methodology and carry out a series of numeric experiments. GA-based design procedure at each stage (layer) of PNN leads to the selection of preferred nodes (or PNs) with optimal parameters (such as the number of input variables, input variables, and the order of the polynomial) available within PNN. A detailed design procedure is discussed in detail. To evaluate the performance of the GA-based PNN, the model is experimented with by using Medical Imaging System (MIS) data for application to Multi-variable software process. A comparative analysis shows that the proposed GA-based PNN is model with higher accuracy as well as more superb predictive capability than other intelligent models presented previously.

## 1 Introduction

The panoply of existing methodologies and ensuing detailed algorithms are confronted with nonlinear systems, extreme problem dimensionality, a quest for high accuracy and generalization capabilities of the ensuing models. When the complexity of the system to be modeled increases, experimental data as well as some degree of prior domain knowledge (conveyed by the model developer) are essential to the completion of an efficient design procedure. It is also worth stressing that the nonlinear form of the model acts as a two-edged sword: while we gain flexibility to cope with experimental data, we are provided with an abundance of nonlinear dependencies that need to be exploited in a systematic manner. In particular, when dealing with high-order nonlinear and multivariable equations of the model, we require a vast amount of data necessary for estimating its complete range of parameters [1-2].

In this study, in solving the problems with the conventional PNN as well as the GMDH algorithm, we introduce a new design approach of GA-based PNN. Optimal design parameters available within the PN (viz. the number of input variables, the order of the polynomials, and input variables) lead to a structurally and parametrically optimized network, which is more flexible as well as simpler in architecture than the

conventional PNN. To evaluate the performance of the proposed model, we exploit Medical Imaging System (MIS) data [5].

## 2 The PNN Algorithm and Its Generic Architecture

### 2.1 PNN Algorithm

The PNN algorithm [3-4] is based on the GMDH method and utilizes a class of polynomials such as linear, quadratic, and modified quadratic. The individual PNs are expressed as a second-order regression equation. In particular, when combining two inputs at each node as the generic structure we arrive at the following relationship;

$$y = A + BX_i + CX_j + DX_i^2 + EX_j^2 + FX_iX_j \quad (1)$$

In the above expression,  $A, B, C, D, E$ , and  $F$  are parameters of the model, while  $y$  is the output of this model;  $X_i$  and  $X_j$  denote two inputs.

The outputs obtained from each of these nodes are then combined to obtain a higher-degree polynomial. In this case, a complex polynomial is formed (referred to as an Ivakhnenko polynomial). This function usually takes on the form

$$y = A + \sum_{i=1}^n B_i X_i + \sum_{i=1}^n \sum_{j=1}^n C_{ij} X_i X_j + \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^n D_{ijk} X_i X_j X_k \dots \quad (2)$$

where  $X_i, X_j$  and  $X_k$  are the nodal input variables, and  $y$  is the output of an individual neuron (node).  $A, B_i, C_{ij}$ , and  $D_{ijk}$  are the coefficients of the Ivakhnenko polynomial.

### 2.2 PNN Architecture

The PNN based on the perceptron learning principle with neural network-type architecture is used to model the input-output relationship of a complex process system. The design of the PNN structure continues and involves the generation of some additional layers. Each layer consists of nodes (PDs or PNs) for which the number of input variables could be the same as in the previous layers or may differ across the network. At each layer, new generations of complex equations are constructed from simple forms. The model obtained at each layer is progressively more complex than the model at the preceding layers. The final model is defined as a function of two, three, or four variables. The network result is a very sophisticated model obtained from a very limited data set. In what follows, the PNN emerges as a versatile architecture whose topology depends on the regression polynomial of a PN.

## 3 Genetic Optimization of PNN

For the optimization of the PNN model, GAs use the serial method of binary type, roulette-wheel in the selection operator, one-point crossover in the crossover operator, and invert in the mutation operator. As the roulette-wheel operator's stochastic characteristic, when creating new population of new generation by selection operator, we

will choose the best chromosome from the last generation. To reduce the stochastic errors of roulette-wheel selection, we use elitist strategy [7-9].

## 4 GA-Based PNN Algorithm

The framework of the design procedure of the Polynomial Neural Networks (PNN) consists of the following steps.

- [Step 1] Determine system's input variables
- [Step 2] Form training and testing data
- [Step 3] Determine initial information for constructing the PNN structure
- [Step 4] Determine PN structure using genetic design.
- [Step 5] Estimate the coefficients of the polynomial corresponding to the selected node (PN)
- [Step 6] Select nodes (PNs) with the best predictive capability, and construct their corresponding layer.  
All nodes of the corresponding layer of PNN architecture are constructed by genetic optimization. The objective function (performance index) is a basic instrument guiding the evolutionary search in the solution space [10].
- [Step 7] Check the termination criterion

In this study, we use a measure (performance index) that is the Mean Squared Error (MSE).

$$E(PI_s \text{ or } EPI_s) = \frac{1}{N} \sum_{p=1}^N (y_p - \hat{y}_p)^2 \quad (3)$$

where,  $y_p$  is the  $p$ -th target output data and  $\hat{y}_p$  stands for the  $p$ -th actual output of the model for this specific data point.  $N$  is training( $PI_s$ ) or testing( $EPI_s$ ) input-output data pairs and  $E$  is an overall(global) performance index defined as a sum of the errors for the  $N$ .

- [Step 8] Determine new input variables for the next layer

The PNN algorithm is carried out by repeating steps 4-8 consecutively.

## 5 Experimental Studies

This section includes comprehensive numeric studies illustrating the design of the GA-based PNN model. We use a well-known medical imaging system (MIS) [5] data.

We consider a Medical Imaging System (MIS) subset of 390 software modules written in Pascal and FORTRAN for modeling. Table 1 summarizes the list of parameters used in the genetic optimization of the network. Table 2 summarizes the performance of the network when changing the maximal number of inputs to be selected; here the "Max" parameter was set up to 2 through 5 and 9.

**Table 1.** Summary of the parameters of the genetic optimization

Parameters			1 <sup>st</sup> ~ 5 <sup>th</sup> layer
GA			Maximum generation 300
			Total population size 150
			Selected population size ( $W$ ) $30(l=2\sim 5), 60(l=9)$
			Crossover rate 0.65
			Mutation rate 0.1
PNN			String length $3+3+30(l=2\sim 5), 5+3+60(l=9)$
			Number of inputs to be selected ( $l$ ) $1 \leq l \leq \text{Max } (2\sim 5, 9)$
			Type (T) $1 \leq T \leq 3$
			Weighting factor ( $\theta$ ) $0 \leq \theta \leq 1$

*l,T:* integer**Table 2.** Performance index of the network of each layer versus the increase of maximal number of inputs to be selected

Max	1 <sup>st</sup> layer			2 <sup>nd</sup> layer			3 <sup>rd</sup> layer			4 <sup>th</sup> layer			5 <sup>th</sup> layer																											
	Node	T	PIs	EPIs	Node	T	PIs	EPIs	Node	T	PIs	EPIs	Node	T	PIs	EPIs																								
2	4	3	2	47.22	41.75	12	20	2	44.19	33.44	22	24	2	37.65	28.72	21	23	2	34.35	26.77	8	27	2	23.84	25.15															
3	7	4	3	2	44.20	32.29	12	19	14	2	33.15	23.17	18	8	4	2	29.07	18.13	15	18	21	1	27.24	17.82	24	25	28	2	25.64	17.05										
4	6	4	7	3	2	38.48	26.44	22	29	28	4	2	28.44	21.21	24	2	17	22	3	26.39	16.72	20	25	7	0	2	23.38	14.75	18	17	8	14	1	22.47	12.52					
5	7	4	6	11	3	2	37.36	26.48	4	8	18	27	0	3	30.18	19.96	19	9	24	4	13	3	23.85	16.71	20	13	14	5	0	3	23.01	13.77	15	21	6	13	4	1	22.23	12.56
9	Selected=5	2	37.36	26.48	Selected=5	2	22.63	24.62	Selected=4	2	19.7	17.18	Selected=4	2	18.84	13.24	Selected=8	1	18.04	11.89																				

Table 3 summarizes the results of comparative analysis of the proposed model with respect to other constructs.

**Table 3.** Comparison of identification error with previous models

Model				PI	PI <sub>s</sub>	EPI <sub>s</sub>
Regression model				36.13		
PNN				8.456		
SONFN[16]	Simplified Linear	Generic Type	Basic Architecture	40.753	17.898	
				35.745	17.807	
FPNN[17]	No. of inputs: 2	Triangular	Type : 2	32.195	18.462	
		Gaussian	Type : 1	49.716	31.423	
	No. of inputs: 3	Triangular	Type : 1	32.251	19.622	
		Gaussian	Type : 1	39.093	19.983	
Our model	max-inputs: 5 ( $\theta=0.5$ )	$\theta=0.5$	Type : $1 \leq T \leq 3$	22.238	12.566	
	max-inputs: 9			7.161	18.043	11.898

## 6 Conclusions

In this study, the GA-based design procedure of Polynomial Neural Networks (PNN) and its design methodology were proposed to construct optimal model architecture for nonlinear and complex system modeling. The design methodology comes with hybrid structural optimization and parametric learning viewed as two phases of modeling building. That is, the one phase (hybrid structural optimization) is realized via both

GAs and a structural phase of an evolutionary algorithm as the main characteristics of the GMDH method while the other phase (parametric optimization) is carried out by a standard least square estimation (LSE)-based learning. The comprehensive experimental studies involving well-known datasets (Medical Imaging System (MIS) data) quantify a superb performance of the network in comparison to the existing models. First of all, we could efficiently search for the optimal network architecture (structurally and parametrically optimized network) by the design methodology of GA-based PNN in comparison to that of the conventional PNN

**Acknowledgements.** This work has been supported by KESRI(I-2004-0-074-0-00), which is funded by MOCIE(Ministry of commerce, industry and energy).

## References

1. Cherkassky, V., Gehring, D., Mulier, F.: Comparison of adaptive methods for function estimation from samples. *IEEE Trans. Neural Networks.* **7** (1996) 969-984
2. Dicherson, J.A., Kosko, B.: Fuzzy function approximation with ellipsoidal rules. *IEEE Trans. on Systems, Man and Cybernetics, Part B.* **26** (1996) 542-560
3. Oh, S.-K., Pedrycz, W.: The design of self-organizing Polynomial Neural Networks. *Information Science.* **141** (2002) 237-258
4. Oh, S.-K., Pedrycz, W., Park, B.-J.: Polynomial Neural Networks Architecture: Analysis and Design. *Computers and Electrical Engineering.* **29** (6) (2003) 703-725
5. Lyu, M.R. (ed.): *Handbook of Software Reliability Engineering*. McGraw-Hill and IEEE Computer Society Press, New York and Washington (1995)
6. Park, H.-S., Park, B.-J., Kim, H.-K., Oh, S.-K.: Self-Organizing Polynomial Neural Networks Based on Genetically Optimized Multi-Layer Perceptron Architecture. *International Journal of Control, Automation, and Systems.* **2**(4) (2004) 423-434
7. Goldberg, D.E.: *Genetic Algorithm in search, Optimization & Machine Learning*. Addison wesley, Boston (1989)
8. Kenneth, A., De, J.: Are Genetic Algorithms Function Optimizers?. *Parallel Problem Solving from Nature 2*, Manner, R. and Manderick, B.(eds.), North-Holland Amsterdam (1992)
9. Michalewicz, Z.: *Genetic Algorithms + Data Structures = Evolution Programs*. 3rd edn. Springer-Verlag, Berlin Heidelberg New York (1996)
10. Oh, S.-K., Pedrycz, W.: Identification of Fuzzy Systems by means of an Auto-Tuning Algorithm and Its Application to Nonlinear Systems. *Fuzzy Sets and Systems.* **115** (2) (2000) 205-230
11. Oh, S.-K., Pedrycz, W., Park, B.-J.: Relation-based Neurofuzzy Networks with Evolutionary Data Granulation. *Methematical and Computer Modeling.* **40** (7-8) (2004) 891-921
12. Oh, S.-K., Pedrycz, W.: Fuzzy Polynomial Neuron-Based Self-Organizing Neural Networks. *Int. J. of General Systems.* **32** (3) (2003) 237-250

# Topical and Temporal Visualization Using Wavelets

T. Mala, T.V. Geetha, and Sathish Kumar

Department of Computer Science and Engineering, College of Engineering, Guindy,  
Chennai, India  
mala@cs.annauniv.edu

**Abstract.** The goal of this paper is to provide a visual structure to view the temporal and thematic content of English news documents. This paper combines both information extraction techniques and visualization techniques to visualize the documents content. Topical information present in the documents are extracted by applying Booksteins model and by calculating the various measures associated with the model. Temporal information is extracted as date information from the documents. Later the extracted information is converted into a signal and both discrete and continuous wavelet transformations are applied over the signal. The content of the document is visualized at varying levels of detail by multi resolution analysis techniques available in wavelets. This gives a wavelet based visualization model which enables us to view the spread of topics across a document, the portions of the document that are most involved in topic description, and the contribution of documents in the corpus from the temporal perspective.

## 1 Introduction

The volume of online news available today has increased tremendously. It would be valuable to have an automated system, which would sift through the mounds of news and display the articles that match an individual's interests [10]. Such a system would need to identify and display the topics within the articles, and the temporal and spatial information associated with the topics. Topical words and temporal information from news articles are generally extracted using Information Extraction techniques. Here the extracted topical words and temporal information are converted into signals and are visualized applying wavelets over the signals. The content of a text document with its temporal information are visualized at varying levels of detail. The varying levels of detail are directly available from the multi resolution levels (MRLs) that result from the application of wavelet transforms to a mathematical signal built from the text narrative [1]. In this paper the emphasis is on wavelet based visualization of topical and temporal information.

## 2 Wavelet Visualization

Wavelet is an excellent tool for visualization. The fundamental idea behind wavelets is to analyze according to scale. The wavelet transform decomposes the input signal

into a low frequency smooth or approximation half and a high frequency detailed half [2]. In this work, wavelet based visualization has been used to visualize topical and temporal information in unstructured textual data with varying degrees of detail. In order to carry out wavelet-based visualization it is necessary to choose the appropriate features of the input, in this case the text document. Therefore the first step in the visualization process is the selection of appropriate features that indicate topical and temporal information and their extraction.

### 3 Topic Word Extraction

Most topical word extraction methods use text classification methodology based on frequently occurring words [5]. Each topic is modeled as an infinite mixture over an underlying set of topic probabilities. In the context of text modeling the topic probabilities provide explicit representation of a document [4]. Bookstein's model has proposed interesting measures that help to determine the content bearing words of a document based on the underlying assumption that occurrences of a term sensitive to content will have a tendency to clump, than those of non- content bearing words [3]. In the following section we discuss some of the measures suggested by Bookstein which are used by us for determining topics. In order to calculate all the measures described above the text document has to be pre processed. Text preprocessing is done by eliminating all the stop words in the document and by comparing the document against a stop word dictionary. The next step in preprocessing is to transform different inflections and derivations of the same word to one common stem which is called as stemming [9]. Later the document was divided into segments. The selection of a segment boundary is done approximately. The term textual unit represented in all the measures below means this segment.

Condensation Measure C1-Term condensations over textual units $p(N, T) = \frac{N! \{p\} \{T\}}{D^N}$	Condensation Measure C2-Term distributions over textual units: $p_m = \binom{T}{m} (1/D)^m (1 - 1/D)^{T-m}$	Linear Measure L1-Number of clumps $p_K = \frac{\binom{N-1}{K-1} \binom{D-N+1}{K}}{\binom{D}{N}}$	Linear measure L2:-Gap length between marked units $E_{L1} = N \left( 1 - \frac{N-1}{D} \right)$
---	--	--	---

where,

- D: Number of textual units in document
- N: Number of units containing the term
- T: Total number of occurrences of the term
- $p_m$ : The probability of m occurrence
- $p_k$ : The probability of occurrence of K clumps

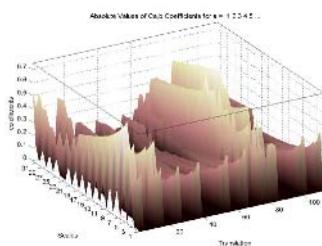
The above measures were calculated for all words in a preprocessed corpus of news documents related to military attacks. The Topical words were selected from the news documents based on the metric values calculated as stated above.

## 4 Date Extraction

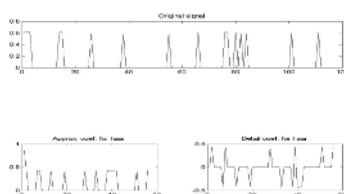
Temporal information associated with a news document gives information regarding the time of occurrence of the event. Much of the temporal information conveyed in natural language text is left implicit. Recent works has focused on developing schema for making this information explicit typically via annotation[11]. The type of time axes considered in a typical Information Extraction task is normally limited to date and time values[6,7]. In our approach Date information is extracted from news documents for temporal information visualization. The date extractor extracts the date by using sliding window, where sliding window takes five words at a time. The date information thus extracted is helpful in visualizing the topical flow of various documents arranged in date order.

## 5 Wavelet Visualization of Topics

The features extracted using the above techniques are used as topical and temporal representation of the document set. The mean topical strength of each topic is calculated and this value with the position of the occurrence of the topical word forms a vector. This vector based information is converted into a signal. Then the signal is transformed for analysis. If we choose a part of the signal, we receive frequency values with a certain variance [8]. Wavelet coefficients are calculated for the signal with scale ranging from 1 to 32 and positions or translations varying from 1 to length of signal. Continuous wavelets transform of the position of the topical word in the document and their respective strengths are plotted. Small coefficients are given dark colors and large coefficients are given light colors. Higher values of coefficients tell us that the signal portion is closely related to the wavelet. For simplicity Haar , the simplest wavelet is used. The Visual output is shown below.



**Fig. 1.** Wavelet Visualization of topical information extracted from a news document



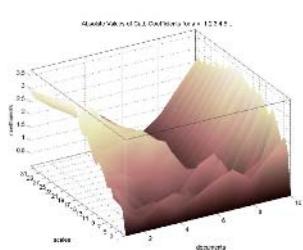
**Fig. 2.** Discrete Wavelet transformation of a news document

In figure 1 the three axes are wavelet coefficients, scales and translations. Translation tells where exactly the mother wavelet is situated with respect to the document position. We can infer that the first half of the document consists of low

scales which means it represents high topical content and the second half of the signal shows high scales which represent low topical strength . The increase in coefficient value in the second half of the signal tells how much the signal is related to the wavelet. This figure also analyses the pattern of clumps of topical words in first quarter half and in the 60 to 80th portion of the document. Discrete wavelet transform of the signal are also obtained which gives the approximations and detail parts clearly. In figure 2 X axis represent the translation parameters and Y axis represent the discrete wavelet coefficients. The approximations show high scale where the frequency resolution is good and poor time resolution. It implies good topical strength resolution but poor document position resolution, whereas the detail coefficients show low scale which implies good time resolution and poor frequency resolution. Here it means good document position resolution but poor topical strength resolution.

## 6 Wavelet Visualization of Date

Similarly the date extracted from the document helps in arranging the documents in temporal order. Mean topical word strength of all words in the document is calculated for each document and is considered as the document strength. The documents strength is visualized by applying the continuous wavelet transform over the documents in temporal order.



**Fig. 3.** Wavelet visualization of temporal information extracted from ten news documents

Fig 3 can be analyzed as in the previous case. Here translation refers to temporal information and the figure shows that according to chronological order the first document has more topical strength and the strength of the topics decay as it moves to later documents and the last set of documents have high coefficients which implies good topical strength.

## 7 Advantages of Wavelet Visualization

We can analyze the document at all levels by spatial document analysis. Document similarity, Topical strength analysis and temporal analysis can also be performed. English news articles related to terrorist activities were taken and the topical words and temporal words were extracted. For topical visualization one document is used and for temporal visualization ten documents were used. A large scale user quantitative evaluation of the above visualization was performed on the basis of the time taken by user to complete the task, number of interface interactions, user opinion about the usability of interface, number of errors the user makes when using the visualization. The user responses confirmed the findings of the paper.

## 8 Future Enhancements and Conclusion

The work presented above helps in visual analysis of topical words and temporal word position in the document. Thematic localization of the document can be visualized by spatial document analysis. Temporal analysis helps in analyzing the similarity between documents that exist in a corpus with respect to their topical strength. This work can be extended by finding other related measures of a document which can help in efficient topic word extraction and applying that for wavelet visualization. Visualization using other wavelets like dabuchois can also be performed. The visualization of the document corpus using the method discussed in this paper enables us to view the spread of topics across a document, the portions of the document that are most involved in topic description, and the contribution of documents in the corpus from the temporal perspective.

## References

1. Nancy E. Miller\*, Pak Chung Wong\*\*, Mary Brewste, Harlan Foote\*, TOPIC ISLANDSTM - A Wavelet-Based Text Visualization System , Pacific Northwest National Laboratory' tt'1998 IEEE.
2. Amen Zwa David S. Ebert Ethan L. Miller , Multiresolution Document Analysis with Wavelets November 22, conference on Information and knowledge management, Rockville,MD,December 1996,pages 50-53
3. A.Bookstein,S.T.Klein,T.Rait "Clumping properties of content bearing words"-white paper.
4. David M.blei,Andrew Y.Ng, Michael .I, "Latent dirichlet allocation", Journal of machine learning research3(2003) 993-1022.
5. H. P. Luhn. A statistical approach to mechanized encoding and searching of literary information. IBM Journal of Research and Development, 1(4):309{317, 1957.
6. N.Chinchor(1997).MUC7 Named entity task definition, September 1997 . [http://www.itl.nist.gov/894.02/related\\_projects/muc/proceedings/ne\\_task.html](http://www.itl.nist.gov/894.02/related_projects/muc/proceedings/ne_task.html)
7. MUC-6 named entity task definition(1995) may(1995) [http://www.cs.nyu.edu/cs/faculty/grishman/NETask20.book\\_1.html](http://www.cs.nyu.edu/cs/faculty/grishman/NETask20.book_1.html)
8. Laurence A.F.Park Spectral based information retrieval –White paper
9. Johan Carlberger, Hercules Dalianis, Martin Hassel, Ola Knutsson”, Improving Precision in Information Retrieval for Swedish using Stemming
10. T.Mala Dr.T.V.Geetha Visualization of Biological Patterns in Event Detection and Tracking Based on SRP algorithm, in proceedings of Ninth International conference on Information Visualization ,6-8 July 2005, pages 587 to 592, London,England. IEEE CS press.
11. G.Katz,J.Pustejovsky,F.Schilder “ Annotation extracting and reasoning about time and events”, Dagstahl seminar proceedings 05151.

# LP-TPOP: Integrating Planning and Scheduling Through Constraint Programming

Yuechang Liu and Yunfei Jiang

Software Research Institute, Sun Yat-Sen University  
Guangzhou 510275, P.R. China  
[ychangliu@hotmail.com](mailto:ychangliu@hotmail.com),  
[lnchsri05@zsu.edu.cn](mailto:lnchsri05@zsu.edu.cn)

**Abstract.** In this paper we present LP-TPOP, a domain-independent temporal partial-order planning algorithm that handles CBI-style action model. By the utilization of ground actions variable binding constraints can be eliminated from partial plans, then all the constraints needed to be handled are temporal constraints, which are maintained in a revised simple temporal constraint network (r-STN) structure. The exact scheduling of plan returned by planning phase is calculated by a linear programming module. By exploiting incremental algorithms the efficiency of LP-TPOP can be further improved. LP-TPOP is proved to be sound and complete.

## 1 Introduction

Classical planning deals with the task of generating an ordering of given actions in order to achieve some goals when the actions are carried out following the order. In classical planning paradigm actions are assumed to be instantaneous. In that case there never is the issue of concurrent execution of actions [2]. Temporal planning enables the reasoning about more complicated interactions among actions by attaching temporal factor to actions' durations, actions' conditions and effects.

Temporal planning can also be viewed as a problem integrating planning with scheduling [1, 3, 5]. Under CBI (constraint based interval) action model [5], LP-TPOP explores coupling planning and scheduling via constraint management. LP-TPOP runs in POCL (partial-order causal link) planning manner. During the expansion of the partial plans, the temporal factor of selected actions is added to r-STN (revised simple temporal network) structure and a consistency checking procedure is invoked. If inconsistency is detected then the action last selected will be dropped, and a backtracking occurs. Once a complete plan is achieved successfully the temporal constraints will be transformed into linear equations and passed to a linear programming engine, and finally a feasible schedule is calculated. While the schedule may be not optimal globally, it is locally optimal w.r.t. a given plan.

The paper is organized as follows: in section 2, an overview of LP-TPOP will be presented. Section 3 is dedicated to the description of components and some formal properties of LP-TPOP, followed by the conclusions and future work in section 4.

## 2 Overview of LP-TPOP Algorithm

Basically the LP-TPOP algorithm is similar to Smith's CBI planning algorithm [5] and the standard partial-order planning algorithm [7]. Roughly there're three basic components: action selection (planning), consistency maintenance, and resolution of temporal constraints (scheduling). Following is the high-level LP-TPOP algorithm.

### **LP-TPOP**

```

input: planning problem P;
output: final partial plan if success, fail otherwise;
begin
1). Create initial partial plan  $P_0$ ;  $P=P_0$ ;
2). Heuristically select  $p#[t1,t2] \in \text{Open-Goals}(P)$ ;
   2.1).  $\text{RES} := \{\text{all actions that can support } p#[t1,t2]\}$ ;
   2.2). If  $\text{RES} = \emptyset$  return(failure);
3). for all partial plans  $P_s = \text{Refine}(\text{RES}, P)$ 
   constructed from all the resolvants in RES:
   3.1). Select  $P_i$  from  $P_s$ ;
   3.2). If  $\text{Consistency\_Check}(P_i) = \text{false}$  then go to 3.1);
   3.3). else if  $\text{Open-Goals}(P_i) = \emptyset$  and  $\text{Schedule}(P_i) = \text{fail}$ 
         then remove  $P_i$  from  $P_s$  and go to 3.1);
   3.4). else if  $\text{Open-Goals}(P_i) = \emptyset$  then
         return( $\text{Schedule}(P_i)$ );
   3.5). If  $\text{Open-Goals}(P_i) \neq \emptyset$  then go to 2);
end.
```

Unlike the way general POCL planning algorithm handles *flaws* [7], LP-TPOP first selects an *open goal* (i.e. unsupported precondition of action in a partial plan) to resolve by adding an appropriate action to current partial plan, and then refines *threats* (i.e. the potential influence an action has on some causal links) by adding temporal constraints (in Refine procedure in step 3)). Step 3.1) is a backtrack point where some partial plan (from multiple candidates resulting from different action selections) is selected to expand. When there're no open goals in the current partial plan and it is consistent, all its temporal constraints are passed to a linear programming engine to calculate the exact start time of each action (step 3.4)). Otherwise it fails (step 3.3)).

The use of ground actions makes LP-TPOP attain efficiency gain by releasing LP-TPOP from the burden of parameter unification and variable binding constraint processing. However, it also results in larger branching factor. To overcome such drawback, some techniques are exploited, e.g. no-good recording technique.

Zeno is another POCL temporal planner. Zeno is much more expressive than LP-TPOP [6]. As a preliminary work, LP-TPOP focuses only on temporal feature and simple effect and goal model. LP-TPOP differs from Zeno in following aspects:

1. Ground actions in LP-TPOP vs. partially instantiated operators in Zeno.
2. Zeno handles different (in)equation systems with different methods, i.e. Gaussian elimination for linear equations, linear programming for linear inequations. In LP-TPOP they are processed in one linear programming engine.
3. Zeno exploits Warshall algorithm for constraint management, while LP-TPOP uses negative cycle detection algorithm on a revised STN structure.

### 3 Components of LP-TPOP

The temporal model of LP-TPOP is real valued time-point model. A temporal interval is defined by two boundary time-points (or variables) of the form  $[s, e]$  (such that  $s \leq e$ , and all the intervals in LP-TPOP are assumed left-close right-open).

CBI (constraint based interval) action model specifies an action with a duration interval, and all preconditions and effects with intervals, which specifies the periods within which they should hold. An example of such actions is illustrated in Fig.1.

(: durative-action A#[t <sub>s</sub> , t <sub>e</sub> ) :duration (= ?duration 20) :condition (and (p1#[t <sub>s</sub> +2, t <sub>s</sub> +8) :effect (and ((not p2) [t <sub>s</sub> +10, t <sub>s</sub> +17)) (p3#[t <sub>s</sub> +5, t <sub>s</sub> +t <sub>p3</sub> ))))	(: durative-action B#[t <sub>i</sub> , t <sub>j</sub> ) :duration (= ?duration 15) :condition (and (p2#[t <sub>i</sub> , t <sub>i</sub> +15) :effect (and ((not p1) [t <sub>i</sub> +13, t <sub>i</sub> +t <sub>np1</sub> )) (p4#[t <sub>i</sub> +2, t <sub>i</sub> +t <sub>p4</sub> ))))
---	---

Fig. 1. Two CBI actions (in PDDL style)

An assumption should be placed that all the intervals within action should be included in the duration of the actions, e.g. in Fig.1  $[t_s+2, t_s+8] \subseteq [t_s, t_e]$  holds.

#### 3.1 Relaxed Temporal Planning Graph (RTPG) for Heuristic Extraction

In the selection of successive partial plan, LP-TPOP exploits similar heuristics extraction procedure via relaxed temporal planning graph, and termination criteria as SAPA [4], but with CBI actions. Generally they are different in two respects:

1. In LP-TPOP only temporal estimates are needed to be considered because it aims to compute plans with the least makespan (total times required to finish the plans).
2. The cost propagation in LP-TPOP is more complicated than that in SAPA because the action model of LP-TPOP is much more delicate than that of SAPA.

#### 3.2 Temporal Constraint Management

There are three types of constraints in LP-TPOP. They are constraints of the form  $[t_1, t_2] \subseteq [t_3, t_4]$  , (or  $t_3 \leq t_1 \wedge t_2 \leq t_4$  ), constraints of the form  $[t_1, t_2] \cap [t_3, t_4] = \emptyset$  , or  $(t_4 \leq t_1 \vee t_2 \leq t_3)$  , and constraints of the form  $t_x = t_1 \vee t_x = t_2 \vee \dots \vee t_x = t_k$  .

**Definition 1 (revised STN structure, r-STN).** A revised STN structure in LP-TPOP is a directed graph  $\langle V, E \rangle$ , where the variable  $V$  is knot set for temporal variables,  $E$  is a set of tuple  $\langle c_{xy}, weight(c_{xy}), tag(c_{xy}) \rangle$ , in which the component tag is used to mark whether the constraint  $c_{xy}$  appears in a disjunction or not:

$$tag(c_{xy}) = \begin{cases} \Lambda, & \text{if } c_{xy} \text{ doesn't appear as a disjunct;} \\ c_{uv}, & \text{if } c_{xy} \text{ appears in the disjunction } c_{xy} \vee c_{uv}. \end{cases} .$$

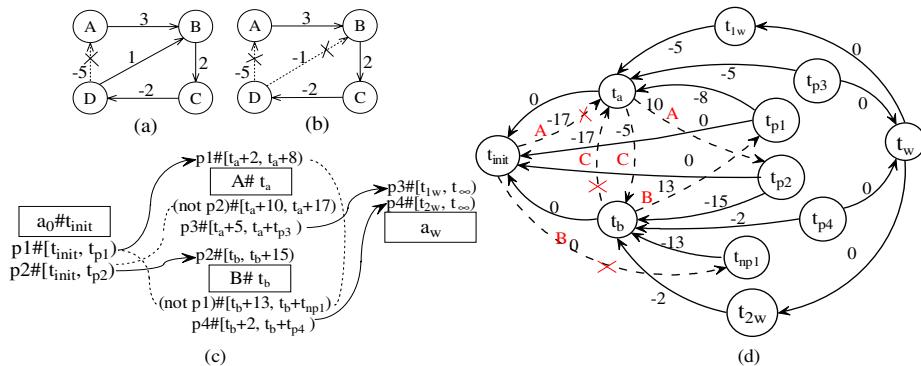
#### Consistency check and threat resolution in LP-TPOP

LP-TPOP checks the consistency of an r-STN by detecting all the negative cycles in it. When all the disjuncts in a disjunctive constraint are involved in negative cycles

the r-STN is inconsistent. Fig.2 (a) and (b) illustrates two inconsistent r-STNs. Take (b) of Fig.2 for example, all the two disjuncts are involved in negative cycles DABCD and DBCD respectively, which makes it inconsistent.

When some potential *threats* are detected in a partial plan, LP-TPOP takes some measures to resolve them. The basic idea is to depart the two intervals on which two conflict propositions are asserted.

Fig.2 (c) and (d) respectively illustrate a partial plan and related r-STN for the problem with those actions in Fig. 1, initial state  $\{p1#[t_{init}, t_{p1}], p2#[t_{init}, t_{p2}]\}$ , and goals  $\{p3#[t_{1w}, t_\infty], p4#[t_{2w}, t_\infty]\}$ . In (c) the solid lines and dotted ones with arrows notate the causal links and the threats respectively. From (d) it can be seen that three disjuncts labeled "X" are eliminated during planning procedure.



**Fig. 2.** Two inconsistent r-STN (a, b) and the exemplified partial plan and related r-STN (c, d)

### 3.3 Scheduling of Partial Plan

Once a consistent complete plan has been achieved, all the temporal constraints collected are passed to scheduling module (step 3.3) and 3.4) in LP-TPOP algorithm).

Tab.1 displays the constraint collected along the planning procedure w.r.t. the example of Fig.2. Once they are passed to scheduling module, an optimal solution  $t_a=15$ ,  $t_b=10$  can be calculated.

**Table 1.** Table of constraint additions in planning procedure

Action	Constraints
initial state	$t_{init} \leq t_{p1}$ , $t_{init} \leq t_{p2}$ , $t_{init} \leq t_{1w}$ , $t_{init} \leq t_{2w}$ , $t_{1w} \leq t_\infty$ , $t_{2w} \leq t_\infty$
adding A	$t_a + 5 \leq t_{p3}$ , $t_{init} \leq t_a$ , $t_a + 5 \leq t_{1w}$ , $t_\infty \leq t_{p3}$ , $t_a + 17 \leq t_{init} \vee t_{p2} \leq t_a + 10$
adding B	$t_b + 13 \leq t_{np1}$ , $t_b + 2 \leq t_{p4}$ , $t_{init} \leq t_b$ , $t_b + 2 \leq t_{2w}$ , $t_\infty \leq t_{p4}$ , $t_{np1} \leq t_{init} \vee t_{p1} \leq t_b + 13$ , $t_a + 17 \leq t_b \vee t_b + 15 \leq t_a + 10$
adding $a_0 \rightarrow A$	$t_{init} \leq t_a + 2$ , $t_a + 8 \leq t_{p1}$
adding $a_0 \rightarrow B$	$t_{init} \leq t_b$ , $t_b + 15 \leq t_{p2}$
supplement at end	$t_{p1} = t_b + 13$ , $t_{p2} = t_a + 10$

### 3.4 Formal Properties of LP-TPOP

Similar to the definition of partial plan in POCL planning [7], a partial plan P in LP-TPOP is a tuple  $\langle A, r\text{-STN}, CL, TDB \rangle$ , where A is action set with each action attached to a starting point, r-STN maintains the temporal constraints, CL is the set of causal links, and TDB is the set of propositions with each attached to an interval.

**Definition 2 (Validity of partial plan).** A partial plan  $P = \langle A, r\text{-STN}, CL, TDB \rangle$  is a valid plan for a planning problem if there's no open goal in P and r-STN is consistent.

**Proposition 1 (Soundness and completeness of LP-TPOP).** LP-TPOP is sound, that is, whenever LP-TPOP returns a plan, it is a valid plan. LP-TPOP is complete, in the sense that whenever there is a valid plan (schedule) for a planning problem P, LP-TPOP will find one.<sup>1</sup>

## 4 Conclusions and Future Work

In this paper, a sound and complete planning algorithm named LP-TPOP integrating planning with scheduling through temporal constraint management is presented. LP-TPOP operates on ground CBI actions in POCL planning framework. While the schedule may be not optimal globally, it is locally optimal w.r.t. a given plan.

While the planner is still under implementation, it is sound and complete, and has moderate efficiency. There's still much future work to do to further optimize LP-TPOP, e.g. more effective heuristics to tackle large branching factor, more features support (e.g. resource, processes, events, etc.) and so on.

## References

1. A. Cesta, F. Pecora, and R. Rasconi. Biasing the Structure of Scheduling Problems Through Classical Planners. In (Ed.), Proceedings of WIPIS-04, ICAPS Workshop on “Integrating Planning into Scheduling”, Whistler, British Columbia, Canada, (2004) 3-7
2. Maria Fox and Derek Long, PDDL2.1: An extension to PDDL for expressing temporal planning domains. In Journal of artificial intelligence research Vol. 20 (2003) 61-124
3. K. Halsey, D. Long, and M. Fox Isolating where Planning and Scheduling Interact, In Proceedings from the 22nd UK Planning and Scheduling Special Interest Group (PlanSIG'03). (2003) 104-114
4. Do, M. and Kambhampati, S. SAPA: A Multi-objective Metric Temporal Planner. In Journal of Artificial Intelligence Research Vol. 20 (2003) 155-194
5. D. Smith, J. Frank, and A. Jónsson, Bridging the Gap Between Planning and Scheduling, in Knowledge Engineering Review Vol. 15, no. 1 (2000) 15:1
6. J. Scott Penberthy, Daniel S. Weld, Temporal planning with continuous change. In Proceedings of AAAI-94 (1994) 1010-1015
7. D. S. Weld. An introduction to least commitment planning. AI Magazine, 15(4), (1994) 27-61

---

<sup>1</sup> However, we do not present the complete proof in this paper for space limitation.

# Integrating Insurance Services, Trust and Risk Mechanisms into Multi-agent Systems

Yuk-Hei Lam<sup>1</sup>, Zili Zhang<sup>1,2</sup>, and Kok-Leong Ong<sup>1</sup>

<sup>1</sup> School of Engineering and Information Technology  
Deakin University

Waurn Ponds, Victoria 3217, Australia

{yuk, zhang, leong}@deakin.edu.au

<sup>2</sup> Faculty of Computer and Information Science  
Southwest University  
Chongqing 400715, China

**Abstract.** In multi-agent systems, there is often the need for an agent to cooperate with others so as to ensure that a given task is achieved timely and cost effectively. Currently multi-agent systems maximize this through mechanisms such as coalition formation, trust and risk assessments, etc. In this paper, we incorporate the concept of insurance with trust and risk mechanisms in multi-agent systems. The novelty of this proposal is that it ensures continuous sharing of resources while encouraging expected utility to be maximized in a dynamic environment. Our experimental results confirm the feasibility of our approach.

## 1 Introduction

In multi-agent systems, software agents are often required to seek external help when the allocated task can not be accomplished on their own. Even though agents are mostly self-interested, they tend to cooperate with other agents in order to improve their global and individual performance [1]. However, it is often difficult for agents to discover or search for the required resources<sup>1</sup>, e.g., manpower, platform access time, services such as a particular skill, etc., in such dynamic environments to achieve their objectives.

Currently, agent systems solve this problem using mechanisms such as insurance services [2], negotiation [3], and trust and risk assessments [4,5]. In this paper, we integrate our insurance mechanism [2] with the trust and risk mechanisms. With insurance, the insured agents are better guaranteed to have the requested resources in execution without any extra effort in negotiation and resource discovery. With trust management, agents can evaluate the trustworthiness of potential collaborators through their own past experience and through the acquisition of another agent's reputation provided by the insurance agents. With risk management, agents can make decisions on purchasing insurance according to their own personal risk attitude.

---

<sup>1</sup> We define resources as anything that agents are willing to share or exchange with other agents.

The remaining sections of this paper are organized as follows. In the next section, we define the concept of insurance and the trustworthiness of the potential partners will be examined. We then discuss how agents reason about demanding and purchasing insurance rationally in Section 3. In Section 4, we present the simulation results and the conclusions are laid out in Section 5.

## 2 The Insurance Model

Without loss of generality, we assume a multi-agent system operating in a networked environment, in which self-interested agents  $\mathcal{A}_1, \dots, \mathcal{A}_n$  are required to complete a set of assigned tasks,  $T = \mathcal{T}_1, \dots, \mathcal{T}_m$ , where  $n \gg m$ . Moreover, each agent owns a set of resources,  $R = \mathcal{R}_1, \dots, \mathcal{R}_k$ , which is used for executing any given task. We further define that  $\mathcal{I}_1, \dots, \mathcal{I}_p$  be a set of insurance agents who provide insurance services in the system. Due to space limitation, this paper only concentrate on the integration of trust mechanism with the existing insurance model.

### 2.1 Local Reputation Ratings

Consider the situation where the insurance agent  $\mathcal{I}_i$  maintains a list of agents,  $PA = \{\mathcal{A}_j, \dots, \mathcal{A}_k\}$ , who are capable of providing the requested resource to the insured agent  $\mathcal{A}_i$ . It is therefore natural for the insured agent  $\mathcal{A}_i$  to seek someone who is considered the most trustworthy, i.e., whose performance has been consistent over many transactions. In this case, the insured agent  $\mathcal{A}_i$  can obtain a general local reputation of each potential collaborative partner (i.e.,  $\mathcal{A}_j \in PA$ ), which leads us to the following:

**Definition 1.** Let  $\mathcal{R}^L(\mathcal{A}_j)$  denote average the local reputation of agent  $\mathcal{A}_j$  from agent  $\mathcal{A}_i$ . Thus,  $\mathcal{R}^L$  is an indication of the trust assigned by  $\mathcal{A}_i$  to  $\mathcal{A}_j$  and for a given set of  $\ell$  transactions with  $\mathcal{A}_i$ ,  $\mathcal{R}^L(\mathcal{A}_j)$  is given as:

$$\mathcal{R}^L(\mathcal{A}_j) = \frac{1}{\ell} \sum_{r=1}^{\ell} \mathcal{R}^L(\mathcal{A}_j, r) \quad (1)$$

where  $\mathcal{R}^L(\mathcal{A}_j, r) \in [0, 1]$  and  $r$  denotes the  $r^{th}$  transaction of  $\mathcal{A}_i$  with  $\mathcal{A}_j$ .

The local reputation of an agent  $\mathcal{A}_j$  is assigned by an agent  $\mathcal{A}_i$ , which forms the personal view of agent  $\mathcal{A}_i$  about the average past performance of agent  $\mathcal{A}_j$ . An agent  $\mathcal{A}_i$  can then make use of this information to predict the future behaviour of agent  $\mathcal{A}_j$ . However, if the agent  $\mathcal{A}_i$  has not conducted any trasactions with the agent  $\mathcal{A}_j$  before, then the agent  $\mathcal{A}_i$  should only consider the global reputation rating of agent  $\mathcal{A}_j$  instead.

### 2.2 Global Reputation Rating

When one is in lack of information of other entities, it is hard to decide who to trust. The word-of-mouth method provides a way for entities to gain knowledge

or to get advice using the testimonies of third parties. Testimonies from other trusted parties are mainly used in the environments where entities have little or no prior knowledge of the potential collaborative partner(s). By using the insurance agents as the third trusted parties who propagate the global reputation rating of agents, the insured agents will be guaranteed to obtain the same reputation's information about the other agents regardless the number of friends they have.

In order for the insurance agents to take the role of the trusted third parties, they need to gather the local reputation rating from the insured agents so as to form the average global reputation of agents,  $\mathcal{R}^G(\cdot)$ . The global view of reputation about agent  $\mathcal{A}_j$  is calculated as

$$\mathcal{R}^G(\mathcal{A}_j) = \frac{1}{\ell} \sum_{n=1}^{\ell} \mathcal{R}^L(\mathcal{A}_j, n) \quad (2)$$

where  $n$  denotes the  $n^{th}$  agent who has submitted reputation rating of the agent  $\mathcal{A}_i$  to the insurance agent  $\mathcal{I}_i$  and  $\ell$  denotes the total number of submissions. In other words, the insurance agent  $\mathcal{I}_i$  collected the reputation ratings submitted by other agents who has made transactions with the agent  $\mathcal{A}_j$  previously.

### 2.3 Trust Degree

Once the values of local and global reputation about each potential collaborative partner are obtained, the insured agent  $\mathcal{A}_i$  is now required to quantify the amount of trust it has on all the potential collaborative partners.

If an agent  $\mathcal{A}_i$  evaluates the trustworthiness of the agent  $\mathcal{A}_j$ , we have the global reputation rating  $\mathcal{R}^G(\mathcal{A}_j)$ , which is a score collected by the insurance agents about  $\mathcal{A}_j$ . At the same time, every agent has its own score for  $\mathcal{A}_j$ 's reputation based on its own experience, i.e.,  $\mathcal{R}^L(\mathcal{A}_j)$ . Therefore, when an agent decides to initiate a new transaction using insurance, it needs to reconcile its knowledge (i.e.,  $\mathcal{R}^L(\mathcal{A}_j)$ ) about  $\mathcal{A}_j$  with the global score (i.e.,  $\mathcal{R}^G(\mathcal{A}_j)$ ) before passing a judgement.

**Definition 2.** Given agents  $\mathcal{A}_i$  and  $\mathcal{A}_j$ , the trust degree  $\mathcal{T}_{\mathcal{D}}$  quantifies the amount of trust  $\mathcal{A}_i$  has on  $\mathcal{A}_j$ , based on  $\mathcal{A}_j$ 's local reputation,  $\mathcal{R}^L(\mathcal{A}_j)$ , held by the agent  $\mathcal{A}_i$ , and the global reputation,  $\mathcal{R}^G(\mathcal{A}_j)$ , provided by the insurance agent  $\mathcal{I}_i$ .  $\mathcal{T}_{\mathcal{D}_i}$  is given as:

$$\mathcal{T}_{\mathcal{D}}(\mathcal{A}_j) = w_1 * \mathcal{R}^G(\mathcal{A}_j) + w_2 * \mathcal{R}^L(\mathcal{A}_j) \quad (3)$$

where  $w_1$  and  $w_2$  are the weight given by agent  $\mathcal{A}_i$  where  $w_1 + w_2 = 1$ .

The purpose of weighting  $\mathcal{R}^G$  and  $\mathcal{R}^L$  is to obtain the appropriate broad-sense trust degree  $\mathcal{T}_{\mathcal{D}}$  that is relevant to the context of a given situation. If an agent relies more on its local reputation rating, then it can adjust the weight

accordingly (i.e.,  $w_1 > w_2$ ). In contrast, an agent might rely on the global reputation rating if it has little or even no prior knowledge of the potential collaborative partner (i.e.,  $w_1 < w_2$ ). Furthermore, if an agent has no past experience with the potential collaborative partners, then the weight can be set as  $w_1 = 1$  accordingly.

To decide which agent is the most trustworthy, the insured agents are required to select the agent with the highest trust degree,  $T_{\mathcal{D}}(\cdot)$ , among the potential collaborative partners, i.e.,

$$\{p \in \text{PA} \mid \forall g \in \text{PA}, p \neq g, T_{\mathcal{D}}(p) > T_{\mathcal{D}}(g)\} \quad (4)$$

### 3 Reasoning Insurance Purchase with Trust

We have so far considered the beneficial of the insurance concept and trust issues in multi-agent systems. However, agents need not purchase insurance for every transaction. In fact, they must reason about the effectiveness of purchasing insurance in different circumstances. In [2], evaluation methods are based on the criticality of task, financial aspects and risk assessment. In this paper, trust issues are also considered to enrich the decision making.

If the potential collaborative partner is not trustworthy from the insured agents perspective, there is no point for the insured agents to commit the insurance contract from the beginning. Therefore, it is crucial for the insured agents to evaluate the trustworthiness of the potential collaborative partners that gives an indication of how they will be performing in the future. Although Equation (3) enable the insured agents to identify the most trustworthy one among the potential collaborative partners, the criteria of being a trustworthy agent is different from agents to agents. As a result, the trust weight threshold  $\Omega(\mathcal{A}_i)$  is set to determine the level of trust accepted by the agent  $\mathcal{A}_i$ . In this case, an agent  $\mathcal{A}_i$  who wishes to evaluate the trustworthiness of agent  $\mathcal{A}_j$  will therefore test for the following rule:

$$\text{if } T_{\mathcal{D}}(\mathcal{A}_j) \geq \Omega(\mathcal{A}_i), \text{ then agent } \mathcal{A}_j \text{ is trustworthy.} \quad (5)$$

The weighting of  $\Omega(\mathcal{A}_i)$  depends on the personal attitude of agents. If an agent is risk neutral, the value of  $\Omega(\mathcal{A}_i)$  will be neutral (i.e., the median of the trust degree,  $T_{\mathcal{D}}$ ). For risk averse agents, the value of  $\Omega(\mathcal{A}_i)$  will be higher than the risk neutral agents as this kind of agents prefer less risk involved in transactions. Alternatively, the risk seeking agents rather take risk, so that it is still acceptable if the value of  $\Omega(\mathcal{A}_i)$  is lower than the average.

### 4 Empirical Results

Experiments have been carried out in order to verify the effectiveness and benefits of applying insurance concept in the multi-agent system. Due to space limitation, we only report a summary of our results here. The full details can be obtained from [6].

From the experimental results, we can conclude that insurance concept helps to stabilize the performance of agents, enable better utilization of resources, and thus increase the throughput of the whole system. In general, agents with insurance have more stable performance and they often can reach a higher success rate. Second, insurance concept works well in all different environments (from only 10% of resources to 90% resources available in the system): agents can still maintain a high success rate with insurance in all cases. Although the reputation evaluation enables agents maximizing the chances in dealing with the most trustworthy agent, it is evidence that the reputation evaluation also destroys most of the cooperation opportunities set by the insurance agents.

## 5 Conclusions

The concept of insurance with trust and risk mechanism provides agents an alternative approach to allocating resources and to avoiding risky situations. The demonstration of the use of insurance are addressed in Appendix A. From our initial simulation results, we have shown the beneficial of applying the insurance, trust and risk concept in multi-agent systems and thus have evidence to support the feasibility of our proposal.

## References

1. Breban, S., Vassileva, J.: A coalition formation mechanism based on inter-agent trust relationships. In: International Conference on Autonomous Agents and Multiagents Systems '02, New York, NY, USA (2002) 306–307
2. Lam, Y., Zhang, Z., Ong, K.: Insurance services in multi-agent systems. In: Proceedings of the 18th Australian Joint Conference on Artificial Intelligence. (2005) 664–673
3. Luo, X., Jennings, N.R., Shadbolt, N., fung Leung, H., man Lee, J.H.: A fuzzy constraint-based knowledge model for bilateral, multi-issue negotiations in semi-competitive environments. *Artificial Intelligence* **148** (2003) 53–102
4. He, M., Jennings, N.R., Leung, H.: On agent-mediated electronic commerce. *IEEE Trans on Knowledge and Data Engineering* **15** (2003) 985–1003
5. He, M., Leung, H., Jennings, N.R.: A fuzzy logic based bidding strategy for autonomous agents in continuous double auctions. *IEEE Trans on Knowledge and Data Engineering* **15** (2003) 1345–1363
6. Lam, Y., Zhang, Z., Ong, K.: Integrating insurance services, trust and risk mechanisms into multi-agent systems. Technical Report TRC06/07, School of Engineering and Information Technology, Deakin University, <http://www.deakin.edu.au/~yuk/TechReports/TrustInsuranceModel06.pdf> (2006)

# Cat Swarm Optimization

Shu-Chuan Chu<sup>1</sup>, Pei-wei Tsai<sup>2</sup>, and Jeng-Shyang Pan<sup>2</sup>

<sup>1</sup> Department of Information Management,  
Cheng Shiu University

<sup>2</sup> Department of Electronic Engineering,  
National Kaohsiung University of Applied Sciences

**Abstract.** In this paper, we present a new algorithm of swarm intelligence, namely, Cat Swarm Optimization (CSO). CSO is generated by observing the behaviors of cats, and composed of two sub-models, i.e., tracing mode and seeking mode, which model upon the behaviors of cats. Experimental results using six test functions demonstrate that CSO has much better performance than Particle Swarm Optimization (PSO).

## 1 Introduction

In the field of optimization, many algorithms were being proposed recent years, e.g. Genetic Algorithm (GA) [1-2], Ant Colony Optimization (ACO) [6-7], Particle Swarm Optimization (PSO) [3-5], and Simulated Annealing (SA) [8-9] etc. Some of these optimization algorithms were developed based on swarm intelligence. Cat Swarm Optimization (CSO), the algorithm we proposed in this paper, is motivated from PSO [3] and ACO [6].

According to the literatures, PSO with weighting factor [4] usually finds the better solution faster than the pure PSO, but according to the experimental results, Cat Swarm Optimization (CSO) presents even much better performance.

Via observing the behavior of creatures, we may get some idea for solving the optimization problems. By studying the behavior of ants achieves ACO, and with examining the movements of the flocking gulls realizes PSO. Through inspecting the behavior of cat, we present Cat Swarm Optimization (CSO) algorithm.

## 2 Behaviors of Cats

According to the classification of biology, there are about thirty-two different species of creatures in feline, e.g. lion, tiger, leopard, cat etc. Though they have different living environments, there are still many behaviors simultaneously exist in most of felines.

In spite of the hunting skill is not innate for felines, it can be trained to acquire. For the wild felines, the hunting skill ensures the survival of their races, but for the indoor cats, it exhibits the natural instinct of strongly curious about any moving things. Though all cats have the strong curiosity, they are, in most times, inactive. If you spend some time to observe the existence of cats, you may easily find that the cats spend most of the time when they are awake on resting.

The alertness of cats are very high, they always stay alert even if they are resting. Thus, you can simply find that the cats usually looks lazy, lying somewhere, but opening their eyes hugely looking around. On that moment, they are observing the environment. They seem to be lazy, but actually they are smart and deliberate.

Of course, if you examine the behaviors of cats carefully, there would be much more than the two remarkable properties, which we discussed in the above.

### 3 Proposed Algorithm

In our proposed Cat Swarm Optimization, we first model the major two behaviors of cats into two sub-models, namely, seeking mode and tracking mode. By the way of mingling with these two modes with a user-defined proportion, CSO can present better performance.

#### 3.1 The Solution Set in the Model -- Cat

No matter what kind of optimization algorithm, the solution set must be represented via some way. For example, GA uses chromosome to represent the solution set; ACO uses ant as the agent, and the paths made by the ants depict the solution sets; PSO uses the positions of particles to delineate the solution sets. In our proposed algorithm, we use cats and the model of behaviors of cats to solve the optimization problems, i.e. we use cats to portray the solution sets.

In CSO, we first decide how many cats we would like to use, then we apply the cats into CSO to solve the problems.

Every cat has its own position composed of  $M$  dimensions, velocities for each dimension, a fitness value, which represents the accommodation of the cat to the fitness function, and a flag to identify whether the cat is in seeking mode or tracing mode. The final solution would be the best position in one of the cats due to CSO keeps the best solution till it reaches the end of iterations.

#### 3.2 Seeking Mode

This sub-model is used to model the situation of the cat, which is resting, looking around and seeking the next position to move to. In seeking mode, we define four essential factors: seeking memory pool (SMP), seeking range of the selected dimension (SRD), counts of dimension to change (CDC), and self-position considering (SPC).

SMP is used to define the size of seeking memory for each cat, which indicates the points sought by the cat. The cat would pick a point from the memory pool according to the rules described later.

SRD declares the mutative ratio for the selected dimensions. In seeking mode, if a dimension is selected to mutate, the difference between the new value and the old one will not out of the range, which is defined by SRD.

CDC discloses how many dimensions will be varied. These factors are all playing important roles in the seeking mode.

SPC is a Boolean variable, which decides whether the point, where the cat is already standing, will be one of the candidates to move to. No matter the value of SPC

is true or false; the value of SMP will not be influenced. How the seeking mode works can be described in 5 steps as follows:

- Step1: Make  $j$  copies of the present position of  $\text{cat}_k$ , where  $j = \text{SMP}$ . If the value of SPC is true, let  $j = (\text{SMP}-1)$ , then retain the present position as one of the candidates.
- Step2: For each copy, according to CDC, randomly plus or minus SRD percents of the present values and replace the old ones.
- Step3: Calculate the fitness values ( $FS$ ) of all candidate points.
- Step4: If all  $FS$  are not exactly equal, calculate the selecting probability of each candidate point by equation (1), otherwise set all the selecting probability of each candidate point be 1.
- Step5: Randomly pick the point to move to from the candidate points, and replace the position of  $\text{cat}_k$ .

$$P_i = \frac{|FS_i - FS_b|}{FS_{\max} - FS_{\min}}, \text{ where } 0 < i < j \quad (1)$$

If the goal of the fitness function is to find the minimum solution,  $FS_b = FS_{\max}$ , otherwise  $FS_b = FS_{\min}$ .

### 3.3 Tracing Mode

Tracing mode is the sub-model for modeling the case of the cat in tracing some targets.

Once a cat goes into tracing mode, it moves according to its' own velocities for every dimension. The action of tracing mode can be described in 3 steps as follows:

- Step1: Update the velocities for every dimension ( $v_{k,d}$ ) according to equation (2).
- Step2: Check if the velocities are in the range of maximum velocity. In case the new velocity is over-range, set it be equal to the limit.
- Step3: Update the position of  $\text{cat}_k$  according to equation (3).

$$v_{k,d} = v_{k,d} + r_1 \times c_1 \times (x_{best,d} - x_{k,d}), \text{ where } d = 1, 2, \dots, M \quad (2)$$

$x_{best,d}$  is the position of the cat, who has the best fitness value;  $x_{k,d}$  is the position of  $\text{cat}_k$ .  $c_1$  is a constant and  $r_1$  is a random value in the range of  $[0,1]$ .

$$x_{k,d} = x_{k,d} + v_{k,d} \quad (3)$$

### 3.4 Cat Swarm Optimization

As we described in the above subsection, CSO includes two sub-models, the seeking mode and the tracing mode. To combine the two modes into the algorithm, we define a mixture ratio (MR) of joining seeking mode together with tracing mode.

By observing the behaviors of cat, we notice that cat spends most of the time when they are awake on resting. While they are resting, they move their position carefully and slowly, sometimes even stay in the original position. Somehow, for applying this behavior into CSO, we use seeking mode to represent it.

The behavior of running after targets of cat is applied to tracing mode. Therefore, it is very clear that MR should be a tiny value in order to guarantee that the cats spend most of the time in seeking mode, just like the real world.

The process of CSO can be described in 6 steps as follows:

Step1: Create  $N$  cats in the process.

Step2: Randomly sprinkle the cats into the  $M$ -dimensional solution space and randomly select values, which are in-range of the maximum velocity, to the velocities of each cat. Then haphazardly pick number of cats and set them into tracing mode according to MR, and the others set into seeking mode.

Step3: Evaluate the fitness value of each cat by applying the positions of cats into the fitness function, which represents the criteria of our goal, and keep the best cat into memory. Note that we only need to remember the position of the best cat ( $x_{best}$ ) due to it represents the best solution so far.

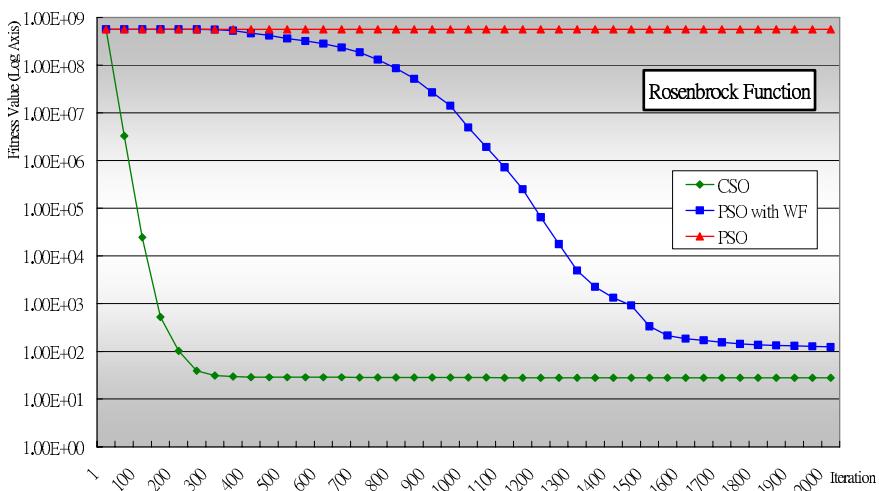
Step4: Move the cats according to their flags, if  $cat_k$  is in seeking mode, apply the cat to the seeking mode process, otherwise apply it to the tracing mode process. The process steps are presented above.

Step5: Re-pick number of cats and set them into tracing mode according to MR, then set the other cats into seeking mode.

Step6: Check the termination condition, if satisfied, terminate the program, and otherwise repeat step3 to step5.

## 4 Experimental Results

We applied CSO, PSO and PSO with weighting factor into six test functions to compare the performance. All the experiments demonstrate the proposed Cat Swarm Optimization (CSO) is superior to PSO and PSO with weighting factor. Due to the space limit of this paper, only the experimental results of test function one shown in Fig. 1.



**Fig. 1.** The experimental result of test function 1

## References

1. Goldberg, D.E.: Genetic Algorithm in Search. Optimization and Machine Learning. Addison-Wesley Publishing Company (1989)
2. Pan, J. S., McInnes, F. R., Jack, M. A. : Application of Parallel Genetic Algorithm and Property of Multiple Global Optima to VQ Codevector Index Assignment. Electronics Letters 32(4) (1996) 296-297
3. Eberhart, R., Kennedy, J.: A new optimizer using particle swarm theory. Sixth International Symposium on Micro Machine and Human Science (1995) 39-43
4. Shi, Y., Eberhart, R.: Empirical study of particle swarm optimization. Congress on Evolutionary Computation. (1999) 1945-1950
5. Chang, J. F., Chu, S. C., Roddick, J. F., Pan, J. S. : A Parallel Particle Swarm Optimization Algorithm with Communication Strategies. Journal of Information Science and Engineering 21(4) (2005) 809-818
6. Dorigo, M., Gambardella, L. M.: Ant colony system: a cooperative learning approach to the traveling salesman problem. IEEE Trans. on Evolutionary Computation. 26 (1) (1997) 53-66
7. Chu, S. C., Roddick, J. F., Pan, J. S.: Ant colony system with communication strategies. Information Sciences 167 (2004) 63-76
8. Kirkpatrick, S., Gelatt, Jr. C.D., Vecchi, M.P.: Optimization by simulated annealing. Science (1983) 671-680
9. Huang, H. C., Pan, J. S., Lu, Z. M., Sun, S. H., Hang, H.M.: Vector quantization based on generic simulated annealing. Signal Processing 81(7) (2001) 1513-1523

# Heuristic Information Based Improved Fuzzy Discrete PSO Method for Solving TSP

Bin Shen, Min Yao, and Wensheng Yi

College of Computer Science, Zhejiang University, Hangzhou, 310027, P.R. China  
tsingbin@zju.edu.cn, myao@zju.edu.cn

**Abstract.** In this paper, we propose an improved fuzzy discrete Particle Swarm Optimization method (IFD-PSO), and apply this method to TSP. We use fuzzy matrix space to represent the corresponding TSP solution, and bring forward the transformation method of fuzzy matrix space. Heuristic information is employed to improve the convergence speed. The experiment results show that IFD-PSO has a better performance and achieves satisfactory effect.

## 1 Introduction

Recently, there have been several work on solving Traveling Salesman Problem (TSP) using Particle Swarm Optimization (PSO)<sup>[1,2,3,4]</sup> algorithm. Reference [2], [3], [4] have proposed Discrete PSO method based on swap operation. The disadvantage of this method is that it is easy to get into the local optimization comparing with the basic PSO. Wei Pang et al.<sup>[1]</sup> apply a fuzzy discrete PSO (FD-PSO) to solve TSP, and can achieve satisfactory effect. In order to further improve the performance of solving TSP using PSO, we propose an improved fuzzy discrete PSO algorithm (IFD-PSO).

## 2 Fuzzy Matrix Representing TSP Solution

### 2.1 Construction of Fuzzy Matrix

Assume the solution of TSP is  $S = \{s_1, s_2, \dots, s_n\} = \{(s_1, s_2), (s_2, s_3), \dots, (s_{n-1}, s_n), (s_n, s_1)\}$ , where  $n$  is the number of cities,  $s_i (i = 1, \dots, n)$  is the  $i$ -th visited city node in this solution, and  $(s_1, s_2), (s_2, s_3), \dots, (s_{n-1}, s_n), (s_n, s_1)$  are the visited directed edges in turn. Then fuzzy matrix  $R$  can be represented as  $R = (r_{ij})_{n,n}$ , where,  $r_{ij} \in [0,1] (i, j = 1, \dots, n)$  means the possibility of choosing directed edge  $(i, j)$ , after city  $i$  has been chosen in TSP solution. In order to avoid directed edge  $(i, i) (i = 1, \dots, n)$  appearing in TSP solution, we need to set the elements in diagonal of fuzzy matrix enough small, thus let  $r_{ii} (i = 1, \dots, n) = -Max$ . Thus the fuzzy matrix representing TSP solution can be constructed.

### 2.2 Transforming Fuzzy Matrix into TSP Solution

While a  $n \times n$  fuzzy matrix  $R$  is gotten, it needs to be defuzzified to obtain the corresponding TSP solution. Firstly,  $R$  is defuzzified according to global maximum

method and first row maximum method respectively. Then we would like to compare the fitness of two solutions, and choose the better one as our final solution. Global maximum method and first row maximum method are listed below, where a flag array of  $n$  bits is set to record whether the corresponding column has been selected or not:

### **Method 1.** Global maximum method

Step 1. Initialize the flag array as all unselected.

Step 2. Choose the maximum element in fuzzy matrix  $R$ , as might well be  $r_{s_1 s_2}$ , which represents the directed edge  $(s_1, s_2)$  has been selected. We need to mark the flags of column  $s_1$  and column  $s_2$  in flag array as “selected”, to avoid returning to the visited cities without visiting all of the cities.

Step 3-1. Choose the maximum element among unselected columns in row  $s_2$  of fuzzy matrix  $R$ , and let it be  $r_{s_2 s_3}$ . Thus directed edge  $(s_2, s_3)$  is selected as the subsequence edge of  $(s_1, s_2)$ . Similarly, we should mark column  $s_3$  as “selected”.

Step 3-2. Repeat Step 3-1 and obtain directed edges  $(s_3, s_4), \dots, (s_{n-1}, s_n)$  in turn. At the same time, this TSP solution visits every city once and only once.

Step 4. Complete the TSP solution with directed edge  $(s_n, s_1)$ , which means traveler returns to the start from the last visited city.

Thus, we achieve the solution  $(s_1, s_2), (s_2, s_3), \dots, (s_{n-1}, s_n), (s_n, s_1)$ .

### **Method 2.** First row maximum method

We only need to change Step2 of Method 1, where “in fuzzy matrix  $R$ ” is instead with “in the first row of fuzzy matrix  $R$ ”. Thus Method 2 is obtained.

## 3 Heuristic Information Based Improved Fuzzy Discrete PSO

### 3.1 Symbol Definitions

**Definition 1.** Fuzzy matrix position (FM-position for short)  $X = (r_{ij})_{n,n}$ , whose diagonal elements are  $-Max$ , represents a corresponding solution in TSP solution space.

**Definition 2.** Fuzzy matrix velocity (FM-velocity for short)  $V = (v_{ij})_{n,n}$ , whose diagonal elements are 0, represents the change of FM-position. It can keep the diagonal elements of fuzzy matrix  $X$  being  $-Max$  among the iterative operations.

**Definition 3.** The addition and subtraction of basic PSO are redefined as the addition and subtraction between matrixes, which are signed as  $\oplus$  and  $\ominus$ ; The multiplication is redefined as every element in  $R$  multiplied by  $\alpha$ , marked as  $\alpha \otimes R$ .

### 3.2 Initialization Using Heuristic Information

In TSP, we know the following heuristic information: the smaller the distance (cost) between two cities, the more the possibility of directed edge between these two cities being selected. Let the distance matrix between cities is  $D = (d_{ij})_{n,n}$ , where,

$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$ ,  $i, j = 1, \dots, n$ ,  $x_i, y_i$  are the X-coordinate and Y-coordinate of city  $i$ . It is obvious that  $d_{ii} = 0, i = 1, \dots, n$ . Because every element in FM-position denotes the possibility of the corresponding edge being selected, FM-position  $X$  is initialized as follows. First, set the diagonal elements as  $-Max$ , then set other element  $x_{ij}$  ( $i, j = 1, \dots, n; i \neq j$ ) as  $1/d_{ij}$ . Thus particles can quickly converge to the local minimum, and try to find the better one. To keep the diversity of particles, we only need to set one particle at the local minimum. Every element except diagonal element in  $v_{ij}$ , ( $i, j = 1, \dots, n; i \neq j$ ) is initialized according to  $-V_{\max} \leq v_{ij} \leq V_{\max}$ , where  $V_{\max}$  is the control parameter.

### 3.3 Constraints of Standardization

In order to keep the comparability of FM-velocity and FM-position during iterative operations, standard FM-position  $X = (x_{ij})_{n,n}$  should satisfy the following conditions:

$$\sum_{j=1; j \neq i}^n x_{ij} = 1, (i = 1, \dots, n) \quad (1) \quad x_{ij} \in [0,1], (i, j = 1, \dots, n; i \neq j) \quad (2) \quad x_{ii} = -Max, (i = 1, \dots, n) \quad (3)$$

Standard FM-velocity  $V = (v_{ij})_{n,n}$  should satisfy the following conditions:

$$\sum_{j=1; j \neq i}^n v_{ij} = 0, (i = 1, \dots, n) \quad (4) \quad -V_{\max} \leq v_{ij} \leq V_{\max}, (i, j = 1, \dots, n; i \neq j) \quad (5) \quad v_{ii} = 0, i = 1, \dots, n \quad (6)$$

We can prove that once FM-position satisfies condition (3) and FM-velocity satisfies condition (6) respectively, we needn't adjust them again to satisfy condition (3) and condition (6) in iterative operations. The proof is similar to the proof in FD-PSO's initialization process<sup>[1]</sup>. We omit it for the limitation of space. We also can prove that once FM-position satisfies condition (1) and FM-velocity satisfies condition (4) respectively, we needn't adjust them again to satisfy condition (1) and condition (4) (except condition (2) or condition (5) is violated). The proof is omitted here.

### 3.4 Standardization of FM-Position and FM-Velocity

After the initialization of FM-position or several iterative operations, condition (2) may be violated, thus it is necessary to standardize the FM-position again. The standardization method of FM-position is shown as below:

Step 1. Check whether or not there exist elements except in the diagonal which is less than 0. If yes, we should set them as 0.

Step 2. Set the diagonal elements of FM-position as the value  $-Max$ , and let every other element  $x_{jk}$  ( $j, k = 1, \dots, n; j \neq k$ ) be  $x_{jk} / \sum_{i=1; i \neq j}^n x_{ji}$ . Thus the transformed FM-position can satisfy condition (1), (2), (3).

After the initialization of FM-velocity or several iterative operations, condition (5) may be violated, so it is necessary to standardize the FM-velocity again. The standardization method of FM-velocity is shown as following:

Step 1. Check whether or not there exist elements violating condition (5). If yes, we set the element which is less than  $-V_{\max}$  as  $-V_{\max}$ , and the element which is big than  $V_{\max}$  as  $V_{\max}$ . Therefore we have  $-V_{\max} \leq v_{ij} \leq V_{\max}$ .

Step 2. Let the diagonal elements hold the original value, and transform every other element  $v_{jk}$  ( $j, k = 1, \dots, n; j \neq k$ ) into  $\frac{v_{jk}}{2} - \frac{1}{n-1} \sum_{i=1, i \neq j}^n \frac{v_{ji}}{2}$ .

Obviously, after the above transformation, the FM-velocity satisfies condition (4) and (6). We also can proof that it satisfies condition (5). The proof is omitted here.

### 3.5 Description of the Algorithm

Step 1. Set the population size as  $Num$ , and set the maximum number of generations as  $MaxNum$ . Initialize and standardize the FM-position and the FM-velocity. Then the local best  $pbest_i = X_0$ , and the global best  $gbest_i$  is set as the best position among  $pbest_i$ .

Step 2. If current iteration number is equal to  $MaxNum$ , go to step 5.

Step 3. Calculate the new FM-position and FM-velocity for all particles.

Compute the new velocity. If the new velocity is against condition (5), standardize the velocity using our proposed method. Calculate the new position. If the new position goes against constraint (2), standardize it using our proposed method. Obtain the fitness of new position. If the fitness is better than that of the local best of the particle, update the local best position with the new position.

Step 4. If there exist some particles, whose fitness of local best position is better than that of the global best, update the global best position with the best one of local best positions. Then go to step 2.

Step 5. Output the global best position, the solution, and its fitness value.

## 4 Experiment Results

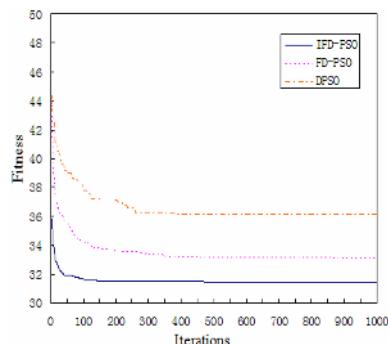
We test IFD-PSO using Burma14 in TSPLib, and compare the experiment results with DPSO<sup>[4]</sup> and FD-PSO<sup>[1]</sup>. The experiment is made on PC (AMD Sempron 2400+, 512M RAM, WinXP OS, matlab7.0). We set the population size as 100, the maximum number of generations as 1000, and test them for 20 runs. Suppose the corresponding fitness is  $E_i(t), (i=1, \dots, 20; t=1, \dots, 1000)$ , then we list the mean fitness

$E(t) = \sum_{i=1}^{20} E_i(t)/20$  in Figure 1 to describe the convergence speed. The inertia weight  $w$  is

different in these algorithms. In the IFD-PSO, a linearly decreasing inertia weight is used which starts at 1, and ends at 0; in the FD-PSO,  $w$  is set as 1 all the time; in the DPSO,  $w$  is 0.9 all along. From Fig. 1 and Table 1, it can be found out that IFD-PSO has the best convergence speed and FD-PSO is in the next place. Consider the robustness and effectiveness from various criterions, IFD-PSO has the best performance, and DPSO is worst.

**Table 1.** Comparisons of IFD-PSO, FD-PSO and DPSO for brumal14

	IFD-PSO	FD-PSO	DPSO
Times of converging to the global optimum value/ Total running times	35%	5%	0%
Best fitness value	30.879	30.879	31.807
Mean fitness value	31.449	33.071	36.119
Worst fitness value	33.437	35.318	40.87
Iteration numbers of the best running	32	461	1000



**Fig. 1.** The convergence curves of IFD-PSO, FD-PSO and DPSO

## 5 Conclusions

In this paper, we propose a PSO method IFD-PSO, and apply it to TSP. We use fuzzy matrix space to represent the corresponding TSP solution, and bring forward the transformation method. Heuristic information is employed to improve the convergence speed. The experiment results show that IFD-PSO can converge to the optimal value more quickly than the current fuzzy discrete PSO, and has a better performance. This method can also be utilized in various applications which can be transformed into TSP.

## References

- Wei Pang, Kang-ping Wang, Chun-guang Zhou, Long-jiang Dong. Fuzzy discrete particle swarm optimization for solving traveling salesman problem. Proceedings of the Fourth International Conference on Computer and Information Technology. 2004: 796-800.
- Wei Pang, Kang-ping Wang, et al. Modified particle swarm optimization based on space transformation for solving traveling salesman problem. Proceedings of the Third International Conference on Machine Learning and Cybernetics, Shanghai, 2004: 2342-2346.
- Maurice Clerc. Discrete Particle Swarm Optimization. [http://clerc.maurice.free.fr/pso/pso\\_tsp/Discrete\\_PSO\\_TSP.htm](http://clerc.maurice.free.fr/pso/pso_tsp/Discrete_PSO_TSP.htm).
- Kang-Ping Wang, et al. Particle Swarm Optimization for Traveling Salesman Problem. The Second International Conference on Machine Learning and Cybernetics, 2003.

# A Network Event Correlation Algorithm Based on Fault Filtration\*

Qiuhua Zheng<sup>1,2</sup>, Yuntao Qian<sup>1,2</sup>, and Min Yao<sup>1</sup>

<sup>1</sup> Computational Intelligence Research Laboratory, College of Computer Science  
Zhejiang University, China

<sup>2</sup> State key Laboratory of Information Security,  
Institute of Software of Chinese Academy of Sciences, Beijing, China  
zheng\_qiuhsua@163.com, ytqian@zju.edu.cn

**Abstract.** This paper proposed a new event correlation technique to enhance the heuristic of the increment hypothesis updating (IHU) algorithm. This approach estimates the likelihood of each fault in the faults set and removes these faults with less likelihood. By this approach we also can determine whether an event is spurious or not. Simulation shows that this approach can get a high accuracy and fast speed of correlation even if the network has a high event loss and spuriousness.

## 1 Introduction

Event correlation, a central aspect of network fault diagnosis, is a process of analyzing alarms received to isolate possible root causes responsible for network's symptoms occurrences. Since failures are unavoidable in large and complex network, an effective event correlation can make network system more robust, and their operation more reliable, ultimately increasing the confidence level in the services they provide. Now network event correlation technique has been a focus of research activity[1-7]. In previous works, we proposed an correlation technique IHUCB that integrates the IHU algorithm with the codebook approach[8]. This approach utilizes the codebook technique to encode the network's fault-symptom model, and uses the IHU algorithm to create the fault hypotheses set, and then calculate these fault hypotheses' likelihood through the codebook approach. This algorithm can correlate multiple faults cases when its codebook only include codes of the single problem, and still has its efficiency and robust to event lost and spuriousness. However, the IHUCB algorithm does neither consider the difference among the likelihood of faults nor deal with spurious events specially. It treats spurious events and real events with the same method. Many fault hypotheses with low probability during the fault hypotheses updating phase are not removed until the final measurement phase. It increases the time of correlation events because of the over-large fault hypotheses set and results in a high false positive rate because of faults in fault hypothesis where some one are to explain received spurious events. In addition, since all of events are thought of real

---

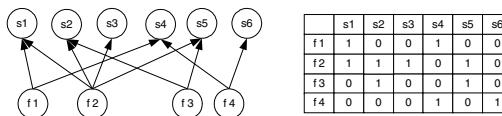
\* This research is supported partly by Science and Technology Project of Zhejiang (2006C21001).

events, sometimes the results of algorithm are not the faults with the smallest distance for real events. To avoid these disadvantages, we introduce a new heuristic approach which estimates the likelihood of faults and removes these faults with less likelihood.

The rest of this paper is organized as the following. In section 2, we describe a formal model of the event correlation. In section 3, we propose the new event correlation algorithm. In section 4, we describe the simulation result for evaluation the proposed technique. Finally, we conclude our work and present the direction for future research in this area in section 5.

## 2 The Event Correlation Model

A fault-symptom map can be represented by a bipartite directed graph that encodes direct causal relationships between faults and a set of observed symptoms when the fault occurs. This map can also be represented by a dependence matrix where  $d_{ij}=1$  means that if fault  $f_j$  occurs it will lead to symptom  $s_i$ , and  $d_{ij}=0$  means that fault  $f_j$  is independent with symptom  $s_i$ , i.e., fault  $f_j$  occurs without causing symptom  $s_i$ . An illustration example can be shown at fig.1.



**Fig. 1.** A fault-symptom map and the dependence matrix

When faults  $F$  occurred in the network system, they will lead to the corresponding symptoms coming forth, and the monitor system will receive a number of alarms. This procedure is called fault propagation. On the contrary, fault diagnosis is the reverse procedure of fault propagation. It aims to analyze the observed events  $E$  according to the dependence matrix  $D$  for finding the set of possible fault hypotheses  $FH$  so that the events  $E$  occurrence likelihood is maximal. In fault propagation procedure, the symptoms occurred in system can be gained by the equation  $S = D * F$ . General, event correlation can be divided into two phases: first, the phase for creating fault hypotheses, whose task is to create these fault hypotheses sets which can explain received events; and secondly the phase for measurement beliefs of these fault hypotheses, whose task is to measure these fault hypotheses sets and choose the best fault hypotheses as result.

## 3 The Event Correlation Algorithm Based on Fault Filtration

In the IHU technique[4], when an event  $e_i$  is received, this algorithm creates a set of fault hypothesis  $FHS_i$  by updating  $FHS_{i-1}$  with an explanation of the event  $e_i$ . The meaning that hypothesis  $h_k$  can explain event  $e_i \in E_O$  is hypothesis  $h_k$  includes at least one fault which can lead to the event  $e_i$  occurrence. At the worst case, there may be  $2^{|V|}$  fault hypotheses that can be responsible for fault events  $E_O$ , so fault hypotheses

set don't contain all subsets that can explain events  $E_O$ . A greedy algorithm will result in fast growth of fault hypothesis set's size then lead to the computational complexity of event correlation algorithm is unacceptable. Literature[4] proposed a heuristics approach which uses a function  $u(f_i)$  to determine whether fault  $f_i$  can be added into hypothesis  $h_k \in FHS_{i-1}$ . Fault  $f_i \in F_{ei}$  can be appended into  $h_k \in FHS_{i-1}$  only if the size of  $h_k$ ,  $|h_k|$ , is smaller than  $u(f_i)$ , where function  $u(f_i)$  is defined as the minimal size of a hypothesis in  $FHS_{i-1}$  that contains fault  $f_i$ . In IHUCB algorithm[8], we introduced a constraint to the maximal fault number to enhance the heuristic approach, and combined with the weighted hamming distance to measure the result. The result of IHUCB algorithm shows that the heuristic approach can reduce hypotheses to a great extent. However, just as mentioned in section 1, IHUCB algorithm does neither consider the differences among the likelihood of faults nor deal with spurious events specially. Therefore, we propose a novel approach, which incorporates the likelihood of fault to decrease the hypothesis and judge whether event is spurious or not.

### 3.1 The Heuristic Approach for Fault Filtration

To avoid the disadvantages of IHUCB, we point out a heuristic approach, which can judge whether the event is spurious and creates fault set which explain events for every received event. It is described in detail as follows: (1) First of all, this approach gets a fault set  $F_{ei}$  which is composed of all faults those can lead to event  $e_i$  occurred. (2) Then, for each fault in set  $F_{ei}$ , find the event vector corresponding to this fault from codebook, and compare this vector with the received event vector  $E_{rec}$  to calculate the number of loss events. When the number of loss events is more than a predicted threshold value  $T$  (In section 3.2, we will introduce how to compute the threshold value), we consider that event  $e_i$  is not caused by this fault and remove it from set  $F_{ei}$ . Repeat this step till all the faults in set  $F_{ei}$  have been processed.(3) If set  $F_{ei}$  is empty, we think event  $e_i$  is spurious and remove this event from  $E_{rec}$ . Otherwise, set  $F_{ei}$  is returned as the fault set which can explain event  $e_i$ .

### 3.2 How to Compute the Threshold Value

There are two kinds of methods to compute the threshold value.

- (1) Static threshold value method. It chooses a fixed threshold value all the time.
- (2) Dynamic threshold value method. It calculates the threshold value dynamically according to the number of events in codebook and the average loss ratio of events. In this paper, we design a method, which makes the probability of loss events number is more than threshold value  $T$  is less than a value  $\alpha$ , i.e.,  $P(Event loss num >= T) < \alpha$ . It can be computed by the following equation.

$$\sum_{i=1}^T C_{EN}^i * LR^i * (1-LR)^{EN-i} > 1 - \alpha \quad (1)$$

## 4 Simulation Study

In this section, we describe the simulation study performed to evaluate the technique presented in this paper. In our simulation, we use  $LR$  to represent the event loss ratio,

$SR$  to represent the event spurious ratio,  $CB$  to represent the codebook and  $FP$  to denote the distribution of fault occurrence. Since it is impossible to exhaust all codebook of a given size, we only test limited codebooks in each size.

Given parameters of  $LR_l$ ,  $LR_h$ ,  $SR_l$ ,  $SR_h$ , for given parameters of the faults number  $FN$  and the codebook' symptoms number  $SN$ , we design  $K$  simulation cases as follows: Randomly create the codebook  $CB_i(1 \leq i \leq K)$ ; Randomly generate prior fault probability distribution  $FP_i$ , which is a uniformly distributed and their sum is 1; Randomly generate the  $LR_j^k(1 \leq j \leq SN)$  for every event in the codebook which is accord with the uniform distribution  $[LR_l, LR_h]$ ; Randomly generate the  $SR_j^k(1 \leq j \leq SN)$  for every event accord with the uniform distribution  $[SR_l, SR_h]$ . For  $i$ -th simulation case( $1 \leq i \leq K$ ), we create  $M$  simulation scenarios as follows. Generate fault number distribution  $FNP$  according with  $FP$  and  $FN$ ; Randomly generate the set  $F_i^k(1 \leq k \leq M)$  of faults according to the fault distribution  $FP_i$  and the fault number distribution  $FNP$ . Generate the code  $C_{Fi}$  of  $F_i^k(1 \leq k \leq M)$  according to the codebook  $CB_i$ . Randomly generate the loss events with  $LR_i$  and the spurious events with  $SR_i$ , then generate the observed events  $E_i$  by adding the noisy events to  $C_{Fi}$ ; Correlate the observed events  $E_i^k(1 \leq k \leq M)$  and gain the correlation result with the algorithm IHUCB and algorithm IHUCBPF; Calculate the detection rate  $DR_i^k(1 \leq k \leq M)$  and the false positive rate  $FPR_i^k(1 \leq k \leq M)$  with the following equations.

$$DR_i^k = |F_{id}^k \cap F_{ic}^k| / |F_{id}^k|, FPR_i^k = |F_{id}^k \setminus F_{ic}^k| / |F_{id}^k| \quad (2)$$

For  $i$ -th simulation case we calculate the mean detection rate  $DR_i = 1/M \sum_{k=1}^M DR_i^k$  and mean false detection rate  $FPR_i = 1/M \sum_{k=1}^M FPR_i^k$ . Then, we calculate the expected values of detection rate and false detection rate denoted by

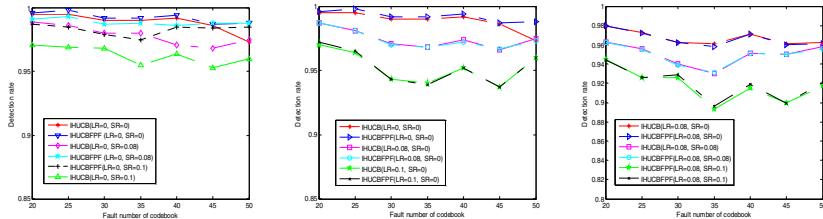


Fig. 2. The detection rate of two algorithms

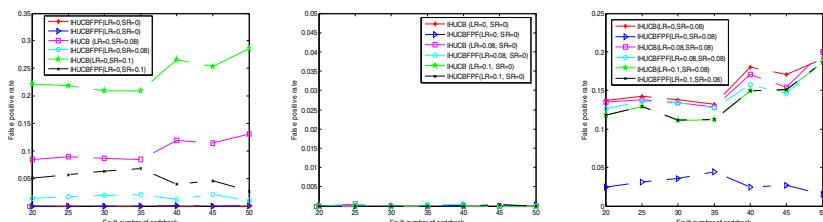
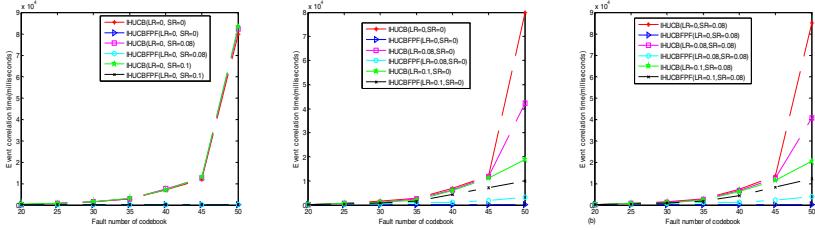


Fig. 3. The false positive rate of two algorithms



**Fig. 4.** The event correlation time of two algorithms

$DR_{FN, SN}$  and  $FPR_{FN, SN}$  respectively. In our simulation, we set  $fn_{max}=7$ ,  $\alpha=2.5\%$ , varied  $FN$  from 20 to 50,  $SN$  from 15 to 25. The parameter  $K$  is 100, and  $M$  is 5000.

Fig.2 and fig.3 present a comparison of the detection rate (DR) and false positive rate (FPR) for IHUCB and IHUCBFPF, respectively. As shown in fig.2, if  $LR$  is 0, when  $SR$  increases,  $DR$  of IHUCB decreases while that of IHUCBFPF is still at a high level. A conclusion could be reached that IHUCBFPF is more robust than IHUCB on the event loss. When  $LR$  is 0,  $FPR$  of IHUCB is higher than that of IHUCBFPF. The reason is that IHUCB deals with the spurious event as a real event. So in the creating fault hypotheses procedure the correct hypothesis is removed due to *premature hypothesis removal*[4], and in the measurement phase a wrong final choice creates. It also displays when  $SR$  is 0 even if change  $LR$  there is no significant differences between  $FPR$ s of two algorithms, that is because our proposed heuristic approach is only enhance the process to spurious events. In fig.3, the graph shows that IHUCBFPF can get a satisfying  $FPR$  on the condition with a high  $SR$  while that of IHUCB is unsatisfying. There is a phenomenon when the event loss ration is high; the  $DR$ s of IHUCB and IHUCBFPF are basically the same. However, IHUCBFPF has a high improvement on  $FPR$ .  $DR$ s are not improved because when  $LR$  is high, the event loss number has exceeded the diameter of the codebook. For this problem, we can increase the event number of codebook to improve the tolerance of event loss. Fig.4 shows that IHUCB is faster than IHUCBFPF, especially in less  $LR$  cases. This reason is that many fault hypotheses with low probability are removed during the hypotheses creating phase in IHUCBFPF while during the final measurement phase in IHUCB, and the time for creating hypotheses is the most important part.

## 5 Conclusion

The IHUCBFPF technique proposed in this paper through estimating the likelihood of each fault in the faults set which can explain the event to remove these faults with less likelihood to enhance the IHUCB algorithm. It uses a codebook model which represents the relationship between fault and symptom. As shown in the simulation, it has a satisfied improvement on the accuracy of detection, false positive rate and time of correlation when event spurious ratio is high. In the near future, we plan to apply this heuristic into the probability codebook.

## References

1. Bouloutas, A.T., S. Calo, and A.Finkel, *Alarm Correlation and Fault Identification in Communication Networks*. IEEE Transactions on Communications, 1994. **42**: p. 523 - 533.
2. Katzela, I. and M. Schwartz, *Schemes for Fault Identification in Communication Networks*. IEEE/ACM Transactions on Networking, 1995. **3**(6): p. 753-764.
3. Steinder, M. and A.S. Sethi. *Increasing Robustness of Fault Localization through Analysis of Lost, Spurious, and Positive Symptoms*. in *INFOCOM2002*. 2002. New York.
4. Steinder, M. and A.S. Sethi, *Probabilistic Fault Diagnosis in Communication Systems through Incremental Hypothesis Updating*. Computer Networks, 2004. **45**(4): p. 537-562.
5. Yemini, S.A., et al., *High Speed and Robust Event Correlation*. IEEE Communications Magazine, 1996. **34**(5): p. 82-90.
6. Liu, G., A.K. Mok, and E.J. Yang. *Composite events for network event correlation*. in *IM99*. 1999.
7. Klinger, S., et al. *A coding approach to event correlation*. in *Intelligent Network Management*. 1995. Santa Barbara, CA.
8. Zheng, Q.H. and Qian. Y.T. *An Event Correlation Approach Based on the Combination of IHU and Codebook*. in *CIS05*. 2005. Xian, China.

# CPR Localization Using the RFID Tag-Floor\*

Jung-Wook Choi, Dong-Ik Oh, and Seung-Woo Kim

Dept. of Computer Science & Engineering, College of Engineering,  
SoonChunHyang University, Asan, Korea  
{jwchoi, dohdoh, seungwo}@sch.ac.kr

**Abstract.** In this paper, we describe our approach to achieve accurate localization using RFID for Cellular Phone Robot (CPR). We solely rely on RFID mechanism and complement coordinate errors accumulated during the wheel-based CPR navigation. We especially focus on how to distribute RFID tags (tag pattern) and how many to place (tag granularity) on the floor to accomplish efficient navigations. We define the error in navigation and use it to compare the effectiveness of various RFID floor settings through a simulation. Identified tag patterns and granularities would be useful to many USN applications where the adoption of RFID technology is appropriate.

## 1 Introduction

Under the ubiquitous computing paradigm, we are connected to the world of computing anywhere and anytime through USN (Ubiquitous Sensor Network). In USN, object information pertaining to the computing is important. One of the most fundamental information needed is the positional (localization) information of the object. Use of laser and ultrasonic sensors is common for the purpose, but they cannot sense objects through obstacles (line-of-sight problem).

Recently, there have been a few attempts of applying RFID in localization [1, 2]. RFID does not suffer from the line-of-sight problem, and its object ID can be trivially retrieved. However, RFID has a problem with triangulation, the most common localization techniques adopted in USN. For this to work, the distance between a sensor and a sensed object needs to be accurately measured. But today's RFID technology does not provide enough accuracy in the distance measurement. In the RFID sensing literatures, the best result so far gives an average error of 0.77 feet in coordinate detection. It is not accurate enough for the robot navigation [1]. Errors in RF distance measurement account for such inaccuracies.

Therefore, in this paper, we suggest a new way of implementing robot localization using RFID. This effort is a part of CPR (Cellular Phone Robot) development [3, 4]. In this development, we place an RFID reader onto CPR and prepare a tag-floor. A tag-floor is a floor installed with RFID tags, each of which possesses coordinate information of itself. With such a setting, CPR may acquire accurate coordinates during navigation, because the localization error does not depend on the distance between the reader and the tag.

---

\* This research was supported in part by the Ministry of Science & Technology of Korea (Grant No. R01-2004-000-10274-0(2005)).

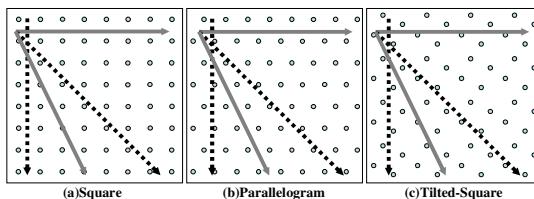
Several tag-floor arrangements have already been suggested [5, 6]. However, these studies do not address the issue of how (tag pattern) and how many (tag granularity) tags should be distributed on the floor. In this study, we suggest a way to determine efficient tag patterns and tag granularities for the tag-floor.

This paper is organized as follows. In section 2, we introduce the CPR navigation. We describe the tag-floor assignment and discuss how it may affect the overall localization performance. Section 3 covers the localization efficiency by presenting and analyzing the simulation results. Section 4 concludes.

## 2 CPR and RFID Tag-Floor Localization

CPR is a new technological concept combining CP (Cellular Phone) and RT (Robot Technology) [3, 4]. This paper focuses on the localization and navigation of CPR named  $\text{CPR}^{\text{Mobility}}$ , which consists of two main controllers: *Trajectory Controller* and *Self-Localization Controller*. The Trajectory Controller is responsible for the wheel-based navigation. The Self-Localization Controller provides coordinate information of the moving CPR. For the robotic movement, a pair of navigational wheels is installed. With the coordinate information acquired from the Self-Localization Controller, Trajectory Controller refines CPR movements to achieve better navigation. We make CPR know of its own coordinate through RFID sensing. We place a small RFID reader onto the CPR. On the in-door floor, we place multiple RFID tags and associate each tag (tag ID) with a coordinate. During an actual navigation, as the CPR detects a tag, its coordinate is fed to the Trajectory Controller.

There may be many sensible ways of placing RFID tags to the floor. Never the less, it is practical to prepare tiles of the same fashion and assemble them to produce a navigational floor. The simplest way of arranging RFID tags on a tile is to place 4 tags in a square (“square”) as in Fig. 1(a). Another way might be “parallelogram” suggested in Fig. 1(b). It is suggested that with the pattern in Fig. 1(a), there are higher possibilities of missing tag detections [5]. However, as we can see from Fig. 1(a) and 1(b), possibilities of missing tags in both cases are about the same. Therefore, we suggest the “tilted-square” pattern as in Fig. 1(c). This pattern seems to have the lowest possibility of missing tag detections during navigation.



**Fig. 1.** Possible misses of tag detection

For the tag-floor navigation, the more tags we place on the floor, the higher the tag encountering possibility will be. We can compensate the error accumulated by the wheel-based navigation more often in this case. However, it would be desirable to place only a minimum number of tags on the floor.

### 3 Efficient Tag-Floor Setting

In order to determine efficient tag granularities and tag arrangement patterns of the tag-floor, we developed a simulation program. We defined the error in navigation and used it to compare the effectiveness of the tag-floor.

#### 3.1 Performance Factors

Since we are considering CPR movement under no obstacles (near obstacles dead-reckoning navigation is performed; see [4] for details), the most important performance factor is the time it takes to travel from a given departure point to a given destination point. This is the sum of straight line movement times at each successive point it goes through, plus the times to adjust its posture at each tag. The movement depends on the arrangement of tags on the floor as well. Therefore, we define CPR navigation time as:

$$T_e(x_s, x_d) = \sum_{i=0}^{n-1} \{tm(x_i, x_{i+1}) + tr(x_i, x_n)\} \quad (1)$$

where,

$T_e(x_s, x_d)$ : Navigation time from  $x_s$  to  $x_d$  under the floor arrangement  $e$

$tm(x_a, x_b)$ : Straight-line movement time from  $x_a$  to  $x_b$

$tr(x_a, x_b)$ : Posture adjustment time from  $x_a$  for  $x_b$

$x_s, x_0$ : Departing points;  $x_d, x_n$ : Destination points

$x_1 \dots x_{n-1}$ : Locations of successive tags encountered during navigation

By subtracting the straight-line movement time between the departure and the destination points from Equation (1) and by normalizing it against the straight line movement time, we define relative navigational error  $RE_e(x_s, x_d)$  to compare the effectiveness of different tag arrangements and granularities.

$$RE_e(x_s, x_d) = [T_e(x_s, x_d) - tm(x_s, x_d)] / tm(x_s, x_d) \quad (2)$$

#### 3.2 Simulation Parameters and Fixed Values

The simulation program measures relative effectiveness of the navigation. In this program we use various parameters to better reflect real-world navigation. Major parameters a user can specify are:

**Tag patterns.** “square,” “parallelogram,” or “tilted-square” can be chosen.

**Tag granularities.** Four tags per tile. Hence, the tile size determines the granularity.

**Read ranges of the tag and the reader.** There are many types of tags and readers. Therefore, we make this parameter flexible.

**Others.** Specify velocity, posture adjustment time, and angular wheel error.

Using the definition of  $RE$  in Equation (2) and by varying the patterns and granularities of the tags, we analyzed CPR navigation performance through

simulation. We used 50,000 of randomly generated departure-destination pairs. We accumulated  $RE_t$  values for each of three floor patterns and used the mean  $RE_t$  values for the performance comparison. Among the simulation parameters, some are fixed to reflect real-world situations. Table 1 summarizes the fixed values used.

**Table 1.** Fixed simulation parameters

Parameters	Values
Floor Size	1000cm × 1000cm
CPR Velocity (Straight, Rotation)	10.0cm/sec, 2.0cm/sec
Minimum Path Distance	400 cm
Angular Error Rate	$\pm \frac{\pi}{36} rad$

### 3.3 Simulation Results and Analysis

In order to determine how many tags are sufficient to generate an efficient navigational performance, we find the correlation between navigational performance and tag granularities. We differentiate tag granularities by adjusting the *read range* and the *tag interval*. The read range is determined by the maximum distance between centers of a tag and a reader. The tag interval is the distance between the centers of one tag to the closest tag on the floor.

**Tag Granularity.** According to the simulation results, we notice that there exists an optimal tag interval where  $RE_t$  becomes minimal regardless of the tag arrangement patterns. By performing simulations on various read range and extracting those minimal  $RE_t$  values, we conclude that there exists an optimal relationship between the read range and the tag interval (tag granularity). Table 2 summaries the optimal results obtained from various read ranges. It shows that by maintaining the interval about 4 times of the range (about 4:1 ratio between the tag interval and the read range), we reach the minimal navigational errors in terms of  $RE_t$ .

**Tag Arrangement Pattern.** Under the optimal tag granularities, we compare the performance of three tag arrangement patterns by consulting RE column in Table 2. In all cases, “*tilted-square*” arrangement produces the best performances in relative effectiveness. With the results, we conclude that the “*tilted-square*” pattern suggested in this study produces the best efficiency for the tag-floor navigation.

**Table 2.** Optimal tag intervals and RE values (T: Tilted-Square, S: Square, P: Parallelogram)

Read range	Optimal tag interval	Relative error (RE) in percent		
6 cm	28 cm	(T) 36.94%	(S) 40.36%	(P) 39.82%
8 cm	33 cm	(T) 32.51%	(S) 35.97%	(P) 35.59%
10 cm	41 cm	(T) 29.75%	(S) 32.76%	(P) 33.08%
12 cm	45 cm	(T) 27.53%	(S) 30.58%	(P) 30.21%
14 cm	53 cm	(T) 26.01%	(S) 28.35%	(P) 29.32%

## 4 Conclusion

For the implementation of the self-localization feature of CPR, we used RFID technology. Unlike other conventional sensors, RFID readers do not suffer from the line-of-sight problem, hence a better implementation of CPR navigation is possible.

In this paper we suggested an RFID tag installed floor for effective CPR navigation. Furthermore, we developed a simulation program to find out better ways of distributing RFID tags on the floor. We defined the error in navigation and used it to compare the effectiveness of various floor settings. The simulation results indicate that 4:1 ratio between the tag interval and the tag/reader range is desirable. They also demonstrate that the "*tilted-square*" is the most adequate tag arrangement pattern. This analytical result should be beneficial for many other USN applications, in which the application of an RFID tag-floor is appropriate. In the continuing research, live test navigation based on the findings of this work will be conducted.

## References

1. G. Kantor and S. Singh, "Preliminary Results in Range-Only Localization and Mapping," Proceedings of the IEEE Conference on Robotics and Automation, Washington, DC, May 2002
2. D. Hahnel, W. Burgard, D. Fox, K. Fishkin, and M. Philipose, "Mapping and Localization with RFID Technology," Intel Research Institute, Seattle, WA, Tech. Rep. IRS-TR-03-014, December 2003
3. Jaeil Choe, Seungwoo Kim, "A Study on Infra-Technology of Robotic Cellular Phone," Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 3571-3576, Sept., 2004.
4. Seungwoo Kim, Jaeil Choe, "A Study on The new Technological Concept of Robotic Cellular Phone (RCP)," International Journal of Intelligent Material Systems and Structures, Vol. 16, no. 12, pp. 995-1005, Dec. 2005.
5. Itiro Siio, "User Position Detection using RFID Tags," Technical Report Proceedings of Japan Information Processing Society, 00-HI-88, pp. 45-50, 2000
6. Svetlana Domnitcheva, "Smart Vacuum Cleaner: An Autonomous Location-Aware Cleaning Device," Proceedings of the 6th International Conference on Ubiquitous Computing, Tokyo, Japan, Sep. 2004.

# Development of a Biologically-Inspired Mesoscale Robot

Abdul A. Yumaryanto<sup>1</sup>, Jaebum An<sup>2</sup>, and Sangyoon Lee<sup>2</sup>

<sup>1</sup> Department of Advanced Technology Fusion, Konkuk University

1 Hwayang-dong, Seoul, South Korea

azis@konkuk.ac.kr

<sup>2</sup> School of Mechanical and Aerospace Engineering, Konkuk University

1 Hwayang-dong, Seoul, South Korea

{pojec, slee}@konkuk.ac.kr

**Abstract.** This paper presents the design and prototype of a mesoscale (13 cm long) six-legged walking robot whose locomotion is actuated by a piezoelectric actuator named LIPCA, which consists of multiple layers of glass/epoxy and carbon/epoxy that encapsulate a unimorph piezoceramic actuator. Inspired by the walking kinematics of cockroaches, our robot uses the alternating tripod gait (the front and rear legs on the same side move together with the middle leg on the other side for the locomotion), and has six legs that are designed to mimic the function of those of cockroaches. All the experiments with the prototype show a possibility of a small, light, and agile walking robot that is actuated by LIPCA without using any conventional electromagnetic actuator.

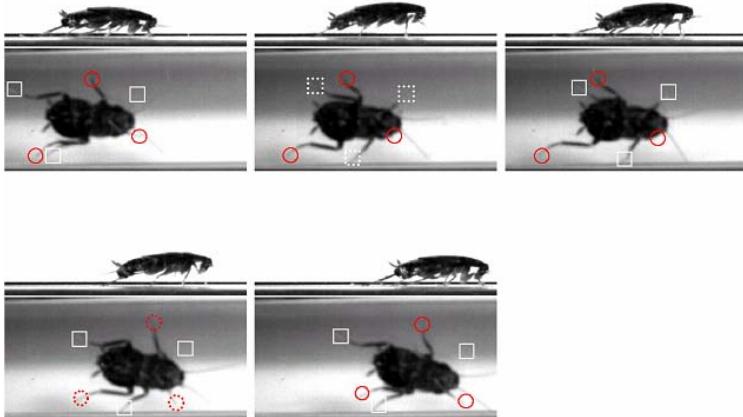
## 1 Introduction

Piezoelectric materials are smart in a sense that they can sense changes in the environment and respond by changing their material properties and geometry. Such materials have been used in the robotics field as sensors [1] and actuators [2, 3]. Recent developments of piezoelectric devices are unimorph-type actuators, in which a piezoelectric ceramic is bonded to a thin metal sheet. The secondary material amplifies the axial displacement of the actuator by constricting the lateral motion. Examples of such type can be THUNDER and RAINBOW [4].

A piezo-composite actuator called LIPCA (Lightweight Piezoceramic Composite curved Actuator) has a different structure: lightweight fiber-reinforced plastic layers instead of heavy metal layers. Experimental results show that LIPCA can produce 60% larger displacement and is 40% lighter than THUNDER [5]. In addition to previous applications of LIPCA as an actuator, LIPCA is used here for actuating a small (13 cm long), six-legged walking robot. Compared to a conventional-scale robot, a mesoscale robot has a limited design space, and hence it needs a simple design approach and a lighter and smaller actuator like LIPCA. Noticing that biological insects are intelligent in terms of the agile and stable motion, we reflect the walking kinematics of cockroaches on the design of our hexapod robot. We report the experimental results with real cockroaches and the design and prototype of a LIPCA-actuated mobile robot.

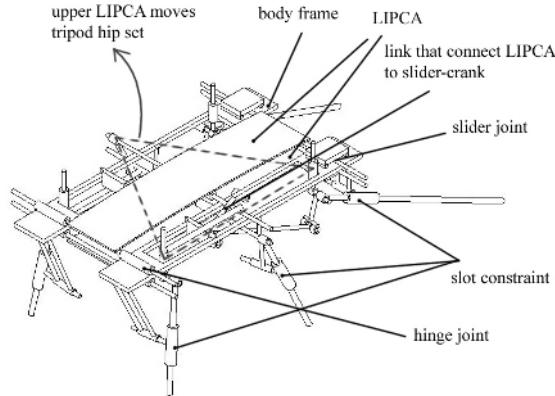
## 2 Design and Prototype of the Robot

The walking mechanism of our robot is inspired by an agile six-legged insect, cockroach. We analyzed the walking kinematics of German cockroaches (*Blattala Germanica*) with an experimental apparatus composed of a high-speed camera, a computer, and a homemade walking track. It is observed that when a cockroach walks, it uses the alternating tripod gait where the front and rear legs on one side and the middle leg on the opposite side move concurrently (Fig. 1). In addition, the front, middle, and rear legs do different functions while walking. The rear leg is used to thrust or accelerate the body while the front leg is used to decelerate the body. The middle leg has two functions: one is to decelerate and the other is to accelerate the body while stabilizing the body in the lateral direction. The tripod gait and the functions of each leg of cockroaches are reflected in the design of our robot.



**Fig. 1.** Walking of a cockroach

Our six-legged robot is actuated by two LIPCA strips, which are placed in the body frame using simply-supported joints (see Fig. 2). By the application of a high AC voltage, the LIPCA strips move up and down alternately except the edges, which are constrained to the body frame. Each LIPCA is used to move one set of legs for the alternating tripod gait where one set of legs stroke on the ground while the other set swings above the ground. We constructed two transfer mechanisms to convert a LIPCA displacement into a stroke in order to realize a walking mechanism. The first mechanism is for connecting the middle part of LIPCA to the hip part of the leg and the second is for joining the hip part and the foot. On the rear legs, the displacement of LIPCA is transferred using a slider-crank mechanism, which amplifies the displacement. The front and middle legs are directly connected without any amplification. Theoretically the hip displacement of the front leg is equal to that of the middle one, while the hip displacement of the rear leg is larger than that of LIPCA.



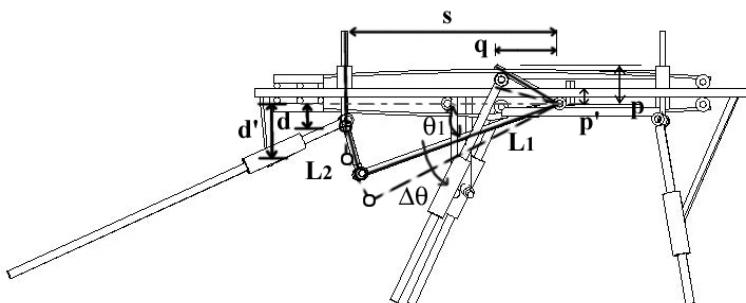
**Fig. 2.** Design of the robot

The amplification can be calculated as follows:

$$Amp = \frac{d' - d}{p - p'} \quad (1)$$

where

$d' = L_1 \sin(\theta_1 + \Delta\theta) - \sqrt{L_2^2 - (s - L_1 \cos(\theta_1 + \Delta\theta))^2}$  ,  $d = L_1 \sin \theta_1 - \sqrt{L_2^2 - (s - L_1 \cos \theta_1)^2}$  , and  $\Delta\theta = \tan^{-1}\left(\frac{p}{q}\right) - \tan^{-1}\left(\frac{p'}{q}\right)$ . All the parameters in the right-hand side are known from the geometry of the robot (see Fig. 3).



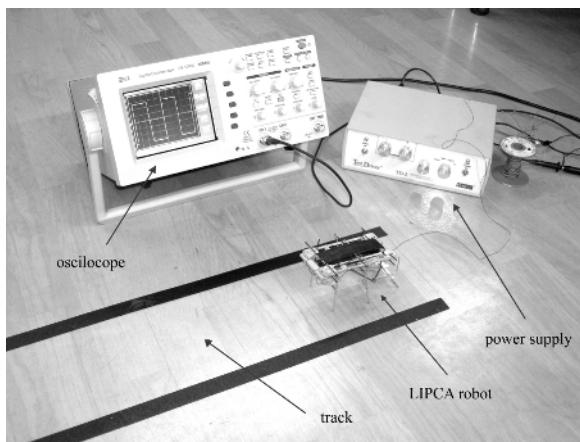
**Fig. 3.** Slider-crank mechanism of the leg

The horizontal displacement of the foot creates a thrust for accelerating the body forward. When positioning the slot in the rear leg we used an optimization method to maximize the displacement in the horizontal direction. The result is that the slot should be placed 15 mm horizontally behind the hip and 23.3 mm vertically below the hip, which will produce 8.7 mm in the backward stroke. The body frame and the linkages of the prototype robot are made of balsa wood and carbon composite rods,

respectively. The legs are made using a metal rod to get a high impulse. The total weight is 35 grams and the dimension is 120 mm × 55 mm × 65 mm.

### 3 Experiments and Results

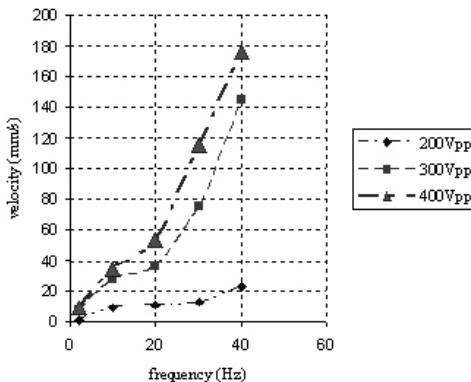
To measure the walking performance of our robot we conducted several experiments with an experimental apparatus (Fig. 4). Both LIPCA strips were driven with a square function of three different voltages ( $\pm 100V$ ,  $\pm 150V$ , and  $\pm 200V$  (200, 300, and 400 V peak-to-peak respectively)), with the phase of  $180^\circ$  at 2, 10, 20, 30, and 40 Hz for each voltage input. For each frequency we measured the speed three times and took the average value, which is shown in Fig. 5. As the operating frequency increases, the alternating tripod gait frequency becomes larger, and thus the robot moves faster. The voltage of the power supply changes the displacement of LIPCA, and thus the rear leg stroke becomes larger onto the ground to increase the speed. At frequencies over 30 Hz, change from 200 Vpp to 300 Vpp brought a larger effect on the velocity than the change from 300 Vpp to 400 Vpp.



**Fig. 4.** Experimental apparatus for walking of the robot

### 4 Discussion and Conclusions

We have reported the design, prototype, and experiments of a mesoscale, light, and agile walking robot that is actuated by a smart material, LIPCA, and inspired by an agilely walking insect, a cockroach. Compared with MG3 [4], our robot is slower at the same frequency and voltage. However, unlike MG3 that has a frequency-based turning capability, our robot can walk straight at any frequency, which implies that our robot has a potential to walk faster with the application of a higher frequency. The second prototype of our LIPCA-actuated robot and a light and small power



**Fig. 5.** Performance of the robot for various applied voltages and frequencies

supply converter are under development, with the goal that the robot becomes a self-powered LIPCA-actuated robot. In the second prototype, we place two LIPCA strips in the middle instead of fixing them in the edges.

*Acknowledgements.* This work was supported by Korea Research Foundation Grant (Intensive Research Center Program 2004, KRF-2004-005-D00047) and the support is sincerely appreciated.

## References

1. Klahold, J., Rautenberg, J., Ruckert, U.: Continuous sonar sensing for mobile mini-robots. Proceeding of the IEEE International Conference on Robotics and Automation, Vol. 1 (2004) 323-328
2. Hollinger, G.A., Briscoe, J.M.: Genetic Optimization and Simulation of a Piezoelectric Pipe-Crawling Inspection Robot. Proceeding of the 2005 IEEE International Conference on Robotics and Automation, Vol. 1 (2005) 484-489
3. Goldfarb, M., Golgola, M., Fischer, G., Garcia, E.: Development of a piezoelectrically-actuated mesoscale robot quadruped. Journal of Micromechatronics, Vol.1, No. 3 (2001) 205-219
4. Wise, S.A.: Displacement properties of rainbow and thunder piezoelectric actuators, Sensors and Actuators, Vol. 69 (1998) 33-38
5. Yoon, K.J., Park, K.H., Lee, S.K., Goo, N.S., Park, H.C.: Analytical design model for a piezo-composite unimorph actuator and its verification using lightweight piezo-composite curved actuators, Smart Material and Structures, Vol.13 (2004) 459-467

# Timed Petri-Net(TPN) Based Scheduling Holon and Its Solution with a Hybrid PSO-GA Based Evolutionary Algorithm(HPGA)

Fuqing Zhao<sup>1</sup>, Yahong Yang<sup>2</sup>, Qiuyu Zhang<sup>1</sup>, and Huawei Yi<sup>1</sup>

<sup>1</sup> School of Computer and Communication, Lanzhou University of Technology,  
730050 Lanzhou, P.R. China

{zhaofq, zhangqy, yihw}@mail2.lut.cn

<sup>2</sup> College of Civil Engineering, Lanzhou University of Techchnology,  
730050 Lanzhou, P.R. China  
yangyahong@mail2.lut.cn

**Abstract.** Modern manufacturing systems have to cope with dynamic changes and uncertainties such as machine break down, hot orders and other kinds of disturbances. Holonic manufacturing systems (HMS) provide a flexible and decentralized manufacturing environment to accommodate changes dynamically. In this paper, A new class of Time Petri Nets(TPN), Buffer-nets, for defining a Scheduling Holon is proposed, which enhances the modeling techniques for manufacturing systems with features that are considered difficult to model. The proposed novel GA algorithm performs the population alternation according to the features of the evolution of the populations in natural. Simulation results show that the proposed GA is more efficient than standard GAs. The proposed HPGA synthesizes the merits in both PSO and GA. The simulation results of the example show that the methods to scheduling holon are effective for fulfilling the scheduling problem.

## 1 Introduction

Holonic manufacturing is a highly distributed control paradigm based on a kind of autonomous and cooperative entity called “holon”[1]. HMS requires a robust coordination and collaboration mechanism to allocate available resources to achieve the production goal. Multi-agent systems (MAS) [2] provides desirable characteristics to proactively handle uncertainties. HMS is usually modeled as a cooperative MAS. Although there are a lot of research works on HMS[3][4], however, deadlock issue has not been addressed[5].

The remainder of this paper is organized as follows. Section 2 proposes a new class of Time Petri Nets(TPN), Buffer-nets, for defining a Scheduling Holon. In section 3, a hybrid PSO-GA(HPGA) based evolutionary algorithm is proposed. A scheduling holon architecture, which integrates TPN models and HPGA techniques is given in Section 4. Section 5 concludes this paper.

## 2 Formulating the Scheduling Operation of a Scheduling Holon Using Petri-Nets

### 2.1 Buffer Nets

**Definition 1.** A timed-PN is called a Buffer net (B-net) if

$$(P = R \cup Q) \wedge (R \cap Q = \Phi)$$

where the set of places  $R$  represents the resources and the set of places  $Q$  represents the buffers, and the following three conditions are also satisfied:

- (a)  $I(r, t) = O(r, t) \quad \forall t \in T, \forall r \in R$
- (b)  $\forall t \in T$ , there exists a single  $p \in Q : I(p, t) = 1$  and a single  $p' \in Q : O(p', t) = 1$ , for  $p \neq p'$ ;
- (c) The subnet  $G' = (Q, T, I', O', M', \tau)$ ; where  $I'$  and  $O'$  are the restrictions of  $I$  to  $(Q \times T)$  and  $O$  to  $(T \times Q)$ .

**Definition 2.** Initial and final states of a B-net:

- 1) For a B-net,  $B_I \subset Q$  is called a set of input buffer places if  $\forall p \in B_I$  and  $t \in T$ ,  $I(p, t) = 1$  and  $O(p, t) = 0$ , i.e.  $B_I = \{p \in Q \mid (\exists t \in T) O(p, t) > 0\}$ .
- 2)  $B_O \subset Q$  is called a set of output buffer places if  $\forall p \in B_O$  and  $t \in T$ ,  $I(p, t) = 0$  and  $O(p, t) = 1$ , i.e.  $B_O = \{p \in Q \mid (\exists t \in T) I(p, t) > 0\}$ .

### 2.2 B-Nets to Model Scheduling Holon of HMS

In a Scheduling Holon, there are  $n$  (where  $n > 1$ ) products to be produced using  $m$  (where  $m > 1$ ) processing units. For each product, the sequence by which the processing units will be visited is pre-specified and is referred to as the product (or job) routing or processing recipes. Normally, the processing time  $\tau_{ij}$  for a product (or job)  $i$  ( $i = 1, 2, \dots, n$ ) in unit  $j$  ( $j = 1, 2, \dots, m$ ) is given.

Operation  $O_{ij}$  can be represented by two transitions  $t_{sij}$  and  $t_{fij}$  for the start and the termination of this operation, respectively, and one place  $p_{ij}$  with time duration  $\tau_{ij}$  for the processing activity.

## 3 Hybrid PSO-GA Based Evolutionary Algorithm(HPGA)

### 3.1 A Novel GA

By introducing the “dying probability” for the individuals and the “war/disease process” for the population, the authors propose a novel approach in this paper to determine the population size and the alternation between generations.

The step of the algorithm is summarized as follows:

- (1) Generate initial population:  $sizeof\_population=POP\_INITIAL$ ; create randomly  $population[sizeof\_population]$ ,  $die\_probability[sizeof\_population]=DIE\_PROBABILITY[0]$ .
- (2) Evaluate the individual to obtain the max fitness and the least fitness of the population:  $value\_max$  and  $value\_min$ .
- (3) Memorize the best solution and stop if  $value\_max > VALMAX$  or  $gap > GAP$ .
- (4) Select  $SELECTION\_POP$  individuals into the reproducing pool randomly according to their fitness.
- (5) Divide individuals in the reproduction pool into couples randomly. All the couples perform the crossover and mutation operations.
- (6) Perform die process. For each individual, according to its die probability determine whether it will die. If the individual should die then  $sizeof\_population--$ ; else if  $die\_probability = DIE\_PROBABILITY[k]$  then  $die\_probability = DIE\_PROBABILITY[k+1]$ .
- (7) Perform the war/disease process: if  $sizeof\_population > POP\_MAX$  then select  $POP\_INITIAL$  individuals randomly into the new population according to their fitness. Go to step 2.

### 3.2 PSO-GA Based Hybrid Algorithm

Particle Swarm Optimization (PSO) also is an evolutionary computational model which is based on swarm intelligence. PSO is developed by Kennedy and Elberhart [6] who have been inspired by the research of the artificial livings, it finds the optimum solution by swarms following the best particle. Based on its advantages, the PSO is not only suitable for science research, but also engineering applications, in the fields of evolutionary computing, optimization and many others[7].

This paper proposes a novel hybrid PSO-GA based algorithm(HPGA). The performance of the algorithm is described as follows:

- (1) Initialize GA and PSO sub-systems, respectively.
- (2) Execute GA and PSO simultaneously.
- (3) Memorize the best solution as the final solution and stop if the best individual in one of the two sub-systems satisfies the termination criterion.
- (4) Perform hybrid process if generations could be divided exactly by the designated iterative times  $N$ . Select  $P$  individuals from both sub-systems randomly according to their fitness and exchange. Go to step 2.

## 4 Numerical Results

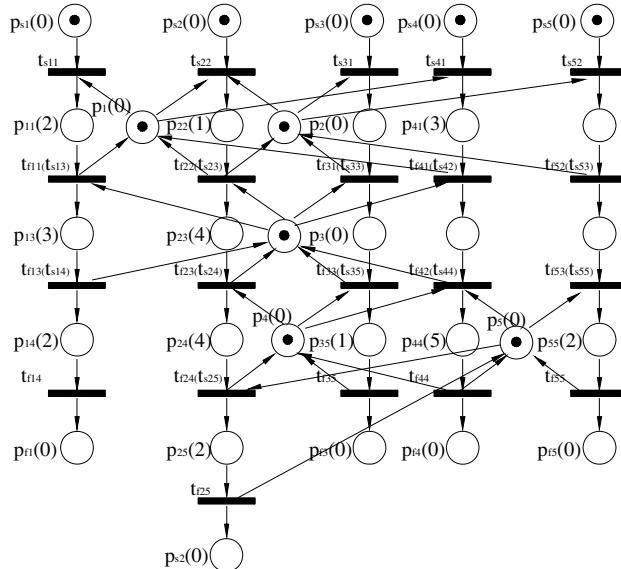
A multipurpose batch plant in a scheduling holon with five products (p1-p5) and five processing units (u1-u5) is considered as the case study to testify our model and algorithm, whose product recipe and processing time are given in Table 1. The TPN model for the NIS policy is shown in Fig. 1.

Both GA,PSO and HPGA algorithms have been implemented in this example. The number of searches can be decreased from 307 to 152 by the use of the HPGA

algorithm and again the results were the same as the results from GA and PSO. The calculation of usage duration for each unit in the multipurpose case is enhanced about 56% and 39% than that of GA and PSO respectively. Comparisons of different algorithm for the case study is shown in Table 2. From the table, we can see that the performance of HPGA is better than that of SGA and PSO.

**Table 1.** Processing times (h) of products

Units	Products				
	P1	P2	P3	P4	P5
U1	2.0	0.0	2.0	3.0	0.0
U2	0.0	1.0	0.0	2.0	3.0
U3	3.0	4.0	5.0	0.0	1.0
U4	2.0	4.0	0.0	5.0	0.0
U5	0.0	2.0	1.0	0.0	2.0



**Fig. 1.** TPN model for a Scheduling Holon(5×5)

**Table 2.** Comparisons of Different Algorithm for the case study

	Average Success(%)	Time	Object Function Solution
PSO	91.1%	95"	767.6933
SGA	78.8%	206"	976.966
HPGA	98.2%	58"	635.2341

## 5 Concluding Remarks

In the manufacture-to-order environment, production plans can only be drawn up and executed successfully with the use of a planning and control concept that provides predictability and stability. So the model and control of HMS based on Time Petri net theory provides a functional structure for a computer application, which enables the planners to cope with logistic and technological logical Scheduling problems on multiple levels of aggregation. The proposed HPGA synthesizes the merits in both PSO and GA. It is a simple and yet effective model to handle different kinds of continuous optimization problems.

## Acknowledgements

This research is supported by Natural Science foundation of GANSU province(grant NO 3ZS041-A25-020 and ZS032-B25-013).

## References

1. Mondal, S., Tiwari, M.K.:Application of an autonomous agent network to support the architecture of a holonic manufacturing system. International Journal of Advanced Manufacturing Technology, 12(2002)931-942
2. Leitao, Paulo; Restivo, Francisco: Experimental validation of ADACOR Holonic control system. Lecture Notes in Artificial Intelligence, 3593(2005)121-132
3. Luder, A., Klostermeyer, A., Peschke, J. et.al: Distributed Automation: PABADIS versus HMS. IEEE Transactions on Industrial Informatics, 1(2005)31-38
4. Giret, A., Botti, V., Valero, S.: MAS methodology for HMS. Lecture Notes in Artificial Intelligence, 3593(2005)39-49
5. Hosack, B., Mahmoodi, F., Mosier, C.T.: A comparison of deadlock avoidance policies in flexible manufacturing systems. International Journal of Production Research, 13(2003)2991-3006.
6. J. Kennedy, R.C. Eberhart: Particle swarm optimization. Proc. IEEE Internat. Conf. on Neural Networks, Perth, Australia,vol. IV, IEEE Service Center, Piscataway, NJ (1995)1942–1948
7. Da, Yida,Xiurun, Ge.: An improved PSO-based ANN with simulated annealing technique. Neurocomputing, 63(2005)527-533

# Recognition Rate Prediction for Dysarthric Speech Disorder Via Speech Consistency Score

Prakasith Kayasith<sup>1,2</sup>, Thanaruk Theeramunkong<sup>2</sup>, and Nuttakorn Thubthong<sup>3</sup>

<sup>1</sup> Assistive Technology Center, National Electronics and Computer Technology Center (NECTEC), Thailand Science Park, Klong Luang, Pathumthani 12120, Thailand  
Prakasith.kayasith@nectec.or.th

<sup>2</sup> School of Information and Computer Technology, Sirindhorn International Institute of Technology (SIIT), Thammasat University, Klong Luang, Pathumthani 12121, Thailand  
thanaruk@siit.tu.ac.th

<sup>3</sup> Acoustics and Speech Research Laboratory (ASRL), Department of Physics, Faculty of Science, Chulalongkorn University, Bangkok 10330, Thailand  
Nuttakorn.T@Chula.ac.th

**Abstract.** Dysarthria is a collection of motor speech disorder. A severity of dysarthria is traditionally evaluated by human expertise or a group of listener. This paper proposes a new indicator called *speech consistency score (SCS)*. By considering the relation of speech similarity-dissimilarity, SCS can be applied to evaluate the severity of dysarthric speaker. Aside from being used as a tool for speech assessment, SCS can be used to predict the possible outcome of speech recognition as well. A number of experiments are made to compare predicted recognition rates, generated by SCS, with the recognition rates of two well-known recognition systems, HMM and ANN. The result shows that the *root mean square error* between the prediction rates and recognition rates are less than 7.0% ( $R^2 = 0.74$ ) and 2.5% ( $R^2 = 0.96$ ) for HMM and ANN, respectively. Moreover, to utilized the use of SCS in general case, the test on unknown recognition set showed the error of 11 % ( $R^2 = 0.48$ ) for HMM.

## 1 Introduction

Dysarthria is a term given to a group of speech disorder in which the transmission of messages controlling the motor movements for speech is interrupted. Severe dysarthric speech may be completely *unintelligible* to unfamiliar listeners. However, it has been shown in previous studies [1 - 3] that, with carefully designed, ones with dysarthria can truly be benefited from incorporated speech recognition system into assistive devices.

In the area of speech assessment for people with speech disorder, there are two common tests based on perceptual analysis, namely articulatory test and intelligibility test. Basically both articulatory and intelligibility tests are subjective to human perception [4 - 5]. Besides, the results cannot be applied explicitly to evaluate the level of severity in term of speech processing features such as a consistency in speech. One may apply a straight-forward method by running the test with full processes speech recognition system. However, the choice is quite complicated and time consuming since we need both transcribing and training processes. To this end, this

paper proposes an indicator called *speech consistency score (SCS)*, which can be directly applied to predict a possibly outcome performance of those alternative speech technologies without those pre-processing processes.

## 2 Speech Consistency Score (SCS)

In this work, a *speech consistency score (SCS)* is defined as a ratio of similarity to dissimilarity of speech so that the comparison of consistency among speakers can be calculated. A *similarity score (SIM)* represents a level of speech signal similarity of the same words produced by a speaker while a *dissimilarity score (DIS)* represents dissimilarity of speech signals between different words produced by the same speaker. The relation of SIM and DIS can be used to evaluate the level of signal distortion represented by a group of speech features, which essentially used in modern speech-communication technologies.

To cope with a problem of time variation of speech samples, a technique of dynamic time wrapping (DTW), with adaptive slope constraint and accumulated penalty score, is applied to measure SIM and DIS. A signal will be separated into a sequence of smaller frames (25 ms width with 10 ms interval). Each frame will be represented by speech features. The definition of features can be changed depending on project interest. For example, in this study, we focused on a consistency score of speech recognition features. Therefore, MFCC features and their derivatives were chosen.

To evaluate a consistency of speech, three parameters, i.e. SIM, DIS and SCS, were taken into account. The similarity (SIM) value represents the similarity within the same word which was calculated from  $m$  samples (each speaker would be asked to speak the same word  $m$  times). In this case, an average distance within the same word  $w$  ( $\bar{X}^w$ ) can be calculated from equation (1), where  $w \in \{1, 2, \dots, n\}$ . The distance between sample  $i^{th}$  and  $j^{th}$  of word  $w$  ( $|X_i^w - X_j^w|$ ) is calculated by a DTW technique (with Euclidean's distance for frame comparison). Therefore, the value of similarity can be represented by the average distances for all words of each speaker,  $SIM_{sp}$ , as showed in equation (2) where  $n$  is the number of words with in the test set.

$$\bar{X}^w = \frac{1}{m} C_2 \sum_{i=1}^m \sum_{j=i+1}^m |X_i^w - X_j^w| \quad (1)$$

$$SIM_{sp} = \frac{1}{n} \sum_{i=1}^n \bar{X}^i \quad (2)$$

As the second parameter, the dissimilarity (DIS) value represents the average distances between all difference words within the test set. The method starts from choosing a template for each word, and then calculates the average distance for every pair of words. The template for each word,  $T^w$ , is chosen by selecting a sample from  $m$  samples within the same word  $w$  (out of five samples in this case). The selection criterion is to choose the utterance with the minimum sum of distances away from the others, as shown in equation (3). Then the *DIS* value for each speaker ( $DIS_{sp}$ ) is

calculated from the average distance of all words, as showed in equation (4), where  $n$  is the number of words in the test set.

$$T^w = \arg \min_{X_i^w} \left( \sum_{j=1}^m |X_i^w - X_j^w| \right), \text{ where } X_i^w \in \{X_1^w, X_2^w, \dots, X_m^w\} \quad (3)$$

$$DIS_{sp} = \frac{1}{n} \sum_{w=1}^n T_w \quad (4)$$

The last parameter defined as a *speech consistency score* (SCS) is formulated by the ratio of SIM to DIS. Generally, an ability to produce a variety of speech is different for each speaker. While SIM indicates similarity within a same word, dissimilarity displays dissimilarity among different words pronounced by the same speaker. These values (if being consider separately) fail to fulfill a meaningful comparison among different speakers, the ratio of SIM to DIS (a speech consistency score, SCS) is evaluated instead as shown in equation 5.

$$SCS_{sp} = \frac{DIS_{sp}}{SIM_{sp}} \quad (5)$$

### 3 Experiment

#### 3.1 Subjects

To evaluate our method, the speech corpus of sixteen speakers had been constructed from eight CP-Dysarthric children, (7 – 14 years old), and eight normal speakers including four adults (23 – 36 years old) and four children (7 – 12 years old). The corpus was created with the balance set of males and females. All CP-Dysarthric children with varied severity of dysarthria were recruited from Srisungwan compulsive school, a school for children with disabilities.

#### 3.2 Speech Corpus and Evaluation Methods

In order to evaluate the proposed method, two set of speech corpus were used. The first corpus, a *control set*, was designed especially for Thai phonemes error analysis. The second one, an *unknown set* was designed as a set of words that are frequently used for Assistive Technology. Recording was carried out under normal environmental conditions, in a quiet room with the door closed but no additional sound proof materials. Subjects were instructed to speak each word in isolation.

The results of SCS were compared with the evaluation results obtained from articulatory and intelligibility tests, as well as the accuracy rates of two speech recognition (SRR) models, HMM and ANN. Incorporated with SCS, recognition results of the control set from both models are used to generate the prediction function. Then the predicted rate of each speaker was calculated to compare with HMM recognition rate of the unknown set. All results were evaluated using the average of *root mean square error* ( $E_{rms}$ ) and the *correlation coefficient* ( $R^2$ ).

## 4 Results and Discussion

Table 4.1 shows the results of SCS of each speaker. The table shows the details of all normal speakers where the mean value of SCS is 1.5 ( $\sigma = 0.11$ ) while the mean value of dysarthric speakers is 1 ( $\sigma = 0.10$ ). The ratio of DIS and SIM represents relative difference of speech signal distribution of different words pronounced by a speaker. A higher SCS shows higher distribution (less overlap area) among different words for that speaker. Therefore, we can expect a high correlation between SCS and recognition rates. As shown from the experiment, the results are agreed to the expectation. In some cases such as DF01, DF04, and DM02, the ratios are even less than 1. This indicates a highly overlap of speech signals of among different words. Therefore the results from recognition systems are low for these speakers.

When all pairs of SCS and SRR results in Table 4.1 was used to generate the prediction function, the correlation coefficient ( $R^2$ ) is about 0.74 for HMM and 0.96 for ANN. That is a correlation between SCS and ANN is higher than a correlation between SCS and HMM. The result tell us that ANN based model is closer than HMM, to the DTW based method of finding SCS in terms of time alignment and a direct pattern comparison. The HMM is a time-dynamic recognition system that relies on probabilistic learning and language model that may not reflect the property of SCS. Next, the predicted recognition rates ( $P_{SCS}$ ) based on each SCS were calculated. The evaluation results are shown in Table 4.2.

**Table 4.1.** Experiment results for normal speakers (left table) and dysarthric speakers (right table): compare to speech recognition rate (SRR) of the control set

Code	SRR <sub>HMM</sub>	SRR <sub>ANN</sub>	SCS	Code	SRR <sub>HMM</sub>	SRR <sub>ANN</sub>	SCS
AF01	0.99	0.93	1.52	DF01	0.38	0.38	0.86
AF02	0.99	0.97	1.57	DF02	0.49	0.54	1.00
AM01	0.98	0.95	1.67	DF03	0.77	0.65	1.04
AM02	0.98	0.97	1.44	DF04	0.51	0.47	0.94
NF01	0.98	0.95	1.56	DM01	0.55	0.53	1.01
NF02	0.92	0.84	1.48	DM02	0.49	0.37	0.89
NM01	0.95	0.84	1.28	DM03	0.72	0.60	1.05
NM02	0.94	0.91	1.56	DM04	0.75	0.77	1.16

At the end of both tables are *root mean square errors* ( $E_{rms}$ ) between recognition method and each speech evaluation methods (SCS, speech articulatory test, and speech intelligibility test, as show as  $E_{SCS}$ ,  $E_{Arti}$ , and  $E_{Intel}$ , respectively).

**Table 4.2.** Predicted recognition rate results' error and correlation evaluation on control set (HMM and ANN) and unknown set (HMM)

Corpus	System	$E_{Arti}$	$E_{Intel}$	$E_{SCS}$	$R^2$
<b>Control Set</b>	HMM	0.1274	0.1023	0.0688	0.74
	ANN	0.1574	0.1508	0.0246	0.96
<b>Unknown Set</b>	HMM	0.1221	0.0971	0.1117	0.47

According to the experiments on the control data set, our proposed method (SCS) shows the lowest prediction error, compared to the others (articulatory test and intelligibility test) for both recognition systems (HMM and ANN). The experiment on the unknown set showed that our SCS is comparable to the other standard methods. The lowest prediction error came from intelligibility test followed by our method and the articulatory test with the errors of 9.7%, 11.17%, and 12.21%, consecutively.

## 5 Conclusion and Future Work

Speech assessments for people with speech disorder are very important. The current methods, articulatory test and intelligibility test, are subjective to each experts and listeners. Moreover, it is not clear that the standard evaluations gained by human will reflect the recognition rate of modern speech recognition models. In addition, both articulatory test and intelligibility test are a time consuming and labor task. This paper proposes a criterion called *speech consistency score* (SCS) which can be used not only to evaluate the severity of speech disorder but also to predict the possible accuracy outcome from a speech recognition system. The prediction can be served as a decision index whether this dysarthric speaker could be benefit from the technology or not.

As for future works, the research will scope on exploring more parameters such as the overlap factor, the energy and time consistency, to improve the accuracy of prediction for the unknown set. One more issue is how to incorporate word (or phoneme) density distribution into the model in order to construct a general framework for predicting the recognition accuracy for any set of speech.

## References

1. Deller, J., Hsu, D., Ferrier, L.: On the use of hidden Markov Modeling for Recognition of Dysarthric Speech. *Computer Methods and Programs in Biomedicine* **35** (1991) 125 -139
2. Kotler, A., and Thomas-Stonel, N.: Effects of speech training on the accuracy of speech recognition for an individual with speech impairment. *Journal of Augmentative and Alternative Communication* **12** (1997) 71- 80
3. Rosen, K., and Yampolsky, S.: Automatic Speech Recognition and a Review of Its Functioning with Dysarthric Speech. *Journal of Augmentative and Alternative Communication* **16** (2000) 46 - 60
4. Berenthal, J. E. and Bankson, N. W: *Articulation and phonological disorders* (3<sup>rd</sup> ed.). Prentice Hall, Boston (1993)
5. Bodt, M. S. D., Huici, M. E. H., Heyning, P. H. V. D.: Intelligibility as a Linear Combination of Dimensions in Dysarthric Speech. *Journal of Communication Disorders* **35** (2002) 283 - 292.

# An Emotion-Driven Musical Piece Generator for a Constructive Adaptive User Interface

Roberto Legaspi, Yuya Hashimoto, and Masayuki Numao

The Institute of Scientific and Industrial Research, Osaka University  
8-1 Mihogaoka, Ibaraki, Osaka, 567-0047, Japan  
{roberto, hasimoto}@ai.sanken.osaka-u.ac.jp,  
numao@sanken.osaka-u.ac.jp

**Abstract.** This paper presents the results of recent modification in the *Constructive Adaptive User Interface* (CAUI) that induces a model of emotional impressions towards certain musical piece structures and improvises a piece based on the model. The CAUI previously employed a ready-made melody generating module with its internal workings abstracted from the CAUI. Utilizing such black-box modules, however, may impede further effort to enhance the CAUI. To address this problem, a replacement module that automatically creates tunes tailored to the listener's impressions has been incorporated. Current results indicate that the CAUI may induce relevant relations that can support the adaptive improvisation of impression-causing tunes of a musical piece.

## 1 Introduction

To understand the significant link that unites music and the emotions has been a subject of considerable interest involving various fields (exemplified in [1]). Although the field of AI has played a crucial role in computer music for almost five decades (reviewed in [2]), and more recently, the few works in Machine Learning that aim to find regularities from musical performance examples (e.g., [8,7,6]), the consideration of emotions in *intelligent* music systems has received little attention.

This paper reports the results of current modification in the *Constructive Adaptive User Interface* (CAUI) [3] that induces a model of the listener's emotional impressions towards certain musical piece structures and subsequently re-arranges or composes a piece based on the model. The CAUI previously employed an external ready-made module, whose internal workings are abstracted from the CAUI, to generate a melody while necessitating some degree of manual support. Utilizing such black-box modules, however, may impede further effort to improve the CAUI. Hence, a tune-creating module that is adaptive to the listener's impressions has been integrated to the CAUI in place of the abstracted module. With this new module, the CAUI is able to generate a chord progression consisting of tones that make up a specific tune and alters certain tones based on an existing music theory thereby creating a non-monotonic musical piece.

## 2 Acquisition of Impression-Causing Musical Structures

The CAUI collects a person's impressions of certain musical pieces, based on which it extracts a musical structure causing a specific impression.

The CAUI compiles a listener's evaluations of musical pieces using a *web-based* evaluation instrument (illustrated in [3]). By using Osgood's semantic differential method in psychology, each subject can rate a piece as one of 5 grades for 6 pairs of impression adjectives, namely, *favorable-unfavorable*, *lively-dull*, *stable-unstable*, *beautiful-ugly*, *happy-sad*, and *heartrending-joyful*. For the first pair, for instance, a 5 means a piece is favorable and a 1 means otherwise. Each subjective rating in the 5-1 scale reflects the degree of the listener's impression. 75 well-known pieces, from which 8 or 16 successive bars were extracted, were evaluated by 14 subjects.

All the musical pieces used to train the CAUI were prepared in the predicate *music/2* form that represents an entire piece given 2 (as specified in the denominator) arguments, namely, a *song\_frame/7* and a list of *chord/12* predicates. The predicate *song\_frame/7* describes the musical frame in terms of tonality, rhythm and musical instrument, and the predicate *chord/12* describes a chord structure in terms of its main chord, key, root, duration and function.

Using the subjects' ratings and the predicate-represented musical pieces, the CAUI employs FOIL[4] to generate the rules that describe the musical structures that cause specific impressions, and subsequently uses *rx[5]* to refine the FOIL-obtained rules. Each rule is described using a target predicate. The CAUI aims to learn three kinds of target predicates, namely, *frame/1*, *pair/2*, and *triplet/3*, which represent the whole framework of music, and a pattern of two and three successive chords, respectively. The number of generated rules may vary with each subject. For instance, when a model was induced for subject A for his impression of a *heartrending* tune 10, 69, and 70 rules in the form of *frame/1*, *pair/2*, and *triplet/3*, respectively, were learned.

## 3 Composition of a Musical Piece

Evolutionary methods have been instrumental in music generation (e.g., [9]). The CAUI is distinct since its GA utilizes for its fitness function the model of user-specific impression-reflective structures and music theory.

The bit string in GA is extended to a row of columns in which the first column contains the bit-representation of the components of a *song\_frame/7*, and the rest of the columns contain *chord/12* bit-representations. A one-point cross-over splits and exchanges a whole column thereby creating alternative chord progressions, and mutation changes a column's structure thereby altering the music framework and the chord structure. A fitness function is employed to evaluate each possible alternative.

The fitness function reflects the user-specific model and music theory:

$$\text{Fitness\_Function}(M) = \text{Fitness\_User}(M) + \text{Fitness\_Theory}(M). \quad (1)$$

where  $M$  is a piece described by predicate *music/2*. This makes possible to generate a chord progression that fits the music theory and causes the required feeling. The *Fitness\_Theory(M)* penalizes a chord progression when it violates the music theory. The *Fitness\_User(M)* function is computed as:

$$\text{Fitness\_User}(M) = \text{Fitness\_Frame}(M) + \text{Fitness\_Pair}(M) + \text{Fitness\_Triplet}(M). \quad (2)$$

To compute for each function, the target predicates learned for each of the impression adjectives (*ia*) in the various pairs *ia*<sub>1</sub>-*ia*<sub>2</sub> are used. Table 1 specifies the computation algorithm and Table 2 specifies the meaning of each variable used in the algorithm.

**Table 1.** Algorithm for computing *Fitness\_Frame(M)*, *Fitness\_Pair(M)*, and *Fitness\_Triplet(M)*

- 
1. A list *L* of *Pred<sub>i</sub>* is extracted from *M* where *Pred<sub>i</sub>* depends on the targeted fitness function. The length of *L* is denoted by *n*.
  2. *m* patterns of *x*-successive *Pred<sub>i</sub>* are subsequently extracted from *L*. Each pattern is denoted as *P<sub>i</sub>*.
  3. Each *P<sub>i</sub>* is input to four other sub-functions, namely,  $\delta_F$ ,  $\delta'_F$ ,  $\delta_{FR}$ , and  $\delta'_{FR}$ :
    - $\delta_F$  returns +2 with reference to *ia*<sub>1</sub> and  $\delta'_F$  returns -2 with reference to *ia*<sub>2</sub> if *T* corresponds to any *T* from among the existing ones learned by FOIL for *ia*<sub>1</sub> and *ia*<sub>2</sub>.
    - $\delta_{FR}$  returns +1 with reference to *ia*<sub>1</sub> and  $\delta'_{FR}$  returns -1 with reference to *ia*<sub>2</sub> if *P<sub>i</sub>* corresponds to any *T* in the existing ones learned by FOIL and  $\wedge x$  for *ia*<sub>1</sub> and *ia*<sub>2</sub>.
 The returned values have been determined empirically. The returned values for *P<sub>i</sub>* from all four functions are averaged. In the event that both  $\delta_F$  and  $\delta_{FR}$  hold true, only  $\delta_F$  is counted. The same holds true for  $\delta'_F$  and  $\delta'_{FR}$ . The average is denoted as *Eval(P<sub>i</sub>)*.
  4. Each corresponding fitness function (e.g., *Fitness\_Frame(M)*) is computed as:

$$\sum_{i=1}^m \text{Eval}(P_i) . \quad (3)$$


---

**Table 2.** Meaning of each variable in the above algorithm

Fitness function	<i>Pred<sub>i</sub></i>	<i>x</i>	<i>m</i>	<i>T</i>
Fitness_Frame	song_frame/7	single	<i>n</i>	frame/7
Fitness_Pair	chord/12	two	<i>n</i> -1	pair/2
Fitness_Triplet	chord/12	three	<i>n</i> -2	triplet/3

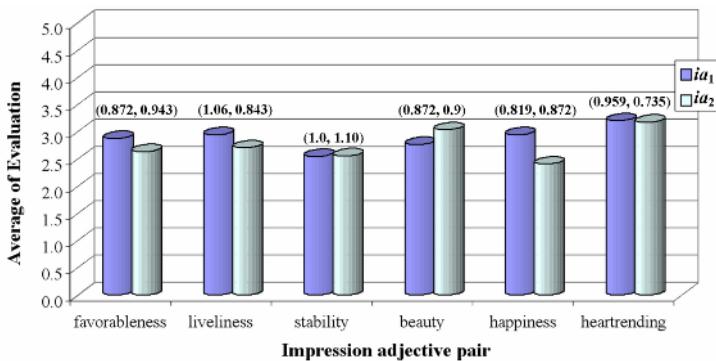
Fig.1 shows that the CAUI automatically composed a heartrending piece without any *handcrafted* background knowledge on this impression. For each participant, a total of 48 tunes were composed, i.e., 4 tunes for each impression adjective (4×6×2).

## 4 Evaluation of the Composed Pieces and Future Works

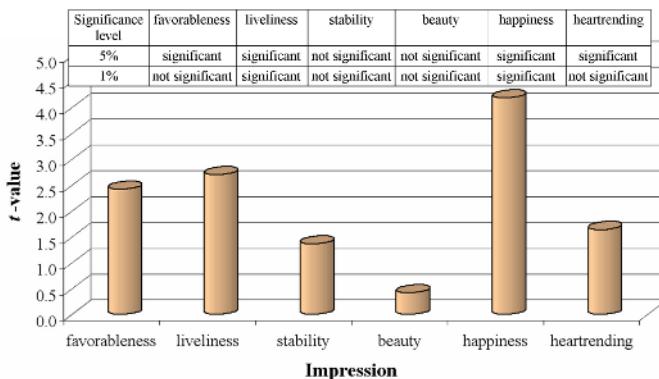
The subjects were asked to evaluate, this time, the composed musical pieces using the same evaluation instrument. Fig.2 shows the average results of the students' evaluations, where the parenthesized values indicate the standard deviation, for each impression adjective. Fig.3 shows the resulting t-test values and significance level when t-test was applied to the participants' evaluative ratings of each impression pair.



**Fig. 1.** A CAUI-composed *heartrending* musical piece



**Fig. 2.** Average results of the subjects' evaluations of composed pieces



**Fig. 3.** Results on performing t-test on the subjects' evaluations

The empirical results indicate that the CAUI has been moderately successful in creating tunes tailored to the subjects' impressions for 4 out of 6 adjective pairs. As for the adjective pairs for *stability* and *beauty*, satisfactory results were not achieved. This can be attributed to the lack in performance for creating adequately well-structured tunes from the viewpoint of music theory rather than any shortcoming in inducing the user-specific model of impression-structure relations.

The results obtained are acceptably sufficient in this developmental stage and provide the motivation to further enhance the CAUI's capability. Tunes obtained at this stage are rhythmically monotonic due to the exclusive use of 8-beat basic rhythmic pattern. The use of 4- and 16-beat would certainly contribute to a variety of patterns. Furthermore, the system only created 8-bar tunes and would need predicates that could represent musical periods that would provide more consistency to the tunes.

## 5 Conclusion

What makes the CAUI distinct from other systems is its machine-learning framework capable of creating emotion-driven musical pieces based on a model of impressions towards certain piece structures as generalized from examples. Its current implementation is an exploration attempt to automatic tune composition in which one's sensibility is reflected while keeping such knowledge relevant to music theory.

## References

1. Juslin, P.N., Sloboda, J.A. (eds.): *Music and Emotion: Theory and Research*. Oxford University Press, New York (2001)
2. Lopez de Mantaras, R., Arcos, J.L.: AI and Music: From Composition to Expressive Performances. *AI Magazine*, Vol. 23, No. 3 (2002) 43-57
3. Numao, M., Takagi, S., Nakamura, K.: Constructive Adaptive User Interfaces - Composing Music Based on Human Feelings. In Proceedings of the Eighteenth National Conference on Artificial Intelligence (2002) 193-198
4. Quinlan, J.R.: Learning Logical Definitions from Relations. *Machine Learning*, Vol. 5, Kluwer Academic Publishers, Boston (1990) 239-266.
5. Tangkitvanich, S., Shimura, M.: Refining a Relational Theory with Multiple Faults in the Concept and Subconcept. In *Machine Learning: Proceedings of the Ninth International Workshop* (1992) 436-444.
6. Thom, B: Unsupervised Learning and Interactive Jazz/Blues Improvisation. In Proceedings of the Seventeenth National Conference on Artificial Intelligence (2000) 652-657
7. Tobudic, A., Widmer G.: Relational IBL in Classical Music. *Machine Learning*, Springer Netherlands (2006)
8. Widmer, G.: Discovering Simple Rules in Complex Data: A Meta-learning Algorithm and Some Surprising Musical Discoveries. *Artificial Intelligence*, Vol. 146, No. 2 (2003) 129-148
9. Wiggins, G., Papadopoulos, G., Phon-Amnuaisuk, S., Tuson, A.: Evolutionary Methods for Musical Composition. *International Journal of Computing Anticipatory Systems* (1999)

# An Adaptive Inventory Control Model for a Supply Chain with Nonstationary Customer Demands

Jun-Geol Baek<sup>1</sup>, Chang Ouk Kim<sup>2</sup>, and Ick-Hyun Kwon<sup>3</sup>

<sup>1</sup> Department of Industrial Systems Engineering, Induk Institute of Technology,  
Wolgye-dong, Nowon-gu, Seoul, 139-749, Republic of Korea

<sup>2</sup> Department of Information and Industrial Engineering, Yonsei University,  
Sinchon-dong, Seodaemun-gu, Seoul, 120-749, Republic of Korea

<sup>3</sup> Department of Industrial Systems and Information Engineering, Korea University,  
Anam-dong, Sungbuk-gu, Seoul, 136-701, Republic of Korea

**Abstract.** In this paper, we propose an adaptive inventory control model for a supply chain consisting of one supplier and multiple retailers with nonstationary customer demands. The objective of the adaptive inventory control model is to minimize inventory related cost. The inventory control parameter is safety lead time. Unlike most extant inventory control approaches, modeling the uncertainty of customer demand as a statistical distribution is not a prerequisite in this model. Instead, using a reinforcement learning technique called action-reward based learning, the control parameter is designed to adaptively change as customer demand pattern changes. A simulation based experiment was performed to compare the performance of the adaptive inventory control model.

## 1 Introduction

This paper deals with the inventory replenishment problem of a single item in a two-stage supply chain system, in which a supplier replenishes the inventories of multiple retailers with nonstationary customer demands. By the nonstationary customer demand process, we mean that the mean and variance of demand distribution change along with time. At the retailer, if customer demands are not satisfied at sales points of time, the demands are treated as lost sales. Between the supplier and each retailer, there exist a constant transportation lead time. However, the retailers' actual lead times are not constants unless the supplier has enough amount of inventory to meet the retailers' orders.

Also, in this paper, we consider the varying reorder point system, based on which a centralized adaptive inventory control model is proposed. By adaptive, we mean that, as the customer demand changes, the inventory control model automatically adjusts reorder points in the direction of reducing inventory related costs. Therefore, associated decisions are concerned with when the retailers' inventories are replenished. In more detail, at each review period, the supplier accesses each retailer's inventory position and sales history data. With the data, the supplier anticipates the time point at which the inventory position of the retailer drops down below zero at first. If the time interval between the inspection time and the anticipated time is equal to the sum of the supplier's lead time and retailer's transportation lead time, then the supplier places

an order to the outside source. As soon as the supplier receives the ordered quantity, he/she delivers it directly to the retailer without keeping it in the supplier's warehouse.

If the anticipation is accurate, the retailer is guaranteed to replenish its inventory without delay. However, because anticipating the time point at which the inventory position of the retailer drops down below zero is not accurate especially in nonstationary demand trend, the supplier should have a time buffer to adaptively regulate his order release time. The order release time of the supplier is expedited or delayed using the time buffer. Positive (negative) time buffer makes it release an order to the outside source earlier (later) than total lead time. The control parameter of the supplier is the length of the time buffer. Hereafter, the time buffer is called safety lead time.

In this paper, the control parameter of the adaptive inventory control model is designed to adaptively change using an action-reward based learning approach, one of reinforcement learning techniques (Sutton and Barto 1998). In the context of the action-reward based learning, the learner or decision maker is called agent, and it interacts with non-static control domain. These two interact continuously, the agent selecting actions and the domain responding to those actions and giving rise to rewards, numerical values that are the inputs to the performance measure of the agent. The agent selects future actions based on the updated performance measure. For example, in the adaptive inventory control model, decision maker is the supplier, and the control domain is the two-stage supply chain wide inventory. Also, the performance measure is average inventory related cost occurred during lead times, and the action corresponds to safety lead time.

## 2 An Adaptive Inventory Control Model

We define the following notations to explain the adaptive inventory control model.

$L_0$ : lead time of supplier,

$L_i$ : transportation lead time of retailer  $i$  ( $i = 1, 2, \dots, N$ ),

$Q_i$ : order quantity of retailer  $i$ ,

$s_{ij}$ :  $j$  th safety factor of retailer  $i$ ,

$st_{ij}$ : safety lead time of the supplier when safety factor  $s_{ij}$  is applied,

$\hat{\sigma}_e(t)$ : standard deviation of forecast errors estimated at inspection time  $t$ ,

$D_i(t)$  ( $\hat{D}_i(t)$ ): (expected) customer demand at retailer  $i$  at inspection time  $t$ ,

$Z_i(t)$ : inventory level (if  $Z_i(t) \geq 0$ ) or shortage (if  $Z_i(t) < 0$ ) at site  $i$  at inspection time  $t$  ( $i = 0, 1, 2, \dots, N$ ),

$h_i$ : inventory holding cost per stock keeping unit (SKU) at site  $i$  ( $i = 0, 1, 2, \dots, N$ ),

$l_i$ : shortage cost per SKU at site  $i$  ( $i = 0, 1, 2, \dots, N$ ),

$C_i(t) = [Z_i(t)]^- \cdot l_i + [Z_i(t)]^+ \cdot h_i$ : chain wide inventory related cost at site  $i$  at inspection time  $t$  ( $i = 0, 1, 2, \dots, N$ ).

The supplier monitors the inventory position of retailer  $i$  ( $i = 1, 2, \dots, N$ ) and sales history at each discrete inspection time. With the sales history data, the supplier updates linear time series model  $\hat{D}_i(t') = a_0 + a_1 t'$ . This model is used for estimating the amount of customer demands at future time  $t'$ . In the model, coefficients  $a_0$  and  $a_1$  are also updated using exponential smoothing method (see Brown (1962) for detailed update formula).

At each inspection time  $t$ , the inventory position of retailer  $i$  at future time  $t'$  is defined as the inventory position observed at the inspection time  $t$  minus the sum of estimated demands during the time interval between the inspection time  $t$  and the future time  $t'$ . Now suppose that the future time  $t'$  is set to the time at which the inventory position of retailer  $i$  falls down to zero. Then the JIT delivery policy can be briefly stated as follows.

*At inspection time  $t$ , if  $\{(the\ time\ t'\ that\ the\ time\ series\ model\ predicts\ the\ inventory\ position\ of\ retailer\ i\ reaches\ zero) - t\} \leq L_0 + L_i + st_{ij}$ , then the supplier issues an order of  $Q_i$  to the outside source.* (1)

If the demand process is stationary and its variance is very small, the forecasting model will accurately estimate demand during the total lead time. As the result, the JIT delivery policy can replenish retailers' inventories at the time the inventory positions of retailers are close to zero. However, a problem arises if the retailers encounter sudden genuine changes in the underlying demand processes in terms of the changes of mean and (or) variance, resulting in the overestimation (or underestimation) of the demand. Of course, forecasting errors can be reduced to some extent with more sophisticated time series models. However, the models cannot fundamentally resolve the problem of forecasting errors generated due to the change of demand process. Safety lead time can adjust order placement time. For example, suppose that demand is underestimated. In this case, adding a positive safety lead time to actual lead time enforces the JIT delivery policy to place order earlier than the policy without the safety lead time. Since the delivery of ordered quantity takes the actual lead time, the JIT delivery policy considering positive safety lead time brings the effect of expediting order process. Similarly, forecasting with negative safety lead time will delay order process, and this is effective when demand is overestimated.

The action-reward based learning approach is used for determining appropriate safety lead time. In general, safety lead time can be obtained from a multiplication function of lead time and forecast error (Bernard 1999). Let  $S_i = \{s_{i1}, s_{i2}, \dots, s_{ik}\}$  is the set of safety factors for retailer  $i$ . Then, at inventory replenishment time, safety lead time  $st_{ij}$  corresponding to safety factor  $s_{ij}$  can be derived from

$$\text{Find } st_{ij} \text{ such that } \sum_{t'=t}^{t+st_{ij}} \hat{D}_i(t') = s_{ij} \times \hat{\sigma}_e(t) \times (L_0 + L_i)$$

where estimated standard deviation of forecast errors  $\hat{\sigma}_e(t)$  is commonly approximated as  $1.25 \times MAD$  (See Brown (1962) for detailed justification about the approximation).

Suppose that some  $s_{ij}$  is selected at an inventory replenishment time, and an order for retailer  $i$  is placed at inspection time  $t$  according to the JIT delivery policy specified in (1). The ordered quantity will be delivered to retailer  $i$  at time  $t + L_0 + L_i$ . Then the responsibility is reflected to value  $\bar{C}(s_{ij})$  of  $s_{ij}$  as follows:

$$\bar{C}_{new}(s_{ij}) = \bar{C}_{old}(s_{ij}) + StepSize \left[ \frac{1}{L_0 + L_i} \cdot \sum_{t'=t}^{t+L_0+L_i} C_i(t') - \bar{C}_{old}(s_{ij}) \right] \quad (2)$$

In this model, because the inventory level of the supplier is always zero and the supplier centrally controls the whole supply chain, the inventory holding and shortage costs of the supplier are zero. However, since the objective is the minimization of the total average cost of the supply chain, the supplier must select safety lead time based on the retailer's costs.

If  $\bar{C}_{new}(s_{ij})$  is reduced, then  $s_{ij}$  can be regarded as an appropriate safety factor for the current demand trend. Hence, the selection chance of  $s_{ij}$  at the next time should be increased. It is needless to say that the safety factors with least average costs should have great chance of being selected in order to reduce the total cost of the supply chain. Therefore, the next safety factor is determined according to the following rule.

$$\Pr\{\text{next safety factor} = s_{ij}\} = \frac{e^{1/\bar{C}_{new}(s_{ij})}}{\sum_{j=1}^k e^{1/\bar{C}_{new}(s_{ij})}} \quad (3)$$

The completed inventory control procedure of the adaptive inventory control model is explained as follows.

- Step 0. The supplier selects a safety factor  $s_{ij}$  initially for each retailer  $i$  ( $i = 1, 2, \dots, N$ ).
- Step 1. At inspection time  $t$ , if order placement condition (1) satisfies for retailer  $i$ , then the supplier issues an order of size  $Q_i$  to the outside source.
- Step 2. If the ordered quantity  $Q_i$  is arrived from the outside source, then the supplier immediately delivers  $Q_i$  to retailer  $i$ .
- Step 3. After retailer  $i$  receives the ordered quantity  $Q_i$ , the supplier updates  $\bar{C}_{new}(s_{ij})$  according to the learning formula in (2). Select the next safety factor according to the probabilistic rule in (3). Set  $s_{ij}$  = the next safety factor. Go to Step 1.

### 3 Simulation Based Experiment

The simulated supply chain consists of one supplier and four retailers. Different customer demand process is assumed for each retailer. Time interval for inspecting the retailers' inventory position and customer demands is set one day. The length of a simulation run is 5000 days. Given a specific demand process for each retailer, 20 simulation runs were performed and their average is measured. The set of safety factors for each

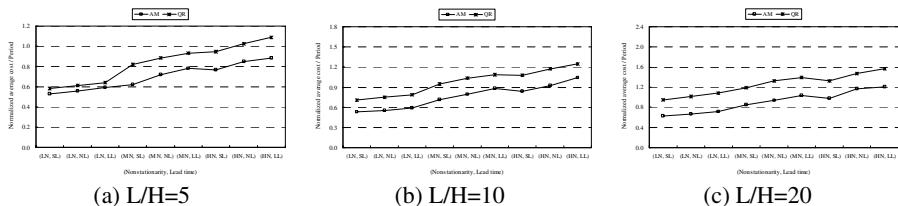
retailer  $i$  ( $i = 1, 2, 3, 4$ ) is defined as  $S_i = \{-1, -0.75, -0.5, -0.25, 0, 0.25, 0.5, 0.75, 1\}$ . Four experiment factors are considered: lead time, demand pattern, lost\_sales\_cost/inventory\_holding\_cost ratio (L/H ratio), and supplier\_cost/retailer\_cost ratio (S/R ratio). For each different combination of the levels of the factors, 20 simulation runs were performed and their average value was taken into consideration for performance comparison.

For lead time of site  $i$ , we consider three levels: short lead time (SL):  $L_i = 0.3 \times cycle_i$ , normal lead time (NL):  $L_i = 0.6 \times cycle_i$ , and long lead time (LL):  $L_i = 0.9 \times cycle_i$ , where  $cycle_i = Q_i / demand\ rate_i$  and  $demand\ rate_i$  is the average demand per period.

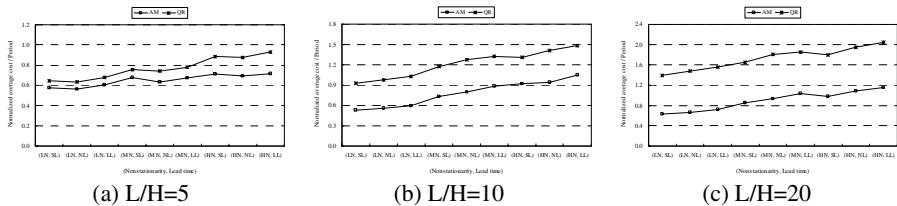
In this paper, we assume that the customer demand is nonstationary. In the case of nonstationary customer demand, the mean of the normal distribution is designed to change at every random interval  $T$  according to the rule of  $mean_j = mean_{j-1} + slope$ . In this rule,  $slope$  and  $T$  are randomly created by uniform distributions  $U(-sm, sm)$  and  $U(tu/2, tu)$ , respectively.  $sm$  and  $tu$  characterize the nonstationarity of demand process. In this experiment, we set the two parameters as follows: low nonstationarity (LN):  $sm = 1.0$  and  $tu = 30$ , medium nonstationarity (MN):  $sm = 2.0$  and  $tu = 15$ , and high nonstationarity (HN):  $sm = 4.0$  and  $tu = 8$ .

In general, since lost sales cost per item is larger than inventory holding cost per item, we consider three cases: small difference (L/H ratio = 5), medium difference (L/H ratio = 10), and large difference (L/H ratio = 20). Finally, we also consider two different cost structures between the supplier and the retailers as follows: normal difference (S/R ratio = 1/3), no difference (S/R ratio = 1).

From Fig. 1 and 2, we can observe that, in most cases, the performance of the adaptive inventory control model (AM) proposed in this paper is better than a traditional inventory control model such as (Q, R) model, which determines the reorder point using average customer demand. The reason would be that the adaptive inventory control model is designed to adaptively adjust the control parameter of the inventory control model as customer demand patterns change. As shown in Fig. 1-(c) and Fig. 2-(c), the performance of the adaptive inventory control model is more better than the traditional inventory control model (QR) when L/H ratio is high (e.g., L/H ratio = 20). This implies that AM is more effective when the lost sales cost per item is very large compared with the inventory holding cost. In addition, the effectiveness of AM becomes more obvious as S/R ratio becomes larger (e.g., S/R ratio = 1), which



**Fig. 1.** Simulation results in the case of S/R=1/3



**Fig. 2.** Simulation results in the case of S/R=1

can be found in a relationship between the supplier and retailers that are members of different companies. Fig. 2-(c) supports such results. This result would accrue from the fact that the supplier does not keep inventory any time in the adaptive inventory control model.

## 4 Conclusions

In most cases, due to unpredictable customer needs and economic situation, customer demands fluctuate with time, showing nonstationary patterns. To cope with this situation, we propose an adaptive, intelligent inventory control model, with the assumption that supplier is able to access online information about customer demand, as well as the inventory position of each retailer. Applying an action-reward based learning approach, one of reinforcement learning techniques, the control parameter of the inventory control model is designed to adaptively change as customer demand patterns change. A simulation based experiment was performed to compare the performance of the adaptive inventory control model.

## References

1. Achabal, D. D., McIntyre, S. H., Smith, S. A., and Kalyanam, K.: A decision support system for vendor managed inventory. *Journal of Retailing*. 76 (2000) 430-454
  2. Bernard, P.: *Integrated Inventory Management*. John Wiley & Sons (1999)
  3. Brown, R.: *Smoothing, Forecasting, and Prediction of Discrete Time Series*. Prentice-Hall (1962)
  4. Sutton, R. and Barto, A.: *Reinforcement Learning*. MIT Press (1998)
  5. Zhao, X. and Xie, J.: Forecasting errors and the value of information sharing in a supply chain. *International Journal of Production Research*. 40 (2002) 311-335

# Context-Aware Product Bundling Architecture in Ubiquitous Computing Environments\*

Hyun Jung Lee<sup>1</sup> and Mye M. Sohn<sup>2, \*\*</sup>

<sup>1</sup> Sungkyun Institute of Management Research  
Sungkyunkwan University  
Myung Ryun 3-53, Chong No-Ku, Seoul 110-745, Korea  
hjlee5249@yahoo.com

<sup>2</sup> Department of Systems Management Engineering  
Sungkyunkwan University  
300, Chunchun-dong, Jangan-gu, Suwon, Kyunggi-do, 440-746, Korea  
myesohn@skku.edu

**Abstract.** We propose **C**ontext-**A**ware **P**ROduct **B**undling **A**rchitecture (CARBA). It is necessary for the various products to be easily and immediately integrated, according to customers' changed requirements. In order to integrate information from various resources such as airline, hotel reservation, and so on, a semantic web service supporting an ontology based travel information system is required. CARBA is basically implemented as a semantic web service, with several components for reconfiguring a bundle of traveling products, and is guaranteeing traveler's mobility in ubiquitous computing environments.

## 1 Introduction

Mobile technology is constantly evolving to better fulfill mobile user's information service requirements, such as access of dispersed information on the web at any time and location [5]. However, it has been difficult to access changing context-dependent information and take information integration service to reflect changed context like alteration of traveling schedule, for high mobile user, even if mobile technology is constantly evolving. CARBA is basically designed as a product bundling and semantic web service, which are for performing of unpredictable requirements for relevant and actionable information to perform the task at hand [2], with several components to integrate services or products. In order to overcome data and information heterogeneity among services or products to be integrated, we adopt ontology as in [1]. The target problem addressed here is travel product bundling [6]. In this paper, we describe the architecture of CARBA and the context-aware traveling product bundling procedure with an illustrative example. Finally, the paper concludes with how the CARBA works effectively and a discussion on future research issues.

---

\* This work was supported by grant No. R01-2006-000-10303-0 from the Basic Research Program of the Korea Science & Engineering Foundation.

\*\* Corresponding author.

## 2 Overall Architecture of CARBA

CARBA has three components to reconfigure a bundle of traveling products as depicted in Figure 1.

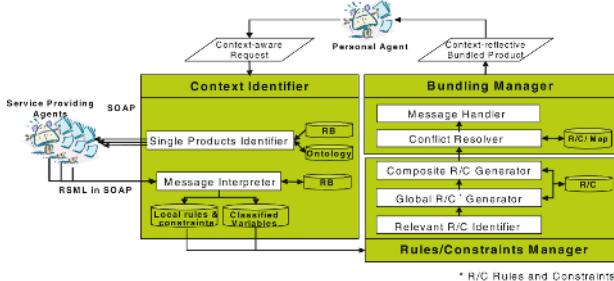


Fig. 1. Architecture of CARBA

In context identifier, single product identifier receives context-aware requests from mobile agents and parses the requests and message interpreter classifies the variables from single products into local and global variables using a rule base. In rules/constraints (R/C) manager, relevant R/C identifier and global R/C generator determines local and global relationship among variables, respectively. Composite R/C generator generates R/C within a bundled product. Finally, the conflict resolver resolves conflict between variables, and the message handler communicates with the service requester's agent, creating a context reflective bundled product.

## 3 Context-Aware Traveling Product Bundling Procedure

### 3.1 Context-Aware Procedure

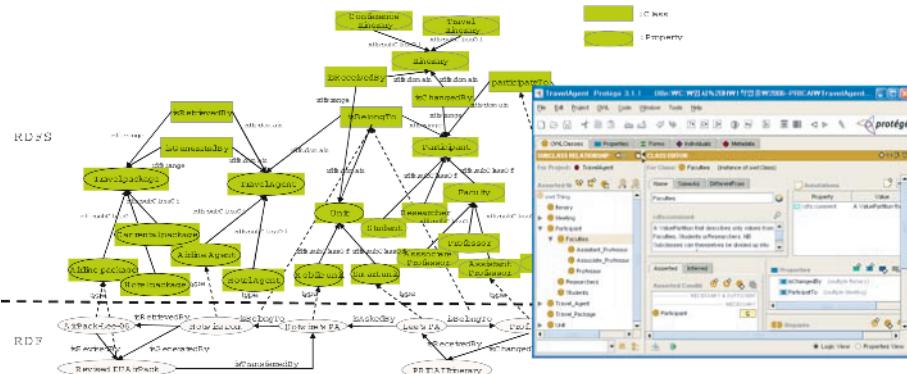
Two activities occur in the context-aware procedure: communication with the service providers and a single product identifier, and the classification of variables and rules/constraints using a message interpreter. The single product identifier decomposes traveler's requirements in the form of HTML, into rules and facts. It adapts the Rule Identification Markup Language (RIML) to identify rules and data implied in HTML. In this paper, rules and facts are manually identified by a knowledge engineer, but Park suggests ontology based rule acquisition [4]. Identified rules and facts are used to decide adequate service providers. An illustrative rule is as follows:

```

IF          ((NewFact.variable HAS DepartureCity
    AND (NewFact.variable HAS ArrivalCity))
    AND (NewFact.variable HAS DepartureDate))
THEN        SingleProduct IS Airline
  
```

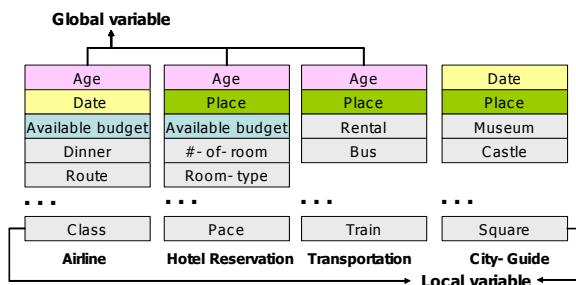
To reflect traveler's changed requirement, the single product identifier may interact with an airline service using above rule. Partial ontology for intelligent agents and interaction models is depicted in Figure 2. The Message interpreter classifies the local

rules, constraints and variables implied in the SOAP/RSML message. The SOAP/RSML message which is a variation of a SOAP message implies processing results, and rules [3] and/or constraints which should be announced to service requesters. The message interpreter also classifies variables into global variables which are shared among single products and local variables which are referenced only by a single product, and may be related to a local rule itself and extracts R/C.



**Fig. 2.** Partial Ontology for airline service

As in illustrated Figure3, the variable “Age” is a global variable because it is referenced by the Airline, Hotel reservation, and Transportation product. However, variable “Route” is only referenced by an airline product. It is called a local variable such as “Dinner” and “Class.”



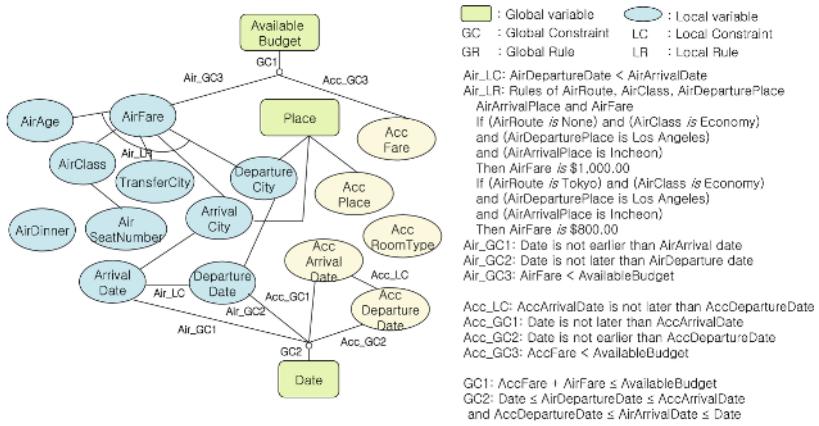
**Fig. 3.** Local and global variables

### 3.2 Rules and Constraints Generating Procedure

There are rules and constraints between configurable variables in a single product, which are defined in the product DB of each web service provider and extracted to bundle several single products. In general, configurable variables have other relevant variables, according to rules and constraints. Therefore, relevant rules and constraints identifier identifies the relationship among configurable variables which come from

several products. Global rules and constraints generator, for a single product to be bundled, generates global rules and constraints as follows: between global variable to bundle a product, and local variable from a single product, and between a global variable from a bundled product, and local variables with several single products.

Figure 4 represents a bundled product which is combined by Airline and Hotel reservation products. Local R/C represents the relationship between local variables. Global R/C is represented by the relationship between global and local variables within a single product. For instance, the local variable “AirFare” has relevance to global variable “AvailableBudget.” The local rules and constraints represent the relationship among local variables. They do not have any relationship with other variables arising from other single products such as “AirClass.”



**Fig. 4.** A bundled product of “Airline” and “Hotel Reservation”

Composite rules and constraints generator generates global R/C. For instance, GC1 is the composition of constraints Air\_GC3 and Acc\_GC3. Air\_GC3 and Acc\_GC3 are sources from Airline product and Hotel Reservation product, respectively. GC2 is the composition of constraints Air\_GC1, Air\_GC2, Acc\_GC1, and Acc\_GC2. This generator creates new rules and constraints between combined products.

### 3.3 Traveling Product Bundling Procedure

Conflict resolver resolves conflict among constraints of composite variables. Therefore the composite rule is newly generated and context-reflective bundled product is expressed using XML. Message handler transmits the context-reflective bundled product to the traveler requesting the services. This overall product bundling procedure is repeated until the traveler is satisfied with proposed context-reflective bundled product.

## 4 Evaluation

We conducted simulation using NetLogo software to evaluate the performance of the proposed CARBA, two types of product bundling system have been designed. One

type uses CARBA to search and bundle products, while the other is general search system, which customers have to access several sites to bundle products. This experimental design is based on the expectation that the CARBA would contribute more to the performance of context-aware bundling product bundling, which can be measured by the decrease in search costs. To add realism to our experiment, we assumed that there exist 27 customers and 100 search sites. A customer agent can modify its requirements freely within 10 times. To evaluate performance of CARBA, we tested the following hypothesis: ***The searching cost of bundling product in CARBA is equal to the cost obtained by general search system.*** To show the validity of the proposed approach, we performed simulation experiments using several types of changing requirements. Performance is measured using the cost function of each agent as follows.

$$\text{Cost of CARBA} = ((w_{C1} \times \text{search cost of site} + w_{C2} \times \text{no. of coordination} + w_{C3} \times \text{no. of bundling}) / (w_{C1} + w_{C2} + w_{C3})) \times \text{no. of requirements changing}$$

$$\text{Cost of General System} = ((w_{G1} \times \text{search cost of sites} + w_{G2} \times \text{coordination cost} + w_{G3} \times \text{bundling cost}) / (w_{G1} + w_{G2} + w_{G3})) \times \text{no. of requirements changing}$$

$w_{C1}$  and  $w_{G1}$  represents the weight of search,  $w_{C2}$  and  $w_{G2}$  represents weight of coordination, and  $w_{C3}$  and  $w_{G3}$  represent weight of bundling. A paired  $t$ -test with 161 pairs was conducted to verify if the costs obtained from CARBA are significantly lower. We found that  $t = 21.34$ . Thus the test rejects the null hypothesis, which means the search costs of bundling products in CARBA is significantly ( $p < 0.0001$ ) lower than those of the general search system.

## 5 Conclusion

The ubiquitous computing world is emerging more rapidly than expected, therefore, it is becoming clear that the use of a context-aware decision support for mobile users would provide significant benefits in most mobility situations, including the traveling product bundling problem. The CARBA was designed for bundling of traveling products which requires a high level of ubiquity in order to fulfill the changing requirements of travelers as a semantic web service. With CARBA, it is expected that the high mobility issues of travelers is handled more effectively. In order to prove this assumption, an experimental situation is illustrated with an illustrative context-aware traveling product bundling problem consisting of travelers, and service providers, serving processing results, local rule, and constraints of single product, and CARBA. It is believed that this research provides a foundation for realizing semantic web efforts faster, providing improved quality of service for mobile users in ubiquitous environments.

## References

1. Caragea, D., et al., Information Integration and Knowledge Acquisition from Semantically Heterogeneous Biological Data Sources. In: Proc. of the 2nd Int. Workshop on Data Integration in Life Sciences (DILS'05), San Diego, CA (2005)

2. Choueiry, B.Y. and Noubir, G.: On the Computation of Local Interchangeability in Discrete Constraint Satisfaction Problems, Proc. AAAI-98, pp. 326 - 333 (1998)
3. Lee, J.K. and Sohn, M.: eXtensible Rule Markup Language. Communications of the ACM. Vol. 46, No. 5, pp. 59-64 (2003)
4. Lyytinene, K., and Yoo. Y. Research Commentary: The Next Wave of Nomadic Computing, Information Systems Research, Vol. 13, No. 4, pp. 377-388 (2002)
5. Weiser, M. Ubiquitous Computing, IEEE Computer (1993)
6. Werthner, H., and Ricci F. "E-Commerce and Tourism," Communication of ACM, Vol. 47, No. 12, pp. 101-105 (2004)

# A Relaxation of a Semiring Constraint Satisfaction Problem Using Combined Semirings

Louise Leenen<sup>1</sup>, Thomas Meyer<sup>2</sup>, Peter Harvey<sup>1</sup>, and Aditya Ghose<sup>1</sup>

<sup>1</sup> Decision Systems Laboratory, School of IT and Computer Science,  
University of Wollongong, Australia  
[{11916, pah06, aditya}@uow.edu.au](mailto:{11916, pah06, aditya}@uow.edu.au)

<sup>2</sup> National ICT Australia  
School of Computer Science and Engineering,  
University of New South Wales, Sydney, Australia  
[thomas.meyer@nicta.com.au](mailto:thomas.meyer@nicta.com.au), [tmeyer@cse.unsw.edu.au](mailto:tmeyer@cse.unsw.edu.au)

**Abstract.** The Semiring Constraint Satisfaction Problem (SCSP) framework is a popular approach for the representation of partial constraint satisfaction problems. In this framework preferences (semiring values) can be associated with tuples of values of the variable domains. Bistarelli et al. [1] define an abstract solution to a SCSP which consists of the best set of solution tuples for the variables in the problem. Sometimes this abstract solution may not be good enough, and in this case we want to change the constraints so that we solve a problem that is slightly different from the original problem but has an acceptable solution. In [2] we propose a relaxation of a SCSP where we define a measure of distance (a semiring value from a second semiring) between the original SCSP and a relaxed SCSP. In this paper we show how the two semirings can be combined into a single semiring. This combined semiring structure will allow us to use existing tools for SCSPs to solve Combined Semiring Relaxations of SCSPs. At this stage our work is preliminary and needs further investigation to develop into a useful algorithm.

## 1 Introduction

The considerable interest in *over-constrained problems*, *partial constraint satisfaction problems* and *soft constraints* is motivated by the observation that with most real-life problems, it is difficult to offer *a priori* guarantees that the input set of constraints to a constraint solver is solvable. Many real-life problems are inherently over-constrained. In order to solve an over-constrained problem we have to identify appropriate *relaxations* of the original problem that are solvable. Early approaches to such relaxations largely focussed on finding maximal subsets (with respect to set cardinality) of the original set of constraints that are solvable (such as Freuder and Wallace's work on the MaxCSP problem [3]). Subsequent efforts considered more fine-grained notions of relaxation, where entire constraints did not have to be removed from consideration ([4], [5], [6]).

Bistarelli et al. [1] proposed an abstract semiring CSP scheme that generalised most of these earlier attempts, while making it possible to define several

useful new instances of the scheme. The SCSP scheme assumes the existence of a semiring of abstract preference values, such that the associated multiplicative operator is used for combining preference values, while the associated additive operator is used for comparing preference values. An SCSP constraint assigns a preference value to all possible value assignments to the variables in its signature. These preferences implicitly define a relaxation strategy.

In our previous paper [2] we define how an SCSP may be relaxed by introducing a mechanism by which we can minimally alter (or relax) constraints of the problem. We also introduce a measure of distance between an original constraint and its relaxed version that is modeled via a second semiring.

Our aim in this paper is to show that we can combine the first semiring (these semiring values are used as preference values associated with tuples of constraints) and the second semiring (these semiring values are used as distance values between constraints and relaxed constraints) into a single semiring. With a single semiring we can resort to existing SCSP tools to solve Relaxed SCSPs.

## 2 The SCSP Framework and Relaxations of SCSPs

This section contains a summary of the SCSP framework of Bistarelli et al. [1], as well as a summary of the results in [2] where we propose a technique to relax the constraints of the original problem.

**Definition 1.** A *c-semiring* is a tuple  $S = \langle A, +, \times, \mathbf{0}, \mathbf{1} \rangle$  such that

- $A$  is a set with  $\mathbf{0}, \mathbf{1} \in A$ ;
- $+$  is defined over (possibly infinite) sets of elements of  $A$  as follows: for all  $a \in A$ ,  $\sum(\{a\}) = a$ ,  $\sum(\emptyset) = \mathbf{0}$  and  $\sum(A) = \mathbf{1}$ , and  $\sum(\bigcup_{i \in I} A_i, i \in I) = \sum(\{\sum(A_i), i \in I\})$  for all sets of indices  $I$ . (When  $+$  is applied to sets of elements, we use the symbol  $\sum$ .)
- $\times$  is a commutative, associative, and binary operation such that  $\mathbf{1}$  is its unit element and  $\mathbf{0}$  is its absorbing element, and  $\times$  distributes over  $+$ .

The elements of the set  $A$  are the preference values to be assigned to tuples of values of the domains of constraints. Let  $\leq_S$  be a partial order over  $A$ :  $\alpha \leq_S \beta$  iff  $\alpha + \beta = \beta$ .  $\mathbf{0}$  is the minimum element and  $\mathbf{1}$  is the maximum element.

**Definition 2.** Consider a constraint system  $CS = \langle S_p, D, V \rangle$  where  $S_p = \langle A_p, +_p, \times_p, \mathbf{0}_p, \mathbf{1}_p \rangle$  is a *c-semiring*,  $V$  is an ordered finite set of variables, and  $D$  is a finite set of allowed values for the variables in  $V$ . A constraint over  $CS$  is a pair  $c = \langle \text{def}_c^p, \text{con}_c \rangle$  with  $\text{con}_c \subseteq V$ , and  $\text{def}_c^p : D^k \rightarrow A_p$  ( $k$  is the cardinality of  $\text{con}_c$ ). A Semiring Constraint Satisfaction Problem (SCSP) over  $CS$  is a pair  $P = \langle C, \text{con} \rangle$  where  $C$  is a finite set of constraints over  $CS$  and  $\text{con} = \bigcup_{c \in C} \text{con}_c$ .  $\langle \text{def}_{c_1}^p, \text{con}_c \rangle \in C$  and  $\langle \text{def}_{c_2}^p, \text{con}_c \rangle \in C$  implies  $\text{def}_{c_1}^p = \text{def}_{c_2}^p$ .

**Definition 3.** Given a constraint system  $CS = \langle S_p, D, V \rangle$  where  $S_p = \langle A_p, +_p, \times_p, \mathbf{0}_p, \mathbf{1}_p \rangle$ , and two constraints  $c_1 = \langle \text{def}_{c_1}^p, \text{con}_{c_1} \rangle$  and  $c_2 = \langle \text{def}_{c_2}^p, \text{con}_{c_2} \rangle$  over  $CS$ , their combination,  $c_1 \otimes c_2$ , is the constraint  $c = \langle \text{def}_c^p, \text{con}_c \rangle$  with  $\text{con}_c = \text{con}_{c_1} \cup \text{con}_{c_2}$  and  $\text{def}_c^p(t) = \text{def}_{c_1}^p(t \downarrow_{\text{con}_{c_1}}) \times_p \text{def}_{c_2}^p(t \downarrow_{\text{con}_{c_2}})$ .

See [1] for the definition of the projection  $t \downarrow_W^W$ , of a tuple  $t$  from a set  $W$  to a set  $W'$ , and the definition of the projection  $c \Downarrow I$  of a constraint  $c = \langle \text{def}_c^p, \text{con}_c \rangle$  over set  $I$  of variables. A solution to an SCSP is a single constraint formed by the combination of all the original constraints. An *abstract solution* consists of the set of  $k$ -tuples of  $D$  whose  $c$ -semiring values are maximal w.r.t.  $\leq_{S_p}$ .

**Definition 4.** Given an SCSP  $P = \langle C, \text{con} \rangle$  over a constraint system  $CS$ , the solution of  $P$  is a constraint  $\text{Sol}(P) = (\bigotimes C) = \langle \text{def}_c^p, \text{con} \rangle$  where  $\bigotimes C = c_1 \otimes c_2 \otimes \dots \otimes c_n$  with  $C = \{c_1, \dots, c_n\}$ . The set  $\text{ASol}(P) = \{\langle t, v \rangle \mid \text{def}_c^p(t) = v \text{ and there is no } t' \text{ such that } v <_{S_p} \text{def}_c^p(t')\}$  is the abstract solution and  $\text{ASolV}(P) = \{v \mid \langle t, v \rangle \in \text{ASol}(P)\}$  contains the maximal preference values.

**Definition 5.** [7] Let a good enough (abstract) solution for a SCSP  $P$  be such that some element in  $\text{ASolV}(P)$  is in the region  $\hat{\beta}$  where  $\hat{\beta} = \{\gamma \in A : \beta \leq_{S_p} \gamma\}$ .

If  $\text{ASolV}(P) \cap \hat{\beta} = \emptyset$  we want to find a relaxation  $P'$  of  $P$ , such that  $\text{ASolV}(P') \cap \hat{\beta} \neq \emptyset$ .  $P'$  should be as close to the original  $P$  as possible.

**Definition 6.** A constraint  $c_j = \langle \text{def}_j^p, \text{con}_j \rangle$  is called a  $c_i$ -weakened constraint of the constraint  $c_i = \langle \text{def}_i^p, \text{con}_i \rangle$  iff the following hold:  $\text{con}_i = \text{con}_j$ ; for all tuples  $t$ ,  $\text{def}_i^p(t) \leq_S \text{def}_j^p(t)$ ; and for every two tuples  $t_1$  and  $t_2$ , if  $\text{def}_i^p(t_1) <_{S_p} \text{def}_i^p(t_2)$ , then  $\text{def}_j^p(t_1) <_{S_p} \text{def}_j^p(t_2)$ .

**Definition 7.** Given a constraint system  $CS = \langle S_p, V, D \rangle$  and an SCSP  $P = \langle C, \text{con} \rangle$ , for each  $c \in C$ , let  $W_c$  be the set containing all  $c$ -weakened constraints, i.e.  $W_c = \{c_j \mid c_j \text{ is a } c\text{-weakened constraint}\}$ . Let  $S_d = \langle A_d, +_d, \times_d, \mathbf{0}_d, \mathbf{1}_d \rangle$  be a  $c$ -semiring and  $wdef_c^d : W_c \rightarrow A_d$  be any function such that the following hold:  $wdef_c^d(c_j) = \mathbf{0}$  iff  $c_j = c$ ;  $\forall c_i, c_j \in W_c$ , if for all tuples  $t$   $\text{def}_i^p(t) \leq_{S_p} \text{def}_j^p(t)$  then  $wdef_c^d(c_i) \leq_{S_d} wdef_c^d(c_j)$ ; and if there exists one tuple  $t$  such that  $\text{def}_i^p(t) <_{S_p} \text{def}_j^p(t)$  and for all tuples  $s$  we have  $\text{def}_i^p(s) \leq_{S_p} \text{def}_j^p(s)$ , then  $wdef_c^d(c_i) <_{S_d} wdef_c^d(c_j)$ .

**Definition 8.** – The  $c$ -weakened constraint  $c_i$  is closer to  $c$  than the  $c$ -weakened constraint  $c_j$ , iff  $wdef_c^d(c_i) <_{S_d} wdef_c^d(c_j)$ .  
– The  $c$ -weakened constraint  $c_i$  is no closer to  $c$  than the  $c$ -weakened constraint  $c_j$ , iff  $wdef_c^d(c_j) \leq_{S_d} wdef_c^d(c_i)$ .  
– The  $c$ -weakened constraints  $c_i$  and  $c_j$  are incomparable w.r.t. closeness to  $c$  iff  $wdef_c^d(c_i) \not\leq_{S_d} wdef_c^d(c_j)$  and  $wdef_c^d(c_j) \not\leq_{S_d} wdef_c^d(c_i)$ .

The function  $wdef_c^d$  assigns a distance value from the set of the  $c$ -semiring  $S_d$  to each  $c$ -weakened constraint, and is restricted as follows. Let  $c_{ik}$  be a  $c_i$ -weakened constraint, and  $c_{jm}$  and  $c_{jn}$  be  $c_j$ -weakened constraints. If  $wdef_{c_j}^d(c_{jm}) <_{S_d} wdef_{c_j}^d(c_{jn})$ , then  $wdef_{c_i}^d(c_{ik}) \times_d wdef_{c_j}^d(c_{jm}) <_{S_d} wdef_{c_i}^d(c_{ik}) \times_d wdef_{c_j}^d(c_{jn})$ .

**Definition 9.** A SCSP  $P' = \langle C', \text{con} \rangle$  is a  $d$ -relaxation of the SCSP  $P = \langle C, \text{con} \rangle$  where  $S_d = \langle A_d, +_d, \times_d, \mathbf{0}_d, \mathbf{1}_d \rangle$ , iff there is a bijection  $f : C \rightarrow C'$  and  $\forall c \in C$ ,  $f(c)$  is a  $c$ -weakened constraint. Let  $R(P) = \{P' \mid P' \text{ is a } d\text{-relaxation of } P\}$ , and  $R_{\hat{\beta}}(P) = \{P' \in R(P) \mid \text{ASolV}(P') \cap \hat{\beta} \neq \emptyset\}$ .

A  $d$ -relaxation  $P' = \langle C', \text{con} \rangle$  of  $P = \langle C, \text{con} \rangle$  is such that every c-weakened constraint  $c' \in C'$  is the closest possible to the constraint  $c \in C$  while the abstract solution of  $P'$  is still good enough (w.r.t.  $\hat{\beta}$ ).

**Definition 10.** Given a  $d$ -relaxation  $P' = \langle C', \text{con} \rangle$  of a SCSP  $P = \langle C, \text{con} \rangle$  such that  $P' \in R_{\hat{\beta}}(P)$ , let  $d(P') = \times_d \min_{c \in C} (\text{def}_c^d(f(c)))$  be the distance between  $P$  and  $P'$ . The set  $MR_{\hat{\beta}}(P) = \{P' \in R_{\hat{\beta}}(P) \mid \nexists P'' \in R_{\hat{\beta}}(P) \text{ such that } d(P'') <_S d(P')\}$  contains the relaxations closest to  $P$ .

### 3 A Combined Semiring

**Definition 11.** Suppose  $S_A = \langle A, \oplus_A, \otimes_A, \mathbf{0}_A, \mathbf{1}_A \rangle$  and  $S_B = \langle B, \oplus_B, \otimes_B, \mathbf{0}_B, \mathbf{1}_B \rangle$  are two  $c$ -semirings. Let a Combined  $C$ -Semiring be  $S_U = \langle U, \oplus_U, \otimes_U, \mathbf{0}_U, \mathbf{1}_U \rangle$  with  $U = \{\langle \langle a_1, \dots, a_k \rangle, b \rangle \mid a_i \in A, \text{ and } b \in B\}$  for some fixed non-negative integer  $k$ . If we have  $u_1, u_2 \in U$ , with  $u_1 = \langle \langle a_{1_1}, \dots, a_{1_k} \rangle, b_1 \rangle$  and  $u_2 = \langle \langle a_{2_1}, \dots, a_{2_k} \rangle, b_2 \rangle$ , then the following statements hold.

- $u_1 \otimes_U u_2 = \langle \langle a_{1_1} \otimes_A a_{2_1}, \dots, a_{1_k} \otimes_A a_{2_k} \rangle, b_1 \otimes_B b_2 \rangle$ .
- $u_1 \oplus_U u_2 = \langle \langle a_{1_1} \oplus_A a_{2_1}, \dots, a_{1_k} \oplus_A a_{2_k} \rangle, b_1 \oplus_B b_2 \rangle$ .
- $\mathbf{0}_U = \langle \langle a_1, \dots, a_k \rangle, b \rangle$  such that every  $a_i = \mathbf{0}_A$  and  $b = \mathbf{0}_B$ , and  $\mathbf{1}_U = \langle \langle a_1, \dots, a_k \rangle, b \rangle$  such that every  $a_i = \mathbf{1}_A$  for  $i = \{1, \dots, k\}$ , and  $b = \mathbf{1}_B$ .
- A pre-order  $\leqslant_U$  over the set  $U$  is defined as  $u_1 \leqslant_U u_2$  iff  $b_1 \leqslant_B b_2$ .

**Definition 12.** Let  $P = \langle C, \text{con} \rangle$  be an SCSP over a constraint system  $CS = \langle S_p, D, V \rangle$  and  $P' = \langle C', \text{con} \rangle$  be a  $d$ -relaxation of  $P$ . A Combined Semiring Relaxation of  $P$  is a tuple  $\langle P', g \rangle$  with  $S_U = \langle U, \oplus_U, \otimes_U, \mathbf{0}_U, \mathbf{1}_U \rangle$ , where  $g : C \times C' \rightarrow U$ , i.e. for every  $c = \langle \text{def}_c^p, \text{con}_c \rangle \in C$  and every  $c$ -weakened constraint  $c_r \in C'$ ,  $g(\langle c, c_r \rangle) = u_{cr}$  with  $u_{cr} \in U$ .

Assume all tuples of values of  $D$  are strictly ordered. Let  $u_{cr} = \langle \text{Pref}_{cr}, b_{cr} \rangle$ , where  $b_{cr}$  is the distance value associated with the constraint  $c_r$ , and  $\text{Pref}_{cr} = \langle a_{cr_1}, \dots, a_{cr_k} \rangle$  where  $a_{cr_i}$ , for  $i = \{1, \dots, k\}$ , are the preference values associated with the constraint  $c_r \otimes_p c_{BEST}$ . Let  $c_{BEST} = \langle \text{def}_{c_{BEST}}^p, \text{con} \rangle$  be a dummy constraint with  $\text{def}_{c_{BEST}}^p(t) = \mathbf{1}_p$  for every tuple  $t$  over the set of variables  $\text{con}$ .

Note that  $\text{def}_{cr \otimes_p c_{BEST}}^p(t) = \text{def}_{cr}^p(t \downarrow \text{con}_c) \otimes_p \mathbf{1}_p = \text{def}_{cr}^p(t \downarrow \text{con}_c)(t)$ . The coordinates in the set  $\text{Pref}_{cr}$  are the preference values associated with the  $k$  tuples in the relaxed constraint (over the variables in the set  $\text{con}$ ) while  $b_{cr}$  represents the distance between the relaxed problem  $P'$  and the original problem  $P$ .

**Definition 13.** Given a Combined Semiring Relaxation  $RP = \langle P', g \rangle$  of an SCSP  $P = \langle C, \text{con} \rangle$  and a  $d$ -relaxation  $P' = \langle C', \text{con} \rangle$ , the solution of  $RP$  is a constraint defined as  $R\text{Sol}(RP) = (\bigotimes C')$  with  $g(\bigotimes C, \bigotimes C') = u_{CR}$ . Suppose  $u_{CR} = \langle \text{Pref}_{CR}, b_{CR} \rangle$ , and  $\text{Pref}_{CR} = \langle a_{CR_1}, \dots, a_{CR_k} \rangle$ . Then the abstract solution of  $RP$  is the set  $R\text{ASol}(RP) = \{\langle t, a \rangle \mid a \in \text{Pref}_{CR}, t \text{ is the tuple with which } a \text{ is associated, and there is no } \langle t', a' \rangle \text{ such that } a <_{S_p} a'\}$ . Let  $ASolV(RP) = \{a \mid \langle t, a \rangle \in R\text{ASol}(RP)\}$ .

Let  $\text{Rel}(P) = \{RP \mid RP \text{ is a Combined Semiring Relaxation of an SCSP } P = \langle C, \text{con} \rangle\}$ . Now we define a set containing the best Combined Semiring Relaxations with solutions that are good enough.

**Definition 14.** Suppose for every  $RP = \langle P', g \rangle \in \text{Rel}(P)$  with  $P' = \langle C', \text{con} \rangle$ , we have  $g(\bigotimes C, \bigotimes C') = u_{CR} = \langle \text{Pref}_{CR}, b_{CR} \rangle$ , and  $\text{Pref}_{CR} = \langle a_{CR_1}, \dots, a_{CR_k} \rangle$ . Let  $\text{Rel}_{\hat{\alpha}}(P) = \{RP \in \text{Rel}(P) \mid \text{RASolV}(RP) \cap \hat{\alpha} \neq \emptyset \text{ and there is no relaxation } PR' \in \text{Rel}(P) \text{ such that } b_{CR} <_{S_d} b_{CR'}\}$ .

## 4 Conclusion and Future Work

If the preference value associated with the abstract solution of an SCSP is not regarded as good enough, a suitable relaxation of the SCSP that has a good enough solution is found by adjusting the preferences associated with the tuples of some of the constraints (i.e. c-semiring values of the first semiring) of the original SCSP. In other words, the constraints of the original problem are relaxed until the resulting problem has a satisfactory solution. Distance values (i.e. c-semiring values from a second semiring) are associated with each relaxed constraint so that different relaxations of a problem can be compared in terms of their distance to the original problem. In this paper we show how to combine these two semirings into a single semiring.

The combined semiring allows us to rely on existing techniques for solving SCSPs. At this stage we simply have a technical result and need to investigate computational aspects of this process. We aim to develop techniques to calculate solutions to a maximal Combined Relaxation of SCSP efficiently.

## References

1. Bistarelli, S., Montanari, U., Rossi, F.: Semiring-based constraint solving and optimization. *Journal of the ACM* **44**(2) (1997) 201–236
2. Leenen, L., Meyer, T., Ghose, A.: Relaxations of semiring constraint satisfaction problems. In: Proceedings of the International Conference on Constraint Programming Preferences and Soft Constraints Workshop (SOFT-05). (2005)
3. Freuder, E.C., Wallace, J.W.: Partial constraint satisfaction. *Artificial Intelligence* **58** (1992) 21–70
4. Wilson, M., Borning, A.: Hierarchical constraint logic programming. *Journal of Logic Programming* **16** (1993) 277–318
5. Dubois, D., Fargier, H., Prade, H.: The calculus of fuzzy restrictions as a basis for flexible constraint satisfaction. In: Proc. of IEEE Conference on Fuzzy Systems. (1993)
6. Fargier, H., Lang, J.: Uncertainty in constraint satisfaction problems: a probabilistic approach. In: Proc. ECSQARU. (1993)
7. Ghose, A., Harvey, P.: Partial constraint satisfaction via semiring CSPs augmented with metrics. In: Proceedings of the Australian Joint Conference on AI. Volume 2557 of Lecture Notes in Computer Science., Springer (2002)

# Causal Difference Detection Using Bayesian Networks

Tomoko Murakami and Ryohei Orihara

Corporate Research & Development Center  
1, Komukai Toshiba-cho, Saiwai-ku Kawasaki 212-8582, Japan  
`{tomoko.murakami, ryohei.orihara}@toshiba.co.jp`

**Abstract.** In analysis of the market, detecting not only differences in consumer groups or changes but also their causal factors observed in consumer behavior is expected because it enables the marketer to take marketing actions. Although rule-discovery approaches can efficiently identify differences in groups or changes, it is still difficult to explain the causes of them. In this paper we propose an algorithm to detect causal differences in two bayesian networks by search and probability inference. We perform some experimental studies to analyze consumer behavior in purchasing personal computer.

## 1 Introduction

Understanding consumers' purchasing behavior of your own and/or competitors' products is important for marketing and customer relationship management. There has been interests in discovery of knowledge related to consumer behavior using data mining methods since data acquiring technologies such as point of sales(POS) system were developed. Detecting differences is one of the basic and important tasks in consumer behavior analyses. Difference is interpreted in two aspects, one is difference between several groups, and the other is difference in a single group as it varies through attributes such as time or location. For example, the former is difference in brand selection between male and female consumers, and the latter is change of female consumers' purchasing attitude in 3 years.

In recent years *contrast-set-mining*, which is skill to detect differences in several groups, is proposed and related studies are reported [1,2,3]. There are several rule-discovery approaches specifically designed for identifying the differences between several contrasting groups . They can detect efficiently differences in groups or changes, but it is still difficult to explain the causes of them. In analysis of the market, detecting not only differences in consumer groups or changes but also their causal factors observed in consumer behavior is expected because it enables the marketer to take marketing actions.

In this paper, we propose a method, Causal Difference Detection using Bayesian Networks(*CDDBN*) to identify causal differences in two models based on bayesian networks. CDDBN realizes discovery of causal differences by search and probability inference in bayesian networks. With it, it is possible to efficiently discover the causes of differences in consumer groups or trend changes in the market.

## 2 Related Works

Tian proposed a method of discovering causal relations from data, based on detection and interpretation of local spontaneous changes in the environment [4]. The method is evaluated by  $\chi^2$  test to investigate if causal relations change as time variation. Although Tian showed discovering changes of causal relations in data under the dynamic environment, the reason why changes cause is not discussed. Dong and Li worked on the problem of discovering *emergent patterns* (EPs) [1], which defines a difference in support values for an item in different data sets based on association rule discovery [5]. Bay developed STUCCO algorithm for mining contrast sets by combining statistical hypothesis testing with search [2]. They showed the reduction of processing time for efficient mining and summarized results by applying the algorithm to UCI data set. Webb conducted experiments using Magnum Opus [6], STUCCO [2], C4.5 [7] to mine the differences between contrasting groups [3]. They showed that Magnum Opus could successfully perform the task and also discussed how new and valuable contrast-set-mining is. With these rule-discovery approaches, it is possible to detect differences between two or more probability distributions, but still difficult to discover causal differences. Thus we can obtain answers for a query such as "How does group A differ from B?", but answers for a query such as "Why does group A differ from B?" cannot be returned.

## 3 Causal Difference Detection

We propose causal difference detection algorithm, which is a bayesian method to discover factors which cause differences by search and probability inference in bayesian networks. We illustrate CDDBN. The details of CDDBN algorithm is shown in figure 1. Input data is two bayesian networks and a target probability. The two bayesian networks are composed of same graph structure but distinct probability distribution. Output is a report to describe differences and causal differences in target probability for two bayesian network.

Let us denote them as  $(G, P)$  and  $(G, P')$ , where  $G = (V, E)$  is a graph composed of variables  $V$  and edges  $E$ ,  $P$  and  $P'$  are probability distributions. A trigger to start a search is user input of a target probability for analysis, for example, a probability  $P(c_1)$  meaning that a variable  $C \in V$  takes value  $c_1$ . The task is to discover differences in frequency or causal relation which cause differences in target probability in  $(G, P)$  and  $(G, P')$ . Suppose that  $\{C, L_1, \dots, L_n\} \in V$ ,  $\{L_1, \dots, L_n\}$  are parents of  $C$  in  $G$  and taking arbitrary values. If the user inputs target probability, for example  $P(c_k)$ , then search based on graph structure and probability inference in bayesian networks starts. A difference in a target probability  $\Delta P(c_k)$  in  $(G, P)$  and  $(G, P')$  is calculated by utilizing a parent variable  $L_i = \{l_1, \dots, l_m\}$  in graph structure as follows. we obtain equation (1) by assigning  $P'(c_k|l_i) = P(c_k|l_i) + \Delta P(c_k|l_i)$  and  $P'(l_i) = P(l_i) + \Delta P(l_i)$ .

$$\Delta P(c_k) \doteq \sum_{i=1}^m (P(c_k|l_i) \cdot \Delta P(l_i) + \Delta P(c_k|l_i) \cdot P(l_i)). \quad (1)$$

**Input:** Two Bayesian Networks  $(G, P)$  and  $(G, P')$ ,  
 where  $G = (V, E)$  is a graph composed of variables( $V$ ) and edges( $E$ ).  
**Output:** causal differences in probabilities between  $(G, P)$  and  $(G, P')$ .  
**Let**  $D$  be a set of all values of  $V$   
**Let**  $N \subseteq V, v \subseteq D$   
**Let**  $th1, th2$  be thresholds  
**Let**  $report$  be a function for reporting causal differences  
**Function:**  $reportDiff(G, N, v, D)$   
**Begin**

1. **for** (each  $q \in D$ )
  2.   **if** ( $q = v$ ) **return**
  3.    $D \leftarrow v;$
  4.   **if** ( $P'(N = v) \cdot \log(P'(N = V)/P(N = V)) < th1$ ) **return**
  5.    $flag \leftarrow \text{false}$
  6.   **for** (each parent  $Prt$  of  $N$ )
    7.     **for** (each value  $k$  of  $Prt$ )
      8.       **if** ( $P(N = v|Prt = k) < th2$ ) **continue**
      9.        $\Delta P_{OccDiff} \leftarrow P'(Prt = k) \cdot \log(P'(Prt = k)/P(Prt = k))$
      10.       **if** ( $\Delta P_{OccDiff} > th1$ )
        11.            $flag \leftarrow \text{true}$
        12.            $report(N, v, Prt, k)$
        13.            $reportDiff(G, Prt, k, D)$
      14.       **if** ( $flag = \text{true}$ ) **return**
    15.   **for** (each parent  $Prt$  of  $N$ )
      16.     **for** (each value  $k$  of  $Prt$ )
        17.       **if** ( $P(Prt = k) < th2$ ) **continue**
        18.        $\Delta P_{RelDiff} \leftarrow P'(N = v|Prt = k) \cdot \log(P'(N = v|Prt = k)/P(N = v|Prt = k))$
        19.       **if** ( $\Delta P_{RelDiff} > th1$ )  $report(N, v, Prt, k)$

**End**

**Fig. 1.** Algorithm for CDBBN

Equation (1) indicates that difference in target probability are approximately obtained by calculating frequency  $P(l_i)$ , causal relation  $P(c_k|l_i)$  and  $\Delta P(l_i)$  and  $\Delta P(c_k|l_i)$  differences in  $(G, P')$  and  $(G, P)$  respectively.

If a difference in target probability is greater than threshold number  $th1$  in figure 1, the search of the causes are performed according to equation (1). Differences in causal relation are calculated according to the latter part of equation (1). The probability of frequency  $P(l_i)$  is firstly calculated for each  $i(i = 1 \dots m)$ , then  $\Delta P(c_k|l_i)$  is calculated for  $i$  if  $P(l_i)$  is greater than threshold number  $th2$ . Differences in frequency are similarly calculated according to the former part of equation (1). The probability of causal relation  $P(c_k|l_i)$  is firstly calculated for each  $i(i = 1 \dots m)$ , then  $\Delta P(l_i)$  is calculated for  $i$  if  $P(c_k|l_i)$  is greater than threshold number  $th2$ . If  $\Delta P(l_i)$  is greater than threshold number  $th1$ , the calculation as described above is executed recursively for  $L$  and  $L$ 's parents in  $G$ . The search in bayesian networks  $(G, P)$  and  $(G, P')$  is continued until reaching a root node or visiting all parents of the target variable.

CDBBN assumes that causal relation is not dramatically variant, so that we fix causal structures and introduce two bayesian networks, which are composed of same graph structure but distinct probability distribution. If causal structures are correctly defined as bayesian networks, it is possible to efficiently detect even minor but critical causal differences observed in consumer behavior avoiding redundant search following causal structures. Furthermore, it is possible to detect rare event due to using ratio of probabilities as criteria in detecting causal differences.

## 4 Causal Difference in Consumer Behavior

We conducted experiments in personal computer(PC)'s consumer behavior with CDDBN. With it, we detect causal differences between two consumer groups or changes in a single consumer group through the time. But we describe only the result of detecting causal differences between two consumer groups due to the limitation of the space in the paper.

We detected causal differences in selecting 6 major PC brands by consumer groups in the Kanto and the Kansai, which are eastern and western region of Japan respectively. We firstly prepared two consumer behavior models for each consumer groups, then set a probability of selecting each PC brand as a target probability for analysis, that is, trigger to start a search. The consumer behavior models are bayesian networks constructed by utilizing knowledge possessed by marketers and survey data regularly conducted to investigate PC market. The survey data is obtained by means of questionnaires and collected from 3000 respondents who range in age from 20 to 59. We adopted 24 variables related to consumers' profile, groups, purchasing reason and brand selection in consumer behavior model. They include 3 unobserved variables introduced by marketers, which are related to tendency to follow people around, tendency to lead in the market and computer skills. Conditional probabilities for the observed variables in the bayesian networks are computed based on frequency distribution in the data. For unobserved variables, on the other hand, that are approximately computed by applying expectation-maximization(EM) algorithm [8]. Parameters in the EM algorithm, that are the number of iteration of the *expectation* and *maximization* steps  $i$  and the number of sample data to compute expected sufficient statistics  $N$ , we set  $i = 100$  and  $N = 2000$  to converge to a local maximum. We constructed consumer behavior models by region and time by this means. We secondly applied CDDBN to consumer behavior models. The task was to detect causal differences in region or time. We experimentally set two thresholds for detecting difference  $th1 = 0.05$ ,  $th2 = 0.3$  in CDDBN. we set  $th1 = 0.05$  because approximately 5% increase and decrease from 10% is frequently observed as differences in consumer behavior. we set  $th2 = 0.3$  because high frequency and strong causal relation are experimentally to be obtained in consumer behavior. In CDDBN, probabilities are computed by probability inference, where we use a variation of likelihood weighting [9].

As the result of applying CDDBN algorithm, a probability of selecting brand C in Kanto is larger than that in Kansai. CDDBN reports detected differences and its causal differences. It was revealed that factor causing difference in probability of selecting brand C is that consumers who don't mind PC's reputation are greater in the number in Kanto compared with those in Kansai. Additionally, we obtained two factors causing difference in the number of consumers who don't mind PC's reputation ("reputation = 0") as follows.

*Factor1.* The number of consumers who purchase whatever she/he wants to buy is greater in Kanto due to having an annual household income from 5 to 10 million yen.

*Factor2.* The number of consumers who have advanced values and, either being female, or in age of 10 or 40, is greater in Kanto.

Our results here have shown that CDDBN can detect causal differences in consumer behavior. Marketers in our company's PC division evaluated the results as newly discovered and valuable knowledge.

We compared our result to that of  $\chi^2$  test to investigate differences in frequency and causal relations based on data. It is verified that half of the detected differences by CDDBN are significant between datasets related to consumer groups in the Kanto and the Kansai. As equation (1) indicates that difference in target probability is derived from not only differences in frequency and causal relations but frequency and causal relations themselves, the result of  $\chi^2$  test does not necessarily show the lack of the validity of CDDBN.

## 5 Conclusion

In this paper, we proposed a method *CDDBN* to identify causal differences in two models based on bayesian networks. *CDDBN* realizes discovery of causal differences by search and probability inference in bayesian networks. We conducted experimental studies of discovery the causes of differences in consumer groups in personal computer's market. We consider that *CDDBN* is a new technique for contrast-discovery task in the point of causal difference detection. However it leaves room to be refined. It would be interesting to investigate the issue of correction of difference in causal relation. It would also be interesting to extend *CDDBN* to be able to apply to multiple bayesian networks more than two. We will continuously conduct further experiments and refine the algorithms.

## References

1. Dong, G. and Li, J.: Efficient mining of emerging patterns: Discovering trends and differences. Proc. of the fifth International Conference on Data Mining and Knowledge Discovery, ACM Press (1999) 43-52
2. Bay, D. S. and Pazzani, J. M.: Detecting Group Differences: Mining Contrast Sets. Int. J. Data Mining and Knowledge Discovery, **5(3)** (2001) 213-246
3. Webb, I. G., Butler, S. and Newlands D.: On detecting differences between groups. Proc. of the ninth International Conference on Data Mining and Knowledge Discovery, ACM Press (2003) 256-265
4. Tian, J. and Pearl, J.: Causal Discovery from Changes. Proc. of Uncertainty in Artificial Intelligence (2001) 512-521
5. Agrawal, R., Imielinski, T. and Swami, A.: Mining associations between sets of items in massive databases. Proc. of the tenth Internatinal Conference on Data Management of Data, ACM Press (1993) 207-216
6. Webb, I. G., Magnum Opus version 1.3. Computer software [<http://www.rulequest.com/>]. Distributed by Rulequest Research (2001)

7. Quinlan, J. R.: C4.5: Programs for Machine Learning. Morgan Kaufmann, San Mateo, CA (1993)
8. Dempster, P. A., Laird, M. N. and Rubin, B. D.: Maximum likelihood from incomplete data via the EM algorithm. Royal Statistical Society, B39 (1977) 1-38
9. Fung, R. and Chang, K. C.: Weighting and integrating evidence for stochastic simulation in Bayesian networks. Proc. of Uncertainty in Artificial Intelligence (1989)

# Tabu Search for Generalized Minimum Spanning Tree Problem

Zhenyu Wang<sup>1</sup>, Chan Hou Che<sup>2</sup>, and Andrew Lim<sup>1,2</sup>

<sup>1</sup> School of Computer Science & Engineering

South China University of Technology, Guang Dong, P.R. China

<sup>2</sup> Dept of Industrial Engineering and Logistics Management

Hong Kong Univ of Science and Technology, Clear Water Bay, Kowloon, Hong Kong

wangzy@scut.edu.cn, oscarche@ust.hk, iealim@ust.hk

**Abstract.** The Generalized Minimum Spanning Tree (GMST) problem requires spanning exactly one node from every cluster in an undirected graph. GMST problems are encountered in telecommunications network planning. A Tabu Search (TS) for the GMST problem is presented in this article. In our computational tests on 194 TSPLIB instances, TS found 152 optimal solutions. For those 42 unsolved instances, our algorithm has improved some previously best known solutions. Lower bounds of some unknown problems are improved by our heuristic relaxation algorithm.

## 1 Introduction

The Generalized Minimum Spanning Tree (GMST) problem is defined as follows: Given an undirected graph  $G = (V, E)$ , where  $V$  is the node set partitioned into clusters  $V_k, 1 \leq k \leq m$ ,  $V_i \cap V_j = \emptyset (i \neq j)$  and  $E = \{(i, j) | i \in V_k, j \in V_l, k \neq l\}$  is the Edge set, we need to find a minimum spanning tree including exactly one node from each cluster. The GMST problem is in telecommunications where local area networks (LAN) must be connected with each other [1].

The GMST problem is NP-hard as proved by Myung et al[1]. Feremans et al.[2] described eight formulations for the GMST problem. In Golden et al.[5], a local search and a Genetic Algorithm (GA) were developed to provide solutions for instances including up to 226 nodes. Pop et al. [8] proposed a relaxation model for GMST with up to 240 nodes. Another version of GMST which was required to span at least one node from each node cluster was studied by Dror et al.[4] and Feremans et al.[3]. Haouar et al.[9] proposed two stochastic heuristics and a Lagrangian based lower bound for this variant GMST problem.

This paper proposes a Tabu Search (TS) for the GMST problem while we will implement the GA proposed by Golden et al.[5]. The computation results show that TS is more effective than the GA. In addition, we will develop a relaxation algorithm to estimate the lower bound (LB). The result is promising, in which the relaxation is able to find a better LB than previous results.

The remainder of this paper is organized as follows. In Section 2, we introduce a Tabu Search. Then we describe a lower bounding heuristic method in Section 3.

The computation result is reported in Section 4 while our conclusions follow in Section 5.

## 2 Description of the Heuristic

The TS algorithm is implemented by three phases, **Basic Tabu Search** (BTS) phase, **intensification** phase and **diversification** phase.

Algorithm 1 describes the BTS in pseudocode. The procedure will randomly generate a feasible solution  $S = (v_1, v_2, \dots, v_m)$ ,  $v_i \in V_i$ , as initial solution. Then we will select a cluster  $R_i$  and replace all the nodes in selected cluster  $R_i$  to generate new solutions. Our heuristic approach uses a tabu queue list with fixed length  $l$ , which records the cluster visited by recent iterations. The algorithm will stop until the number of iterations in which the search progresses without any improvement is greater  $k_{max}$ .

```

Result: Solution  $S = (v_1, v_2, \dots, v_m)$ 
Data:  $G$  is the Graph
Create initial feasible solution  $S$ ;
repeat
    Randomly choose Cluster  $R_i$  ;
    if  $R_i \notin$  Tabu then
        foreach node  $v$  from Cluster  $R_i$  do
            // $v_i$  represent Cluster  $R_i$  in  $S$  ;
             $S' \leftarrow S - v_i + v$  ;
            // $MST(S)$  is the function that calculates the cost of solution  $S$ .
            if  $MST(S') < MST(S)$  then  $S \leftarrow S'$  ;
        end
        add  $R_i$  in the Tabu ;
    end
until no improvement after  $k_{max}$ ;
```

**Algorithm 1.** Basic Tabu Search

When designing the algorithm, we found some nodes always appeared in the local optimal solution. These nodes are more likely to appear in the global optimal solution than others. Therefore, we intend to develop a new searching space which is generated by a list of local optimal solutions; so-called “elite candidate list”. We restarted the BTS fifty times and selected the best five solutions as elite candidates. An intensive search space  $G' = (V', E')$ ,  $V'_i \subseteq V_i$ ,  $E' = \{(v_i, v_j) | v_i \in V'_k, v_j \in V'_l, k \neq l\}$  is generated by grouping these five solutions. To avoid solutions being trapped in local optimal, we applied a threshold strategy to increase diversity of the intensive search space, i.e. we multiplied a random value with the number of differences between the subcluster and old cluster. If the result is higher than the threshold, the subcluster will be replaced by the old cluster. Finally, the overall algorihtm is described in Algorithm 2.

```

Result: Solution  $S$ 
Data:  $G$  is the Graph
while  $i < 50$  do
     $S' = BTS(G);$ 
    Add  $S'$  in to Candidate List
     $i \leftarrow i + 1;$ 
end
 $EliteList \leftarrow Top5(CandidateList);$ 
while length of  $EliteList > 1$  do
    //getIntensiveSpace is a function to get a new search space
     $G' \leftarrow \text{getIntensiveSpace}(EliteList) ;$ 
     $S' \leftarrow BTS(G') ;$ 
    if  $MST(S') < MST(S)$  then add  $S'$  into  $EliteList$ ;
    Remove the largest one in elite list ;
end

```

**Algorithm 2.** Overall Tabu Search Algorithm

### 3 Lower Bound Study

We will present a relaxation approach to generate the relaxation instance of the GMST problem. **Step 1.** Generate the set of elite candidates  $T = \{t_1, t_2, \dots, t_k\}$  using the BTS described in Section 2.1. Set  $t_j = \{(v_1, v_2, \dots, v_m) | v_i \in V_i\}$  as a feasible solution. Therefore, subgraph  $G' = (V', E')$  mentioned in Section 2.2 can be obtained. **Step 2.** Generate a set  $V'' = V - V'$  and subset of each cluster  $V''_k = V_k - V'_k$ . **Step 3.** The graph  $G = (V, E)$  is divided into two different subgraphs, elite and non-elite graphs. Choosing one of the subgraphs, we transfer each clusters in this subgraph into a node. The cost of an edge is as follows:

$$e = \begin{cases} e_{ij} & i, j \notin V'' \\ \min \{e_{ij} | j \in V(n_j)\} & i \notin V'', j \in V' \\ \min \{e_{ij} | i \in V(n_i), j \in V(n_j)\} & i, j \in V'' \end{cases}$$

After transformation, we obtain the relaxation of the graph  $G$ .

### 4 Computational Study

TS and GA were coded in Java. All tests were run on a Dual-core Xeon 3GHz Server with 2GB of RAM. The set of instances in our experiment is a subset of the instances described in Fischetti et al. [6]. This subset includes the TSP instances from TSPLIB 2.1(Reinelt[7]) having  $48 \leq v \leq 318$  where the instances with 48 to 226 nodes are identical to those in Feremans et al. [2] and Golden et al. [5]. We also considered geographical problems in [2] and [5]. Fischetti et al.[6] provided two procedures to generate node clusters, center clustering and grid clustering. For grid clustering, there is a parameter  $\mu$  defined by the user. We set  $k_{max} = 50$  and length of tabu list =  $0.12|V|$ . The threshold was set to be 1.5. We

**Table 1.** Computational Result for TS and GA on TSPLIB Instance where the optimal solution is unknown

Problem name	K	E	LB/UB(%)	Previous best	TS		GA	
					Soln	CPU(s)	Soln	CPU(s)
TSPLIB instances, center clustering								
d198	40	18841	94.12	7044	<b>7044</b>	123	7053	149
gr202	41	19532	94.63	243	<b>242</b>	84	242	118
ts225	45	24650	99.78	62315	<b>62268</b>	94	62504	165
pr226	46	24626	100.0*	55515	<b>55515</b>	103	55515	171
gr229	46	25036	90.22	—	<b>787</b>	101	788	171
gil262	53	33507	78.66*	—	<b>942</b>	230	966	259
pr264	53	34028	89.02*	—	<b>21886</b>	267	22028	258
pr299	60	43786	71.09*	—	<b>20316</b>	307	20725	392
lin318	64	49363	84.6*	—	<b>18501</b>	350	18788	437
TSPLIB instances, grid clustering $\mu = 3$								
d198	67	19101	89.05*	8283	<b>8283</b>	312	8287	261
gr202	68	19826	90.44*	293	<b>293</b>	158	293	227
ts225	75	24900	98.75*	79019	<b>79019</b>	301	79408	317
pr226	84	25118	87.22*	62527	<b>62527</b>	207	62527	366
gr229	81	25584	89.3*	—	<b>935</b>	262	935	369
gil262	95	33845	78.53*	—	<b>1318</b>	408	1329	584
pr264	101	34357	—	—	<b>29208</b>	367	29239	594
pr299	102	44114	—	—	<b>23141</b>	753	23743	811
lin318	108	49821	—	—	<b>24152</b>	819	24380	902
TSPLIB instances, grid clustering $\mu = 5$								
d198	40	18772	91.07	7098	<b>7098</b>	94	7098	134
gr202	41	19303	90.52*	232	<b>232</b>	93	232	111
ts225	45	24726	92.38	60659	<b>60639</b>	105	60886	158
pr226	50	24711	98.38	56721	<b>56721</b>	117	56721	191
gr229	47	25129	90.04	—	<b>713</b>	146	714	175
gil262	63	33643	72.4*	—	<b>1018</b>	244	1047	334
pr264	55	34016	91.53	—	<b>21365</b>	141	21513	258
pr299	69	43893	65.95*	—	<b>18614</b>	506	19330	475
lin318	64	49320	78.13*	—	<b>17696</b>	304	17879	429
TSPLIB instances, grid clustering $\mu = 7$								
d198	32	18372	88.72*	6501	<b>6501</b>	56	6501	88
gr202	31	18872	91.13	203	<b>203</b>	45	203	70
ts225	35	24544	98.95	50813	<b>50813</b>	53	50889	109
pr226	33	24355	100.0	48249	<b>48249</b>	44	48249	101
gr229	34	24595	84.66	—	<b>626</b>	55	627	95
gil262	49	33487	73.7*	—	<b>829</b>	122	843	226
pr264	43	33481	85.76	—	<b>20455</b>	74	20461	183
pr299	47	43463	76.17	—	<b>15255</b>	106	15521	258
lin318	49	49106	79.63*	—	<b>14931</b>	114	15191	282
TSPLIB instances, grid clustering $\mu = 10$								
d198	25	18149	91.32	6185	<b>6185</b>	34	6185	60
gr202	21	17904	93.79	177	<b>177</b>	27	177	38
ts225	25	24300	99.9	40339	<b>40339</b>	31	40339	64
gr229	23	23761	87.96	—	<b>515</b>	31	515	52
gil262	36	33208	73.44*	—	<b>655</b>	73	679	144
pr264	27	32814	97.46	—	<b>16554</b>	64	16554	90
pr299	35	43054	74.91	—	<b>11640</b>	94	11781	167
lin318	36	48664	76.72	—	<b>10139</b>	136	10201	174

first tested our relaxation procedure on 44 instances where the optimal solution is unknown. In our experiment, testing both types of relaxation with different lengths of lists ( $l = 5, 10, 20, 30, 40$ ), we generated ten relaxation instances for each instance. In addition, we tested the instance without the relaxation procedure. We used the model described in Pop et al. [8] coding in CPLEX 9.0 in Java. The running time for each instance was 1 hour.

In Table 1, the instances with asterisks mean the lower bound was improved. 19 out of 44 instances were found to have better lower bounds. In addition, 2 instances received the optimal solution. Therefore, the number of instances where the optimal solution is known increased to 152. We tested 194 instances in total. GA found 144 optimal solutions and TS found all optimal solutions. Table 1 shows the results for the instance where the optimal solution is unknown. TS improves the solution from 3 instances and obtains same solutions provided in [5] for all others instances.

## 5 Conclusions

In this paper, we provided a Tabu Search for the GMST problem. We also presented a relaxation procedure to estimate the lower bound for the GMST problem. Our result is noteworthy that we improved the lower bound from previous results. In addition, we solved the instances up to 318 nodes.

## References

1. Myung, Y.S., C.H. Lee, D.W.Tcha. On the generalized minimum spanning tree problem. Networks **26** 231–241, 1995.
2. Feremans, C. General Spanning Trees and Extensions. PhD thesis, Institut de Statistique et de Recherche Opérationnelle, Université Libre de Bruxelles, Bruxelles, Belgium, 2001.
3. Feremans, C., M.Labbé, G.Laporte. The generalized minimum spanning tree problem: Polyhedral analysis and branch-and-cut-algorithm Networks **43** 71-86, 2004.
4. Dror, M., M. Haouari, J. Chaouachi. Generalized spanning tree. Eur. J. Oper. Res. **120** 583-592, 2000.
5. Golden, B., S.Raghavan, D.Stanojević. Heuristic Search for the Generalized Minimum Spanning Tree Problem. INFORMS J. Comput. **17(3)** 290-304, 2005
6. Fischetti, M., J.J. Salazar-Gonzalez, P. Toth. Symmetric generalized traveling salesman problem. Oper. Res. **45** 378-394, 1997.
7. Reinelt, G. TSPLIB, A traveling salesman problem library. INFORMS J. Comput. **3** 376-384, 1991.
8. Pop,P.C, W.Kern, G.Still, A new relaxation method for the generalized minimum spanning tree problem. Eur.J.Oper.Res. **170** 900-908, 2006.
9. Haouari, M. and Chaouachi, J.S. Upper and lower bounding strategies for the generalized minimum spanning tree problem. Eur.J.Oper.Res. **171** 632-647, 2006.

# Investigation of Brood Size in GP with Brood Recombination Crossover for Object Recognition

Mengjie Zhang<sup>1,2</sup>, Xiaoying Gao<sup>1,2</sup>, Weijun Lou<sup>1</sup>, and Dongping Qian<sup>2</sup>

<sup>1</sup> School of Mathematics, Statistics and Computer Science  
Victoria University of Wellington, P. O. Box 600, Wellington, New Zealand  
<sup>2</sup> Artificial Intelligence Research Centre  
Agricultural University of Hebei, Baoding, China  
`{mengjie, xgao, norman}@mcs.vuw.ac.nz, {zmj, qdp}@hebau.edu.cn`

**Abstract.** This paper describes an approach to the investigation of brood size in the brood recombination crossover method in genetic programming for object recognition problems. The approach is examined and compared with the standard crossover operator on three object classification problems of increasing difficulty. The results suggest that the brood recombination method outperforms the standard crossover operator for all the problems in terms of the classification accuracy. As the brood size increases, the system effective performance can be improved. When it exceeds a certain point, however, the effective performance will not be improved and the system will become less efficient.

## 1 Introduction

Since the early 1990s, genetic programming (GP) [1,2] has been applied to a range of object recognition problems such as shape classification, face identification, and medical diagnosis [3,4,5,6]. While showing promise, current GP techniques are limited and frequently do not give satisfactory results on difficult classification tasks. One main problem is that the standard crossover operator is not sufficiently powerful to generate good solutions [2]. In the current crossover operator, two sub-programs (crossover points) are randomly chosen from two parent programs, and two new programs are generated by simply swapping them. However, the totally random choice is clearly unable to guarantee the best choice.

To improve the standard crossover operator, Tackett [7] introduced the “brood recombination” method. In this method, a “brood”  $N$  is created for each crossover operation. The standard crossover operation is repeated  $N$  times on the two same parent programs selected from the population and  $2N$  child programs are generated. These child programs are then evaluated and their fitness ranked. The two programs with the best fitness are considered the “real” children of the parents and retained, but other children are discarded.

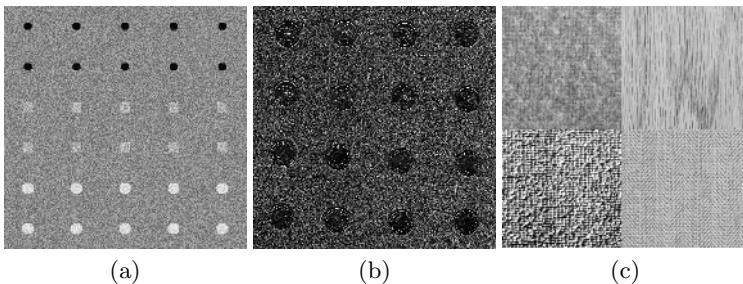
Clearly, the brood size is a key parameter in this approach. This size might be task dependent and related to the evolutionary process parameters such as the number of generations. However, this parameter was not properly investigated in the previous work.

The goal of this paper is to further analyse the brood size in the brood recombination crossover method. To do this, the brood recombination crossover with different brood sizes will be compared with the standard crossover operator in GP on three object classification problems of increasing difficulty. We will investigate whether a larger brood size can lead to better performance and whether there exists a certain point for the brood size beyond which the system performance will not be improved.

## 2 The Approach

### 2.1 Image Data Sets

Experiments were conducted on three different image data sets providing object classification problems of increasing difficulty. Sample images for each data set are shown in figure 1.



**Fig. 1.** Sample images in datasets: (a) Shape; (b) Coins; (c) Texture

The first data set (*shape*, figure 1a) was generated to give well defined objects against a reasonably noisy background. The pixels of the objects were produced using a Gaussian generator with different means and variances for each class. Four classes of 600 small objects (150 for each class) were cut out from the images and used to form the classification data set. The four classes are: *dark circles*, *grey squares*, *light circles* and *noisy background*.

The second set of images (*coin*, figure 1b) contains scanned 10 cent New Zealand coins. The coins were located in different places with different orientations and appeared in different sides (head and tail). The background was also cluttered. Three classes of 500 objects were cut out from the large images to form the data set. The three classes are: *head*, *tail* and *background*. Among the 500 cutouts, there are 160 cutouts for *head*, 160 cutouts for *tail* and 180 cutouts for *background* respectively. Compared with the *shape* data set, the classification problem in this data set is harder. Although these objects are still regular, the problem is quite hard due to the noisy background and the low resolution.

The third set of images (figure 1c) contains four different kinds of texture images, which are taken by a camera under the natural light. The images are

taken from a web-based image database held by SIPI of USC [8]. The four texture classes are named *woollen cloth*, *wood grain*, *raffia* and *herringbone weave* respectively. Because they are quite similar in many aspects, this classification task is expected to be more difficult than that in the *coin* data set. There are 900 sample cutouts from four large images, and each class has 225 samples. This dataset is referred to as *texture*.

For these data sets, the objects were equally split into three separate data sets: one third for the training set used directly for learning the classifiers, one third for the validation set for controlling overfitting, and one third for the test set for measuring the performance of the learned program classifiers.

## 2.2 GP Settings

In the approach, we used the tree-structure to represent genetic programs [1]. The ramped half-and-half method was used for generating programs in the initial population and for the mutation operator [2]. The proportional selection mechanism and the reproduction, crossover and mutation operators [1] were used in the learning and evolutionary process.

**Terminal Set and Function Set.** In this approach, we use four simple features extracted from each data set as terminals. Given an object cutout image, the four pixel statistics, *mean*, *standard deviation*, *skewness*, and *kurtosis*, are calculated as features. In addition, we also used a constant terminal for all the three tasks.

The function set consists of the four standard arithmetic operators and a conditional operation:  $\{+, -, *, /, if\}$ . The  $+$ ,  $-$ , and  $*$  operators have their usual meanings — addition, subtraction and multiplication, while  $/$  represents “protected” division. If the first argument is negative, the *if* function returns its second argument; otherwise, it returns the third argument.

**Fitness Function.** We used classification accuracy on the training set of object images as the fitness function. The output of a genetic program in the GP system is a floating point number. In this approach, we used a variant version of the *program classification map* [6] to translate the single output value of a genetic program into a set of class labels.

**Parameters and Termination Criteria.** In this approach, the population size is 300 for the Shape date set, and 500 for the other sets. The initial maximum program sizes for the three data sets are 3, 5 and 5, and can be increased to 5, 6 and 8 during evolution. The crossover rate, mutation rate and reproduction rate used are 60%, 30% and 10%, respectively. The evolutionary process is terminated when the number of generations reaches 50, or when the classification problem has been solved on the training set or the accuracy on the validation set starts falling down, in which case the evolution was terminated earlier.

To compare the results with different brood sizes and the standard crossover operator, we use the classification accuracy, training time and the number of generations to measure the performances of these methods. For each experiment, we run 80 times and the average results are presented in the following sections.

### 3 Results

To investigate the effect of the brood size in the brood recombination crossover method, we did experiments on the three data sets using different fixed brood sizes with 2, 4, 6, 8, and 10 respectively. The average results on the *test set* of the GP system with these brood sizes together with the standard crossover operator ( $N = 1$ ) are presented in table 1.

**Table 1.** Results of brood recombination crossover with fixed different brood sizes

Brood size $N$	Shape			Coin			Texture		
	Gens	Time(s)	Accu. (%)	Gens	Time(s)	Accu. (%)	Gens	Time(s)	Accu. (%)
1	8.59	0.09	96.16	28.64	1.78	90.37	29.99	1.83	72.45
2	5.26	0.10	98.25	21.88	2.50	92.42	26.01	3.31	76.68
4	3.48	0.10	98.25	19.70	3.07	93.08	21.82	4.23	76.46
6	3.01	0.11	98.12	17.59	3.53	92.82	23.69	6.09	79.82
8	2.66	0.12	98.44	15.85	3.89	93.08	20.00	6.12	80.71
10	2.30	0.13	98.03	15.94	4.49	92.80	17.80	6.50	78.13

As shown in table 1, for all brood sizes investigated here, the brood recombination crossover method achieved better classification accuracy than the standard crossover operator for all the data sets. Although the number of generations used in the evolutionary process for the brood recombination method was smaller than the standard crossover operator, the actual training time was increased. This is mainly because the number of real evaluations in each generation in the brood recombination method was increased.

From the effectiveness point of view, this method improves the standard crossover from two-fold. Firstly, it reduces the effect of disrupting potential building blocks of the standard crossover operator through multiple trials of searching for good crossover points. Secondly, it actually adds a kind of hill-climbing search into the genetic beam search in GP.

The results also show that different brood sizes resulted in different results. For the object classification problems investigated here, it seems that a brood size of 4–8 could be a good starting point.

**Further Analysis.** Further inspection of the results reveals that as the brood size increases to a certain number (4 or 8 for different data sets), the classification accuracy is increased. When the brood size exceeds this number, however, the accuracy achieved starts falling down. These results suggest that there exists such a brood size that could lead to the best performance for a particular task. We refer to this number as *the brood-diversity point* (or range).

In the biological world, the chromosomes for a particular species are usually quite long and the crossover can occur in multiple genes in different positions. Accordingly, a huge number of crossover points can be provided, which allows a large size of brood to produce *distinguished* child chromosomes.

In most GP systems, however, the program size is limited to the parameter, *maximum program size*. In addition, the GP crossover only chooses a single point and swaps the sub trees in the parent programs. Accordingly, when the brood size increases to a certain number, the probability of the crossover operation on the same two parent programs to produce redundant programs will be extremely high. In other words, when the brood size exceeds the brood-diversity point, the brood recombination crossover operator will not only be unable to produce *distinguished* child programs, but also have to take longer time for more evaluations. This will result in a longer training time with non-improved even slightly worse performance in effectiveness due to possibility of pre-mature convergence.

## 4 Conclusions

The goal of this paper was to investigate the effect of brood size in the brood recombination crossover operator in GP for object classification problems. The goal was successfully achieved by testing five different brood sizes. The approach was examined and compared with the standard crossover operator on three object classification problems of increasing difficulty. The experiment results suggest that the brood recombination method with all the brood size investigated here achieved better classification performance than the standard crossover operator but the evolutionary training time was increased.

The results also suggest that different brood sizes usually result in different performances. As the brood size increases to the *brood-diversity point*, the system effective performance can be improved. When the brood size exceeds this point, however, the effective performance will not be improved and the system will become less efficient. Our research suggests that the brood size is related to the program size and specific tasks.

This work reveals that the brood size is closely related to the maximum program size parameter. We will investigate the relationship between the brood size and the program size together with the number of generations in the future.

## Acknowledgement

This work was supported in part by the national Marsden Fund of New Zealand and the University Research Fund (6/9) at Victoria University of Wellington.

## References

1. Koza, J.R.: Genetic programming : on the programming of computers by means of natural selection. Cambridge, Mass. : MIT Press, London, England (1992)
2. Banzhaf, W., Nordin, P., Keller, R.E., Francone, F.D.: Genetic Programming: An Introduction on the Automatic Evolution of computer programs and its Applications. Morgan Kaufmann Publishers (1998)
3. Howard, D., Roberts, S.C., Brankin, R.: Target detection in SAR imagery by genetic programming. Advances in Engineering Software **30** (1999) 303–311

4. Song, A., Ciesielski, V., Williams, H.: Texture classifiers generated by genetic programming. In: Proceedings of the 2002 Congress on Evolutionary Computation CEC2002, IEEE Press (2002) 243–248
5. Tackett, W.A.: Genetic programming for feature discovery and image discrimination. In: Proceedings of the 5th International Conference on Genetic Algorithms, Morgan Kaufmann (1993) 303–309
6. Zhang, M., Ciesielski, V., Andreae, P.: A domain independent window-approach to multiclass object detection using genetic programming. EURASIP Journal on Signal Processing, **2003**(8) (2003) 841–859
7. Tackett, W.A.: Recombination, Selection and the Genetic Construction of Computer Programs. PhD thesis, University of Southern California, Department of Electrical Engineering Systems (1994)
8. Webpage: <http://sipi.usc.edu/services/database/database.cgi?volume=textures>. (by Signal & Image Processing Institute of University of Southern California. accessed on 22 July, 2004)

# An Immune Algorithm for the Optimal Maintenance of New Consecutive-Component Systems

Y.-C. Hsieh<sup>1</sup> and P.-S. You<sup>2</sup>

<sup>1</sup> Department of Industrial Management, National Formosa University

Huwei, Yunlin 632, Taiwan

yhsieh@nfu.edu.tw

<sup>2</sup> Institute of Transportation and Logistics Engineering, National Chia-Yi University

Chia-Yi 600, Taiwan

psyuu@mail.ncyu.edu.tw

**Abstract.** There are two main objectives for this paper : (1) we will propose a more general class of consecutive-component systems which generalizes both the typical consecutive- $k$ -out-of- $n$ :F systems and two-dimensional consecutive- $k$ -out-of- $n$ :F systems, (2) we will propose an immune algorithm to investigate the optimal maintenance policy for the proposed consecutive-component systems. Numerical results are reported and compared with those of implicit enumeration.

## 1 Introduction

The  $C(k, n : F)$  system consists of  $n$  linearly connected components, and it fails if and only if there are consecutive  $k$  or more than  $k$  components failed (Fig 1).



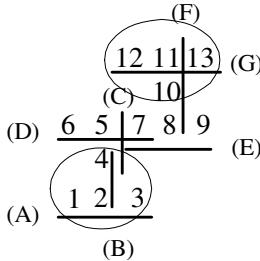
**Fig. 1.**  $C(k, n : F)$  system

The two-dimensional  $C(k, n : F)$  system consists of  $n^2$  components in a square grid of side  $n$ , and it fails if and only if there is at least one square of side  $k$  ( $2 \leq k \leq n-1$ ) that contains all failed components (Salvia and Lasher [8]). The system can be applied into various areas, e.g., safety monitoring systems, design of electronic devices, disease diagnosis, and pattern recognition etc (Hsieh and Chen [4]).

## 2 The New Proposed Consecutive-Component Systems

The proposed consecutive-component systems can be generally defined as systems with consecutive minimal cuts ([6]). For convenience, we consider the following consecutive-component system in Fig 2 as an example. This system contains 13

components, and the underlines of system of Fig 2 denote the minimal cuts of the system. That is, there are 7 minimal cuts for the system, namely  $C_A=\{1,2,3\}$ ,  $C_B=\{2,4\}$ ,  $C_C=\{4,5\}$ ,  $C_D=\{5,6,7\}$ ,  $C_E=\{7,8,9\}$ ,  $C_F=\{8,10,11\}$ , and  $C_G=\{11,12,13\}$ .



**Fig. 2.** The consecutive-component system

There are several practical applications for such a kind of systems, e.g., safety monitoring systems. In Fig 2, we may treat the 13 components as 13 cameras in a specific building. Cameras 1, 2 and 3 are responsible for lane A, and lane A is failed if and only if cameras 1, 2, and 3 are all failed. Similarly, cameras 11, 12 and 13 are responsible for lane G, and lane G is failed if and only if cameras 11, 12 and 13 are all failed. The failure reliability of the safety monitoring system of Fig 2 is defined as the probability that there is one or more of lanes are failed. Suppose that units at 1, 2, 3 and 4 of Fig 2 require the minimal operational reliability at least  $R_a$ , units at 5, 6, 7, 8 and 9 require the minimal operational reliability at least  $R_b$ , and units at 10, 11, 12 and 13 require the minimal operational reliability at least  $R_c$ .

## 2.1 The System Reliability of Consecutive-Component Systems

Suppose that  $C_1, C_2, C_3, \dots, C_k$  are minimal cuts for a specific consecutive-component system. The failure reliability of the consecutive-component system can be obtained by the disjoint subsets method. Consider the following terms.

$$P(C_1) \quad (1)$$

$$P(C_2) - P(C_2C_1) \quad (2)$$

⋮

$$P(C_k) - P(C_k(C_1 \cup C_2 \cup \dots \cup C_{k-1})) \quad (3)$$

Thus  $P(C_1 \cup C_2 \cup \dots \cup C_k)$  is the sum of terms from (1) to (3). There are several advantages of this disjoint subsets method which can be found in Hsieh [5].

## 2.2 Maintenance Policy

In the past decades, the maintenance of systems/components is always one of the main issues (Wang [9]). As known, the replacement of systems/components is the most perfect approach and there are numerous researches concerning this issue (Wang [9]). Following Flynn and Chung [1][2], we define the maintenance problems and notations for our proposed consecutive-component systems.

*Assumptions.*

1. Component  $i$  has  $m_i$  maintenance policies,  $\{0, 1, \dots, m_i - 1\}$ , where 0 denotes no maintenance,  $1 \leq i \leq n$ .
2. The reliabilities for components are known and all components in the system are independent.
3. The total cost is the long run expected average cost per period.

*Notations.*

$S$  : the component set of maintenance.

$n$  : the number of components in the system.

$p_{ij}$  : the reliability for component  $i$  under maintenance policy  $j$ ,  $1 \leq i \leq n$ ,  
 $j \in M_i = \{0, 1, \dots, m_i - 1\}$ .

$q_{ij} : q_{ij} = 1 - p_{ij}$ ,  $1 \leq i \leq n$ ,  $j \in M_i$ .

$c_{ij}$  : the maintenance cost for component  $i$  under maintenance policy  $j$ ,  $1 \leq i \leq n$ ,  
 $j \in M_i$ .

$c_0$  : the failure cost of system.

$P_F(S)$  : the failure probability under  $S$ .

The mathematical programming model for the maintenance problem is :

$$\min_{S \in \Omega} G(S) = \sum_{i \in S} \sum_{j \in M_i} q_{ij} c_{ij} + c_0 P_F(S) \quad (4)$$

$$s.t. \quad 0 \leq q_{ij} \leq 1, \quad 0 \leq P_F(S) \leq 1, \quad c_{ij} \geq 0 \quad (5)$$

Clearly, this proposed maintenance problem for consecutive-component system is a *NP hard* problem (Flynn and Chung [1][2]).

*Definition of Optimal Maintenance Policy.* (Flynn and Chung [1]、Flynn et al. [3]) Denote by  $\Omega$  the set of all subsets of  $\{1, 2, \dots, n\}$ . Results in Chung and Flynn [1][2] ensure the existence of an optimal *critical component policy* (CCP). A CCP is a stationary policy where the decisions are determined by a *critical component set*  $S \in \Omega$  and replacing component  $j$  if and only if  $j$  is failed and  $j \in S$ . The detailed definitions of critical component set and the optimal policy can be found in Chung and Flynn [1][2].

### 3 The Procedure of Immune Algorithm

The steps of proposed immune algorithm are as follows:

**Step 1.** Generate an initial population of strings (antibodies) randomly.

**Step 2.** Evaluate each individual in current population and calculate the corresponding fitness value for each individual.

**Step 3.** Select the best  $n$  individual with highest fitness values.

**Step 4.** Clone the best  $n$  individuals (antibodies) selected in Step 3. Note that the clone size for each select individual is an increasing function of the affinity with the antigen.

**Step 5.** The set of the clones in Step 4 will suffer the genetic operation process, i.e., crossover and mutation (Michalewicz [7]).

**Step 6.** Calculate the new fitness values of these new individuals (antibodies) from Step 5. Select those individuals who are superior to the individuals in the memory set, and then the superior individuals replace the inferior individuals in the memory set. While the memory set is updated, the individuals will be eliminated while their structures are too similar.

**Step 7.** Check the stopping criterion, if not stop then go to Step 2. Otherwise go to next step.

**Step 8.** Stop. The optimal or near optimal solution(s) can be obtained from the memory set.

## 4 Numerical Results and Discussions

For  $C(k, n: F)$  test problems, we let  $M_i=\{0,1,2,3\}$ ,  $p_{i0}=0$ ,  $p_{i1}=0.9$ ,  $p_{i2}=0.95$ ,  $p_{i3}=0.97$ ,  $c_{i0}=0$ ,  $c_{i1}=2$ ,  $c_{i2}=3$ ,  $c_{i3}=6$  for all  $i$  when  $k$  is not fixed. Numerical results are reported in Table 1. For consecutive-component test problems, we consider the system in Fig 2 and let  $M_i=\{0,1,2,3\}$ ,  $p_{i0}=0$ ,  $p_{i1}=0.9$ ,  $p_{i2}=0.97$ ,  $p_{i3}=0.99$ ,  $c_{i0}=0$ ,  $c_{i1}=2$ ,  $c_{i2}=4$ ,  $c_{i3}=15$  for all  $i$ , and  $R_a=0.98$ ,  $R_b=0.95$ ,  $R_c=0.98$ . Numerical results of various  $c_0$  are reported in Table 2. All results are computed by Pentium IV 2.8 GHz PC and programs are coded by MATLAB 6.5.

From Table 1 to Table 2, we have:

- (1) The CPU time is exponentially increasing with the increase of  $n$  for implicit enumeration. However, it is not so drastic for the proposed immune algorithm.
- (2) For all most of test problems, the proposed immune algorithm can obtain the optimal solutions which are the same as those by implicit enumeration.
- (3) Table 2 shows the top-5 multiple optimal solutions for consecutive-component test problems. Clearly, it trends to use high levels of maintenance with the increase of failure cost  $c_0$ .

**Table 1.** Numerical results of  $C(k, n: F)$  problems ( $k$  is not fixed)

$n$	Min Cuts	Immune Algorithm			CPU (sec) A	Implicit Enumeration			CPU (sec) B	A/B	No. Comb.
		$G(S)$	$R$	Maintenance policy		$G(S)$	$R$	Maintenance policy			
4	MC <sub>1</sub>	0.6262	0.9971	0232	0.828	0.6262	0.9971	0232	0.688	1.203488	256
8	MC <sub>2</sub>	1.2976	0.9944	20330332	1.125	1.2976	0.9944	02330332	204.656	0.005497	65536
12	MC <sub>3</sub>	1.7455	0.9933	033303303302	6.562	1.7445	0.9927	023303303302	75651.6	8.67E-06	16777216

No. Comb. = Number of combinations of all maintenance policies. MC<sub>1</sub>= $\{1,2,3\}, \{3,4\}$ .

MC<sub>2</sub>= $\{1,2,3\}, \{3,4\}, \{4,5,6\}, \{6,7\}, \{7,8\}$ . MC<sub>3</sub>= $\{1,2,3\}, \{3,4\}, \{4,5,6\}, \{6,7\}, \{7,8,9\}, \{9,10\}, \{10,11,12\}$ .

**Table 2.** Numerical results of consecutive-component problems with various  $c_0$ 

$c_0$	Immune Algorithm			CPU (sec)	$R_a$	$R_b$	$R_c$
	$G(S)$	$R$	Maintenance policy				
50	1.1735	0.9981	2303303200302 0323303200320 2303303200302	112.31	0.9991 0.9991 0.9993	0.9991 0.9991 0.9988	0.9990 0.9990 0.9994
30	1.1295	0.9973	2302302200320 2302203200320 2302203200302	111.64	0.9991 0.9985 0.9985	0.9982 0.9982 0.9982	0.9994 0.9994 0.9994
10	0.9940	0.9606	0300300300030 3003003000300 0303003000300	107.28	0.9801 0.9801 0.9801	0.9801 0.9703 0.9703	0.9801 0.9900 0.9900

(memory set=5, mutation=0.85, crossover=0.9, generation=5000, affinity =0.1)

## 5 Conclusions

In this paper, (i) we have proposed a more general class of consecutive-component systems, (ii) we have proposed a new immune algorithm to investigate the optimal maintenance policy for the proposed systems. Numerical results have shown the superior performance for all test problems. It has to be emphasized that the proposed immune algorithm can obtain multiple optimal solutions for all test problems and can provide the decision makers alternative choices.

## References

1. Flynn, J., Chung, C.S.: A heuristic algorithm for determining replacement policies in consecutive k-out-of-n systems. Computers and Operations Research 31 (2004) 1335-1348.
2. Flynn, J, Chung, C.S.: A branch and bound algorithm for computing optimal replacement policies in consecutive  $k$ -out-of- $n$  systems. Naval Research Logistics 49 (2002) 288-302.
3. Flynn, J, Chung, C.S., Chiang, D.: Replacement policies for a multicomponent reliability system. Operations Research Letters 7 (1988) 167-172.
4. Hsieh, Y.C., Chen, T.C.: Reliability lower bounds for two-dimensional consecutive- $k$ -out-of- $n$ : F systems. Computers and Operations Research 31 (2004) 1259-1272.
5. Hsieh, Y.C.: Alternative approach for coherent system reliability with minimal cuts/paths. Working paper (2006).
6. Hsieh, Y.C., Chen, T.C.: The reliability of systems with stair-type consecutive minimal cuts. Working paper, submitted. (2006).
7. Michalewicz, Z.: Genetic algorithm + Data structures = Evolution programs, Springer-Verlag Berlin Heidelberg, New York (1994).
8. Salvia, A.A., Lasher, W.C.: 2-dimensional consecutive- $k$ -out-of- $n$  : F models. IEEE Transactions on Reliability 39 (1990) 382-385.
9. Wang, H.: A survey of maintenance policies of deteriorating systems. European Journal of Operational Research 139 (2002) 469-89.

# Immune Genetic Algorithm and Its Application in Optimal Design of Intelligent AC Contactors\*

Li-an Chen<sup>1</sup> and Peiming Zhang<sup>2</sup>

<sup>1</sup> Department of Electronic Engineering, Xiamen University of Technology, P.R.C

<sup>2</sup> Department of Electrical Engineering, Fuzhou University, P.R.C

chenla@vip.163.com, zhangpm2002@163.com

**Abstract.** An application of Immune Genetic Algorithm (IGA) suitable for the optimal design to intelligent AC contactors is presented in this paper. Besides the ability of stochastic global searching of Simple Genetic Algorithm (SGA), the IGA draws into the mechanisms exist in biological immune system such as immune memory, immune regulation, antibody diversity and others. The simulation results show that IGA overcomes the disadvantages of premature convergence of SGA, and improve the global searching efficiency and capability. This algorithm has been successfully used in the optimal design to the intelligent AC contactors.

**Keywords:** Evolutionary computing, optimal design, contactor.

## 1 Introduction

With the rapid development of the technology of artificial intelligence, several stochastic methods originated from the biological evolutionary theories such as Evolutionary Algorithm, Genetic Algorithm (GA) are widely used in the area of optimal design. Differed from the deterministic methods, the stochastic methods have more opportunities to converge the global optimum. Especially, GA has been applied for solving the design and optimization problem of electromagnetic devices. However, the Simple Genetic Algorithm (SGA) has its limitation, for example, it's difficult to overcome the premature convergence, another problem for SGA is the low-search efficiency.

To overcome these drawbacks of SGA, people are trying to improve it and have proposed some modified methods. The Immune Genetic Algorithm (IGA) is a kind of modified genetic algorithm which is based on the theory of biological immune system. The biological immune system is a parallel, distributed, self-organizing and highly adaptive complex system which has special characteristics such as immune recognition, immune memory, immune regulation, and so on<sup>[1][2]</sup>. IGA not only retains the ability of stochastic global searching of SGA, but also introduces the selection mechanism based on antibodies' concentration and the diversity maintaining mechanism which exists in nature immune system. The IGA has better global convergence and very strong self-adaptive ability. In this paper, IGA is applied to the optimal design of the intelligent AC contactors.

\* This program is supported by the Bureau of Science and Technology in Fujian Province and Xiamen city (the project number is 2005F004 and 3502Z20041072 respectively).

## 2 Background knowledge of IGA

GA is a kind of evolutionary algorithm which imitates the natural evolution. The initial population is produced randomly, and the new population is reproduced by three basic genetic operations: selection, crossover, and mutation. The individual in the population is evaluated by setting a proper fitness function, the individual with higher fitness has the higher probability to be selected and to be reserved to the next generation, vice versa. As GA is a kind of stochastic search method, it has few limitations to the design condition, so it's an ideal algorithm to the optimal design problems. But there are some drawbacks existing in GA, it's necessary to improve it.

IGA is an improved genetic algorithm base on the nature immune system. The objective functions of the problem are taken as antigens while the solutions to the problem are taken as antibodies. According to the principle of nature immune system, the biological immune system is able to produce corresponding antibodies to resist the invading of antigens. This process is called immune response. Part of antibodies will be maintained in memory cell. When the same kind of antigen invades again, the memory cell will be stimulated and produce a large number of antibodies, the secondary immune response will be faster and stronger than the primary one. It demonstrates the function of memory of immune system. At the same time, antibodies are encouraged and restrained each other in order to keep population diversity and immune balance. This is self-regulation mechanism of immune system.

Corresponding to the functions and mechanisms of biological immune system, IGA has salient characteristics comparing with SGA.

Assume that the population size is N, the gene length is M. Some terms are defined as follow.

①Diversity: The immune system which is composed by antibodies is an uncertain system in the process of evolution, its diversity can be expressed by Shannon's entropy as

$$H(N) = \frac{1}{M} \sum_{j=1}^M H_j(N) \quad (1)$$

Where  $H_j(N)$  is the entropy of the  $j$ th gene.

②Similarity: Similarity  $A_{ij}$  is the similar level of antibody  $i$  and  $j$ .

$$A_{ij} = \frac{1}{1 + H(2)} \quad (2)$$

Where  $H(2)$  is entropy, it can be calculated by (1) (let  $N=2$ ).

③Antibodies' concentration: Antibodies' concentration presents the weight of the similar antibodies in the population.

$$C_i = \frac{\text{The number of antibodies whos similarity with antibody } i \text{ is greater than } \lambda}{N} \quad (3)$$

Where  $\lambda$  is the constant of similarity. Generally,  $0.9 \leq \lambda \leq 1$ .

④Integrated fitness: The integrated fitness which synthesizes fitness and concentration is given by

$$\text{fitness}' = \text{fitness} * \exp(k * C_i) \quad (4)$$

Actually, integrated fitness  $\text{fitness}'$  is the adjustment to  $\text{fitness}^{[3]}$ . As to the maximum problem,  $k$  is negative. In this paper,  $k$  is chosen as -0.8 by tests. In the operation of selection, the selection probability of the antibodies is proportional to the integrated fitness, so the higher the antibodies' concentration is, the smaller the integrated fitness is, and the smaller the selection probability of the antibodies would be. This is a new strategy to keep the diversity of populations.

Key steps of IGA is as follow: ① Initializing population; ② Calculating fitness; ③ Producing new antibodies; ④ Reproducing population based on concentration; ⑤ Refreshing memory; ⑥ Terminating calculation.

### 3 Application of IGA on Optimal Design to Intelligent AC Contactor

#### 3.1 Introduction to Intelligent AC Contactor

The intelligent AC contactor has a special control principle, which introduces high integrated single-chip microcomputer to on-time control AC contactor's starting, maintaining and breaking operations. In starting process, the intelligent AC contactor is activated at an optimal voltage phase angle which let coil have minimum starting power energy. After a while the power is shut down by the control circuit and then restore it in order to obtain an optimal dynamic behavior. In maintaining and breaking process, it can be operated on the condition of no-noise, no-arc or less-arc.

The principle of this kind of intelligent AC contactor can be referred to [4].

#### 3.2 Optimal Design to Intelligent AC Contactor

The target of optimal design to intelligent AC contactor is to obtain a set of parameters concerning structure and control under which the overall technical and economic index would be best.

##### ① Optimal Variables

Six parameters( $a_z, N, d, \varphi, t_s, \Delta t$ ) which greatly influence the dynamic characteristics of intelligent AC contactor are chosen as the optimal variables. Where  $a_z$ ,  $N$ ,  $d$  are structure parameters which present the width of U-shape iron, the number of coil-turns and the wire diameter of coil respectively.  $\varphi$ ,  $t_s$ ,  $\Delta t$  are control parameters,  $\varphi$  presents the inception phase angle;  $t_s$  presents the time the power is shut down, and  $\Delta t$  presents the duration of power shutting down.

##### ② Objective Functions

Three targets( $V(X), P_t(X), E_k(X)$ ) which most demonstrate the overall economic and technique demand are chosen as the multi-objective functions including volume, power loss and kinetic energy of armature. IGA is used as the optimization algorithm to find the optimal variables of the objective function. Where  $V(X)$  presents the volume of electromagnetic structure which is composed by the core volume  $V_{Fe}$  and the coil effective volume  $V_{cu}$ ;  $P_t(X)$  presents the power loss of coil during the contactor's starting process;  $E_k(X)$  presents the kinetic energy on unit area when armature bumps against stationary core.

### ③Fitness Function

The multi-objective functions can be converted into single-objective fitness function by adding them with weight, meanwhile the minimum problem is changed into maximum problem

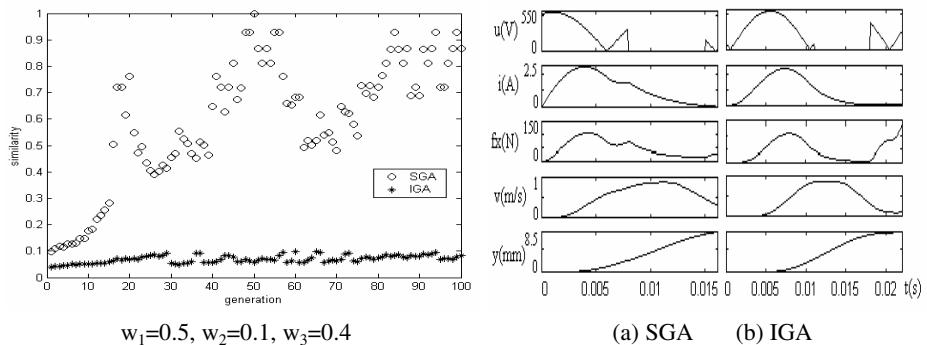
$$F(X) = F_{\max} - \{w_1[\alpha V_{Fe}(X)/V_{Fe0} + \beta V_{cu}(X)/V_{cu0}] + w_2 P_i(X)/P_{i0} + w_3 E_k(X)/E_{k0}\} \quad (5)$$

Where  $F_{\max}$  presents a constant which is big enough to guarantee  $F(x)>0$ ;  $V_{Fe0}, V_{cu0}$  present the pre-optimization values of core volume and the effective coil volume;  $P_{i0}, E_{k0}$  present the pre-optimization values of the power loss during the starting process and the kinetic energy on unit area;  $\alpha, \beta, w_1, w_2, w_3$  present weight factors.

### 3.3 Example and Analysis

The population size is assumed as 50, the probability of crossover is 0.95 and the probability of mutation is 0.08. Weight factors  $\alpha, \beta$  are 0.4 and 0.6 respectively. The calculation results are shown in Fig.1, Fig.2 and Table 1.

Fig.1 shows the diversity compare of SGA and IGA. It's obvious that IGA can effectively keep the diversity of evolutionary populations with the increase of evolutionary generations, but SGA can't. As a result, SGA is prone to be trapped in a local optimum while IGA is capable to alleviate the problem associated with premature convergence benefiting from its diversity-keeping feature. Fig.2 is the dynamic curves of the intelligent AC contactor. The dynamic curves include voltage, exciting current, attractive force, velocity and displacement characteristics.



**Fig. 1.** Similarity of population

**Fig. 2.** Dynamic curves ( $w_1=0.5, w_2=0.1, w_3=0.4$ )

Table 1 shows the results of design optimization in different weight  $w_1, w_2$  and  $w_3$ . Both IGA and SGA have higher integrated technical and economic target. The most obvious effect is the decreasing of kinetic energy  $E_k$ . The reason is that the intelligent AC contactor has a sound starting process by using of optimal control parameters, the armature is stick to stationary core at a near zero velocity(it can be seen in Fig.2). So  $E_k$  is enormously reduced and the mechanical span is prolonged greatly. The effect of material saving is also obvious especially for the copper consumption  $V_{cu}$ , it leads decreasing of

the cost of the device. As for the feature target of power loss  $P_t$ , the result is intercross in certain circumstances. When the weight of  $P_t$  ( $w_2$ ) is large enough (for example,  $w_2=0.3$ ), the optimal result of  $P_t$  is smaller than the pre-optimization one, thus power is saving in the starting process. However the other feature targets including  $V_{cu}$  and  $E_k$  would not be as smaller as that in the circumstance which  $w_2$  is smaller. So the characteristics of  $V_{cu}$  and  $E_k$  are as a sacrifice for power saving. To compare the optimal results of IGA and SGA, the power loss of IGA is slightly greater than that of SGA in certain circumstances, but the copper consumption and the kinetic energy of IGA are smaller than that of SGA, the integrated technical and economic target of IGA is higher than that of SGA. In other words, IGA is capable to search better optimal results than SGA, meanwhile, in the evolutionary process, IGA is able to search optimal results at earlier generation than SGA. In conclusion, the IGA has higher search ability and efficiency than SGA in optimal design of intelligent AC contactor.

**Table 1.** Results of design optimization

Items	Names	Pre-Optimization	$w_1=0.5$			$w_1=0.5$			$w_1=0.5$		
			$w_2=0.1$	$w_3=0.4$	SGA	IGA	SGA	IGA	SGA	IGA	$w_2=0.3$
Optimal Variables	$a_z$ (cm)	3.2	2.4	2.4	2.4	2.4	2.4	2.4	2.4	2.4	2.4
	$N$ (turn)	2820	2620	2820	2820	2720	2520	2720	2520	2720	2720
	$d$ (mm)	0.31	0.23	0.19	0.23	0.23	0.27	0.25	0.27	0.25	0.25
	$\varphi(^\circ)$	0-180	69	171	147	69	90	84	90	84	84
	$t_s$ (ms)	--	7.0	11.0	8.0	7.0	6.0	7.0	6.0	7.0	7.0
	$\Delta t$ (ms)	--	7.0	7.0	11.0	7.0	7.0	7.0	7.0	7.0	6.0
Feature Targets	$V_{Fe}$	95.82	71.87	71.87	71.87	71.87	71.87	71.87	71.87	71.87	71.87
	$V_{cu}$	31.30	12.40	8.73	13.51	12.95	17.28	15.75			
	$P_t$ (W-s)	3.70	4.06	5.49	3.94	3.84	2.93	3.15			
	$E_k$	2657.1	25.06	3.94	28.44	22.12	256.0	145.8			

## 4 Conclusions

A kind of improved genetic algorithm based on immune principle is introduced to the area of optimal design on intelligent AC contactors in this paper. Enlightened by the immune principle, IGA has several distinguished features, such as keeping the diversity of evolutionary populations, alleviating premature convergence and enhancing the search efficiency, etc. The simulation results demonstrate that IGA performs better at the aspects of search speed and search ability than SGA in optimal design of intelligent AC contactor.

## References

1. Xufa Wang, Xianjun Zhang, Xianbin Cao, jun Zhang, and Lei Feng, "An Improved Genetic Algorithm Based on Immune Principle," Mini-Micro System, vol.20, pp. 117-120, Feb.1999.
2. Jang-Sung Chun, Min-Kyu Kim, Hyun-Kyo Jung, "Shape optimization of electromagnetic devices using immune algorithm," IEEE Trans. on Magnetics, Vol.33, No.2, 1997 : 1876-1879.

3. Xunxue Cui, "The Study of Evolutionary Algorithms Based on Multiobjective Optimizaiton," Ph.D. dissertation, University of Science and Technology of China , 2001.
4. Zhihong Xu, and Peiming Zhang, "Research on Intelligent AC Contactor," Low-voltage Electrical Apparatus, vol.26, pp.19-21,53,1998.

# The Parametric Design Based on Organizational Evolutionary Algorithm

Cao Chunhong<sup>1</sup>, Zhang Bin<sup>1</sup>, Wang Limin<sup>2</sup>, and Li Wenhui<sup>2</sup>

<sup>1</sup> College of Information Science and Engineering, Northeastern University,  
Shenyang 110004, P.R. China

Chunhongcao\_li@163.com, zhangbin@neu.edu.cn

<sup>2</sup> College of Computer Science and Technology, Jilin University,  
Changchun 130012, P.R. China

Jeffreywlm@sina.com, liwh@public.cc.jl.cn

**Abstract.** Geometric constraint problem is equivalent to the problem of solving a set of nonlinear equations substantially. In this paper we propose a new optimization algorithm—organizational evolutionary algorithm (OEA) and apply it into the geometric constraint solving. In OEA the colony is composed of the organizations. Three organizational evolutionary operators—split operator, merging operator and coordinating operator can lead the colony to evolve. These three kinds of operators have different functions in the algorithm. Split operator limits the scale of the organization, and makes sure a part of organization come into next generation directly, which maintains the variety of the generation. Merging operator makes use of the leader's information fully and acts as a local searching function. Cooperating operator increases the degree of adaptability between the two organizations by the interactions. The experiment shows that OEA has good capability in the geometric constraint solving.

**Keyword:** parametric design, geometric constraint solving, organizational evolutionary algorithm, split operator, merging operator, coordinating operator.

## 1 Introduction

The parametric design is a geometric constraint-solving problem. Geometric constraint solving approaches are made of three approaches: algebraic-based solving approach, based solving approach and graph-based solving approach. One constraint describes a relation that should be satisfied. Once a user defines a series of relations, the system will satisfy the constraints by selecting proper state after the parameters are modified. The idea is named model-based constraints. Constraint solver is a segment for the system to solve the constraints.

Evolutionary Computation is a kind of random research algorithm simulating natural evolution mechanism. It can solve complex and morbid problems and has been applied in the numerical optimization, combination optimization, machine study and neural networks. The problem puzzling the evolutionary algorithm is that the algorithm may get into the local best solution. In order to overcome the shortages, there are many algorithms to be introduced, such as orthogonal genetic algorithm<sup>[1]</sup>,

microgenetic algorithm<sup>[2]</sup>, the immune programming<sup>[3]</sup>, the genetic algorithm by coordinating exploration and exploitation<sup>[4]</sup> and good point set based genetic algorithm<sup>[5]</sup>.

In 1995, Wilcox firstly introduced the concept “organization” in the economics into the classifier based on the genetic algorithm<sup>[6]</sup>. In reference<sup>[7]</sup>, the author proposed a novel method from bottom to top—organizational coevolutionary algorithm and has achieved good effect. In this paper we apply the organizational evolution into the geometric constraint solving and propose a new optimization algorithm—organizational evolutionary algorithm.

## 2 Organization Evolution Algorithms

**Split Operator:** The condition of splitting the organization is as follows:

$$(|org| > Max_{os}) or \{(|org| \leq Max_{os}) and (U(0,1) < \frac{|org|}{N_0})\} \quad (1)$$

**Merging Operator:** Supposing the leader of  $org_p$  is  $(x_1, x_2, \dots, x_n)$ , new individual is  $r_j = (r_{j,1}, r_{j,2}, \dots, r_{j,n})$ ,  $j=1, 2, \dots, N$ , then in the merging strategy 1,  $r_j$  can be gotten from formula(2):

$$r_{j,k} = \begin{cases} \underline{x}_k, z_{j,k} < \underline{x}_k \\ \bar{x}_k, z_{j,k} > \bar{x}_k, k = 1, 2, \dots, n \\ z_{j,k}, \text{otherwise} \end{cases} \quad (2)$$

here,  $z_{j,k} = x_k + U_k (0, 1) \times (x_k - y_{jk})$ .

In the merging strategy 2,  $r_j$  can be gotten from formula (3):

$$r_{j,k} = \begin{cases} \underline{x}_k + \beta \times (\bar{x}_k - \underline{x}_k), U_k(0,1) < \frac{1}{n} \\ x_k, \text{otherwise} \end{cases} \quad (3)$$

$k=1,2,\dots,n$ , here  $\beta = U(0,1)$  is different to every  $x_k$ .

Before get the computing result of  $r_j$ ,  $z_{j+M}$  can be gotten form formula (4).

$$z_{j+M} = \begin{cases} r_j, Fitness(r_j) \geq Fitness(y_j) \\ r_j, (Fitness(r_j) < Fitness(y_j)) \text{and} \\ \{U_j(0,1) < \exp(Fitness(r_j) - Fitness(y_j))\} \\ y_i, \text{otherwise} \end{cases} \quad (4)$$

**Cooperative operator:** Supposing two parent organizations are  $org_{p_1} = \{x_1, x_2, \dots, x_M\}$  and  $org_{p_2} = \{y_1, y_2, \dots, y_N\}$ . If  $U(0, 1) < CS$ , then the two child organizations can be got by the cooperative strategy 2, here  $CS \in (0, 1)$  is the parameter preestablished.

In cooperative strategy 1, q and r can be gotten from the formula (5), here  $\beta_k = U_k(0, 1)$  :

$$\begin{cases} q_k = \beta_k \times x_k + (1 - \beta_k) \times y_k \\ r_k = (1 - \beta_k) \times x_k + \beta_k \times y_k \end{cases} \quad (5)$$

$k=1, 2, \dots, n$ . In the cooperative strategy 2, if can be gotten from the formula (6), here  $1 < i_1 < n$ ,  $1 < i_2 < n$ , and  $i_1 < i_2$  :

$$\begin{cases} q = (x_1, x_2, \dots, x_{i_1-1}, y_{i_1}, \dots, y_{i_2}, x_{i_2+1}, x_{i_2+2}, \dots, x_n) \\ r = (y_1, y_2, \dots, y_{i_1-1}, x_{i_1}, \dots, x_{i_2}, y_{i_2+1}, y_{i_2+2}, \dots, y_n) \end{cases} \quad (6)$$

### 3 The Accomplishment of Organizational Evolutionary Algorithm in the Geometric Constraint Solving

The constraint problem can be formalized  $a \leftarrow s(E, C)^{[8]}$ , here  $E = (e_1, e_2, \dots, e_n)$ , it can express geometric elements, such as point, line, circle, etc;  $C = (c_1, c_2, \dots, c_m)$ ,  $c_i$  is the constraint set in these geometric elements. Usually one constraint is represented by an algebraic equation, so the constraint can be expressed as follows:

$$\begin{cases} f_1(x_0, x_1, x_2, \dots, x_n) = 0 \\ \dots \\ f_m(x_0, x_1, x_2, \dots, x_n) = 0 \end{cases} \quad (7)$$

$X = (x_0, x_1, \dots, x_n)$ ,  $X_i$  are some parameters, for example, planar point can be expressed as  $(x_1, x_2)$ . Constraint solving is to get a solution  $x$  to satisfy formula (7).

$$F(X_j) = \sum_1^m |f_i| \quad (8)$$

Apparently if  $X_j$  can satisfy  $F(X_j) = 0$ , then  $X_j$  can satisfy formula (7). So the constraint problem can be transformed to an optimization problem and we only need to solve  $\min(F(X_j)) < \mathcal{E}$ .  $\mathcal{E}$  is a threshold. Based on the definitions above all, the whole program of the Organizational Evolution Algorithm can be described as follow:

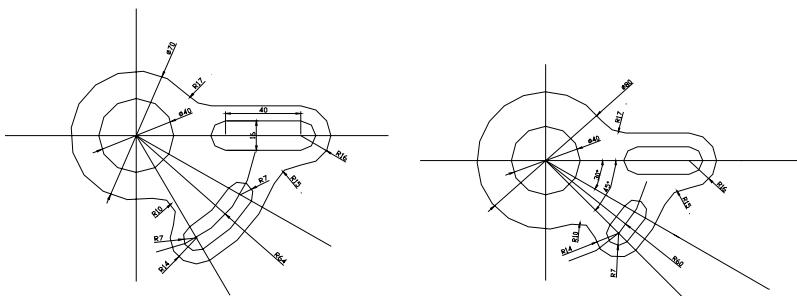
- (1) Initialize the generation  $P_0$  including  $N_0$  organizations, and each organization has one number;  $t \leftarrow 0$ ;
- (2) If the stop conditions are satisfied, output the solution and end, otherwise, turn to step 3.
- (3) To each organization of the generation  $P_t$ , if it can satisfy the split condition (6), then enforce the split operator.
- (4) If the number of the organization in  $P_t$  is bigger than 1, turn to step 5, otherwise turn to step 6.
- (5) Choose two father generation's organizations  $org_{p_1}$  and  $org_{p_2}$  randomly, and then choose one split operator or one cooperative operator randomly acting it on the two father organizations, and turn step 4.
- (6) Joined the remnant organizations of  $P_t$  into  $P_{t+1}$ , set  $t \leftarrow t+1$ , turn to step 2.

## 4 Application Instance and Result Analysis

We can realize from the above figures that once a user defines a series of relations, the system will satisfy the constraints by selecting proper state after the parameters are modified.

We test the capability of OEA and GA in geometric constraint solving form four problems that we choose  $p_1$ ,  $p_2$ ,  $p_3$  and  $p_4$ . Here  $N_0=150$ ,  $\text{Max}_{\text{OS}}=20$ ,  $\text{AS}=0.8$ ,  $\text{CS}=0.6$ . In order to describe the computation complication degree more exactly, we present average evaluation times and average running time. The unit is second; the run circumstance is a computer of PIII-667 and 128M EMS Memory.

In a word the experiment result can show OEA has a good capability in geometric constraint solving. One hand is that it can find the solution of high capability. On the other hand, the evaluation time that it needs to is less; the time complication degree is lower. Moreover, the solution deviation of OEA is smaller and this can indicate that the performance of OEA is very steady and has strong robust capability.



**Fig. 1.** (a) A design instance (b) Solving result

## 6 Conclusion

In this paper we propose a new method using the geometric constraint solving—Organizational Evolutionary Algorithm. The comparison result shows that OEA has a

strong capability of searching the best solution and it is superior to the other algorithms not only in the quality of the solution but also in the computation complication degree. The experiment also shows that it is good in large-scale colony and the efficient searching mechanism of Organizational Evolutionary Algorithm itself. The large colony is not easy to get into the local best solution and the searching mechanism of Organizational Evolutionary Algorithm makes it have a relatively fast convergence speed.

## References

- [1] Leung Y.W., Wang Y.. An orthogonal genetic algorithm with quantization for global numerical optimization. *IEEE Transactions on Evolutionary Computation*, 2001, 5(1): 41~53
- [2] Kazarlis S.A., Papadakis S.E., Theocharis J.B., Petridis V., Microgenetic algorithm as generalized hill-climbing operators for GA optimization, *IEEE Transaction on Evolutionary Computation*, 2001, 5(3): 204-217
- [3] Wang Lei, Pan Jin, Jiao Li-Cheng. The immune programming. *Chinese Journal of Computer*, 2000, 23(8): 806~812
- [4] Jiang Rui, Luo Yu-Pin, Hu Dong-Cheng, Szeto Kwok-Yip. A genetic algorithm by coordinating exploration and exploitation—convergence properties and performance analysis. *Chinese Journal of Computers*, 2001, 24(12): 1233~1241
- [5] Zhang Ling, Zhang Bo. Good point set based genetic algorithm. *Chinese Journal of Computers*, 2001, 24(9): 917~922
- [6] Wilcox J.R.. Organizational learning within a learning classifier system [Master dissertation]. University of Illinois, Illinois, USA, 1995
- [7] Liu Jing, Zhong Wei-Cai, Liu Fang, Jiao Li-Cheng. Classification based on organizational coevolutionary algorithm. *Chinese Journal of Computers*, 2003, 26(40): 446~453
- [8] Liu Shengli, Tang Min, Dong Jinxiang. Geometric Constraint Satisfaction Using Genetic Simulated Annealing Algorithm. *Journal of Image and Graphics*, Vol8, No.8, pp.938-945, 2003

# Buying and Selling with Insurance in Open Multi-agent Marketplace

Yuk-Hei Lam<sup>1</sup>, Zili Zhang<sup>1,2</sup>, and Kok-Leong Ong<sup>1</sup>

<sup>1</sup> School of Engineering and Information Technology  
Deakin University

Waurn Ponds, Victoria 3217, Australia  
`{yuk, zhang, leong}@deakin.edu.au`

<sup>2</sup> Faculty of Computer and Information Science  
Southwest University  
Chongqing 400715, China

**Abstract.** In this paper, we incorporate the insurance concept in buying and selling model for agents to trade in the open multi-agent marketplace. During buying, agents purchase insurance as a method to search for potential sellers and select their partners based on the information provided by insurance agents. During selling, agents purchase insurance as a method to protect themselves against potential risk. The insurance concept greatly simplifies the trading procedure in the open marketplace. The novelty of this proposal is that it ensures a dynamic trading environment while agents continue to seek maximum utility and being fully protected by insurance. Our experimental results confirm the feasibility of our approach.

## 1 Introduction

Electronic Commerce (EC) represents the future of commerce. Nowadays, its rapid growth enables businesses to expand the marketplace not only nationally but also internationally. Trading in the online marketplace, therefore, becomes a part of the important daily business activities. However, conducting businesses online also generate a series of issues, one of which is trust. There is also no exception in the open multi-agent marketplace. In an environment where full of uncertainty, we believe that it is crucial for agents to equip some forms of protection mechanisms while performing any forms of online business transactions (both before and after the completion of transaction).

Online auctions websites such as Amazon.com, eBay and Yahoo!Auctions have provided a service, known as the protection program, so as to gain the confidence and trust of buyers. Even though online auctions websites can compensate buyers either financially or by replacing the item, it is complicated to file complaints as there are different criteria to meet.

Most importantly, the existing protection programs only support buying and paying online. In other words, only buyers are protected in engaging online transactions, whereas sellers have no protection provided by the auctions websites.

We argue that sellers also face risk in transactions, and therefore, need protection against deceitful behaviors (i.e., badmouthing<sup>1</sup>) and uncertainty (i.e., item lost in postage).

In order to solve the above problems, we propose an integral model, which apply insurance concept [2] in buying and selling in the open marketplace. The advantage of applying insurance in this context is that it simplifies the buying and selling procedures as the insurance agents are aimed to evaluate the trustworthiness of agents in the marketplace (not the buying agents or selling agents themselves). Moreover, our insurance agents act as the third trusted parties who propagate the collected global reputation ratings of agents to the insured agents. In this sense, the insured agents can easily obtain the same global reputation rating of others. Our model is flexible in terms of helping agents search for suitable buyers or sellers and making decision on who to trade with under different situations, which can vary from safe to risky situations.

The remaining sections of this paper are organized as follows: The buying model and selling model will be discussed in Section 2 and Section 3 respectively. In Section 4, we present the simulation results. We then give a review of related works in Section 5, and the conclusions are laid out in Section 6.

## 2 Buying Model with Insurance

Without loss of generality, we assume both individuals (i.e., Eric) and enterprises (i.e., E1,...,E10) deploy software agents for trading in the marketplace. Let  $\mathcal{A}_1, \dots, \mathcal{A}_m$  be the set of agents who perform trading in the marketplace. An agent  $\mathcal{A}_i$  contextualized to buying a good  $g$  is denoted as  $B(\mathcal{A}_i, g)$  and an agent  $\mathcal{A}_j$  which has been contextualized to selling a good  $g$  is denoted as  $S(\mathcal{A}_j, g)$ . We further define that  $\mathcal{I}_1, \dots, \mathcal{I}_p$  be a set of insurance agents who provide insurance services in the system.

Traditionally, the buying agents require to visit all possible websites, which offer the good  $g$ , so as to collect information for decision making. We consider that there are two methods available for the buying agents to search for potential sellers. One is to publicly announce the requested good as presented in [3]. The other way is to use insurance services. In this paper, we apply insurance concept as a tool for trading in the open marketplace. Hence, the trading procedure can be greatly simplified and manageable.

### 2.1 Calculating the Net Value and the Premium

When the buying agent  $B(\mathcal{A}_i)$  decided to purchase insurance [2] for good  $g$ , it then announces the maximum pay amount,  $MC(g)$ , to the insurance agent  $\mathcal{I}_k$  in the marketplace. In order to be insured, the buying agent  $B(\mathcal{A}_i)$  makes commitment with the insurance agent  $\mathcal{I}_k$  that it agrees to pay the cost of premium,  $\alpha_1$ , to the insurance agent,  $\mathcal{I}_k$ .

---

<sup>1</sup> Schillo et. al. [1] have pointed out that there is motivation for users to lie in recommendation stage.

**Definition 1.** Let  $IB = \{\text{offer}(S(\mathcal{A}_j), g), \dots, \text{offer}(S(\mathcal{A}_k), g)\}$  be the set of interested selling agents who submit offer price,  $\text{offer}(\cdot, g)$ , to the buying agent,  $B(\mathcal{A}_i)$ , for good  $g$ . Given that the insurance agents are facing different level of risk as the buying agents have options to deal with the selling agents with different global reputation ratings,  $\mathcal{R}^G(\cdot)$ . Towards this, we adjust the cost of premium towards the buying agent as

$$\alpha_1 = \frac{\text{offer}(S(\mathcal{A}_j), g)^{\mathcal{R}^G(S(\mathcal{A}_j))}}{\mathcal{R}^G(S(\mathcal{A}_j))} \quad (1)$$

In the above, the cost of premium,  $\alpha_1$ , is adjusted based on the value of global reputation rating of the potential selling agent  $S(\mathcal{A}_j)$ . We can observe that the buying agents will pay less for premium if they choose to deal with selling agents, which have good reputation ratings. In this case, an untrustworthy selling agent will have less chance to make the deal with the buying agents. Second, the cost of premium gives an insight of how secure the transaction based on the evaluation of insurance agents.

Once the buying agent  $B(\mathcal{A}_i)$  collected the offers from all interested selling agents, it calculates the net value of the good  $g$  for each individual. In the buying agent  $B(\mathcal{A}_i)$ 's perspectives, the net value,  $NV(S(\mathcal{A}_j, g))$ , of good  $g$  provided by the selling agent  $S(\mathcal{A}_j)$  is denoted as

$$NV(S(\mathcal{A}_j, g)) = MC(g) - \text{offer}(S(\mathcal{A}_j), g) - \alpha_1 \quad (2)$$

The net value is used for agents to find out the difference between each individual offer price. The higher value of net value, the better bargain the buying agent can get.

## 2.2 Deciding the Winner and Evaluating the Performance

To decide which selling agent wins the deal, the buying agents are required to select the selling agent with the highest net value among all interested selling agents, i.e.,

$$\{p \in IB \mid \forall g \in IB, p \neq g, NV(p) > NV(g)\} \quad (3)$$

After the transaction is completed, the buying agent  $B(\mathcal{A}_i)$  will evaluate the trustworthiness of selling agent  $S(\mathcal{A}_j)$  in that transaction based on its satisfaction level. To do so, the buying agent  $B(\mathcal{A}_i)$  submits the global reputation of the selling agent  $S(\mathcal{A}_j)$  to the insurance agent,  $\mathcal{I}_k$ .

## 3 Selling Model with Insurance

Suppose that the selling agents are willing to purchase insurance as a way to insure themselves from incidents. Commonly, the selling agents usually would not have financially lost if an incident occurs. Instead, the damaged of reputation rating is the crucial factor they concern the most. Therefore, it is also important for selling agents to purchase insurance as a way to against badmouthing and

uncertainty. Moreover, the selling agents can also use insurance services as a method to search for potential buying agents.

Consider that the selling agent  $S(\mathcal{A}_j)$  engages a transaction with the buying agent  $B(\mathcal{A}_i)$  who agrees to pay the offer,  $\text{offer}(S(\mathcal{A}_j), g)$ , set by the selling agent for good  $g$ . In the completion of transaction, the selling agent  $S(\mathcal{A}_j)$  will be assigned a reputation rating according to its performance. Hence, the updated global reputation rating can be calculated as

$$\mathcal{R}^G(S(\mathcal{A}_j)) = \frac{1}{\ell} \sum_{r=1}^{\ell} \mathcal{R}^L(S(\mathcal{A}_j), r) \quad (4)$$

where  $\mathcal{R}^L(S(\mathcal{A}_j))$  is the local reputation of the selling agent,  $S(\mathcal{A}_j)$ , submitted by previous trading partners to insurance agents.  $r$  denotes the  $r^{th}$  transaction the selling agent  $S(\mathcal{A}_j)$  has conducted previously with any agents in the system and  $\ell$  denotes the total number of transactions.

## 4 Empirical Results

The objective of our experiment is to verify the effectiveness and benefits of our proposed models. Due to space limitation, we only report a summary of our results here. The full details can be obtained from [4].

One of the experimental results shows that the buying agents with insurance find more potential selling agents than the one without insurance in most cases. It is because the insured buying agents retrieved the list of selling agents associated with their reputation ratings from the insurance agents in the beginning of the search phrase. In this sense, the insured buying agents do not need to spend time in evaluating and searching for potential selling agents.

## 5 Related Works

The concept of insurance produces significant benefits for our human society. Nowadays, insurance companies usually compensate the insured party financially. Towards this, most of the insurance studies conducted in the context of economics theory concentrated on issues such as demand of insurance [5] and reinsurance [6]. Other domains have also borrowed insurance concepts to facilitate their needs. In distributed systems, insurance can be used in authentication procedures [7] and insuring cryptographic keys [8]. In the agent technology, the multi-agent brokering systems [9,10] employ broker agents to select and purchase online insurance products in the electronic marketplace.

Online Auctions equipped with the protection program in ensuring the quality of services. Although the protection program is free and user-friendly, it cannot fully meet the needs of all buyers. Auctions websites normally set maximum amount of coverage. If the buyer purchases an expensive item, in which its value is higher than the possible compensation amount, then the protection program

cannot fully protect users against the potential risk. In this sense, buyers must seek for external help, one of which is insurance.

## 6 Conclusions

We consider software agents working in a open marketplace, in which they are required to protect themselves from fraud and deception. Towards this, our models enable agents to use insurance as tools to (i) search for potential buyers and sellers, (ii) evaluate the trustworthiness of potential agents, and (iii) enhance the partner selection phrase.

In view of the above, our buying and selling models with insurance provide agents an extensive protection against deceitful behaviour, and to avoid risky situations. From our initial simulation results, we believe there are advantages in applying insurance concept in selling as well as buying in the open marketplace. Hence, our future work is to make further improvements the existing model.

## References

1. Schillo, M., Funk, P., Rovatsos, M.: Using Trust for Detecting Deceitful Agents in Artificial Societies. *Applied Artificial Intelligence, Special Issue on Trust, Deception and Fraud in Agent Societies* **14** (2000)
2. Lam, Y., Zhang, Z., Ong, K.: Insurance services in multi-agent systems. In: Proceedings of the 18th Australian Joint Conference on Artificial Intelligence. (2005) 664–673
3. Lam, Y., Zhang, Z., Ong, K.: Trading in open marketplace using trust and risk. In: Proceedings of Intelligent Agent Technology 2005. (2005) 471–474
4. Lam, Y., Zhang, Z., Ong, K.: Buying and selling with insurance in open multi-agent marketplace. Technical Report TRC06/06, School of Engineering and Information Technology, Deakin University, <http://www.deakin.edu.au/~yuk/TechReports/BuySellModel06.pdf> (2006)
5. Mossin, J.: Aspects of rational insurance purchasing. *Journal of Political Economy* **76** (1968) 553–568
6. Borch, K.H.: Economics of Insurance. North-Holland Amsterdam/New York/Oxford/Tokyo (1990)
7. Lai, C., Medvinsky, G., Neuman, B.: Endorsements, licensing, and insurance for distributed system services. In: Proceedings of the 2nd ACM Conference on Computer and Communication Security. (1994) 170–175
8. Millen, J., Wright, R.: Reasoning about trust and insurance in a public key infrastructure. In: 13th IEEE Computer Security Foundations Workshop. (2000) 16–22
9. Jonker, C., Lam, R., Treur, J.: A multi-agent architecture for an intelligent website in insurance. In: CIA '99: Proceedings of the Third International Workshop on Cooperative Information Agents III. (1999) 86–100
10. Nunes, L., Oliveira, E.: Brokering in electronic insurance markets. *The Interdisciplinary Journal of Artificial Intelligence and the Simulation of Behaviour* **1** (2003)

# Ensemble Evolution of Checkers Players with Knowledge of Opening, Middle and Endgame

Kyung-Joong Kim and Sung-Bae Cho

Department of Computer Science, Yonsei University  
134 Shinchon-dong, Sudaemoon-ku, Seoul 120-749 South Korea  
`{kjkim, sbcho}@cs.yonsei.ac.kr`

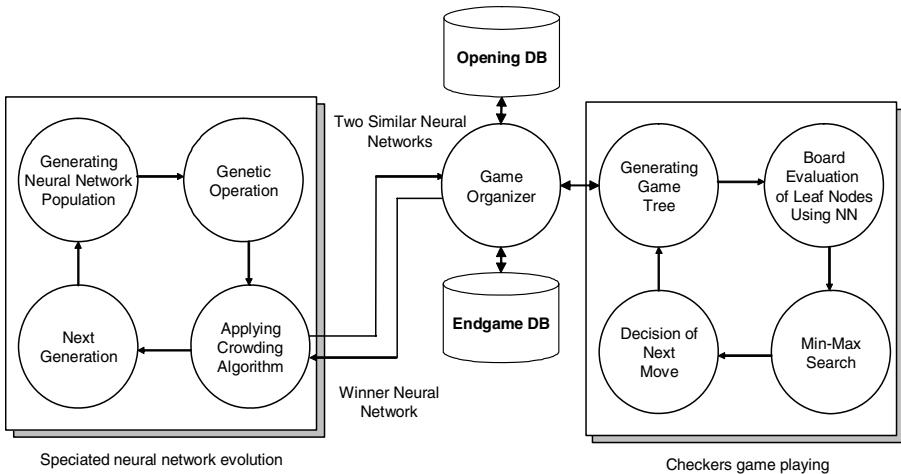
**Abstract.** In this paper, we argue that the insertion of domain knowledge into ensemble of diverse evolutionary checkers can produce improved strategies and reduce evolution time by restricting search space. The evolutionary approach for game is different from the traditional one that exploits knowledge of the opening, middle, and endgame stages, so that it is not sometimes efficient to evolve simple heuristic that is found easily by humans because it is based purely on a bottom-up style of construction. In this paper, we have proposed the systematic insertion of opening knowledge and an endgame database into the framework of evolutionary checkers. Also, common knowledge, the combination of diverse strategies is better than the single best one, is inserted into the middle stage and is implemented using crowding algorithm and a strategy combination scheme. Experimental results show that the proposed method is promising for generating better strategies.

## 1 Introduction

Incorporating a priori knowledge, such as expert knowledge, meta-heuristics, human preferences, and most importantly domain knowledge discovered during evolutionary search, into evolutionary algorithms has gained increasing interest in recent years [1]. In this paper, we propose a method for systematically inserting expert knowledge into an evolutionary checkers framework at the opening, middle, and endgame stages. In the opening stage, openings defined by the American Checkers Federation (ACF) are used. In previous work, we have used speciation techniques to search for diverse strategies that embody different styles of game play and have combined them using voting for higher performance [2]. This idea comes from the common knowledge that the combination of diverse well-playing strategies can defeat the best one because they can complement each other for higher performance. Finally, we have used an endgame database from Chinook, the first man-machine checkers champion. Figure 1 explains the conceptual framework of the proposed method.

The most important idea is the systematical integration of three domain knowledge (opening DB, middle stage knowledge and endgame DB). The middle stage knowledge is coming from the Korean event in the game of Go. In 2003, Internet site TYGEM (<http://www.tygem.co.kr>) held a many-to-one style game between Hoon Hyun Cho, one of the greatest go players, and 3000 amateur players. The winner of

the game was Cho. After the game, he said that it was a very difficult game because there was no obvious mistake of amateur players. Speciation algorithm for evolutionary checkers is adopted for an implementation of the knowledge.



**Fig. 1.** Conceptual diagram of the proposed method

## 2 Incorporating Knowledge into Evolutionary Checkers

### 2.1 Opening Stage

The opening move is the most important opportunity to defeat an expert player because trivial mistakes in the opening can lead to an early loss. The first move in checkers is played by red and there are seven choices (9-13, 9-14, 10-14, 10-15, 11-15, 11-16, and 12-16). Usually, 11-15 is the best move for red but there are many other alternatives. They are described with specific names, such as Edinburgh, Double Corner, Denny, Kelso, Old Faithful, Bristol, and Dundee, respectively. For each choice, there are many well established more sequences which range in length from 2 to 10. The longest sequence is described as the White Doctor: 11-16, 22-18, 10-14, 25-22, 8-11, 24-20, 16-19, 23-16, 14-23, 26-19. Careful analysis over decades of tournament play has proven the usefulness or fairness of the opening sequences. Initial sequences are decided by the opening book until the move is out of the book. Each player chooses their opening randomly and the seven first choices have the same probability to be selected as an opening.

### 2.2 Evolutionary Speciated Checkers

Following Fogel [3], a checkers board is represented by a vector of length 32 and components in the vector could have a value of  $\{-K, -1, 0, +1, +K\}$ , where  $K$  is the

value assigned for a king, 1 is the value for a regular checker, and 0 represents an empty square. For reflecting spatial features of the board configuration, sub-boards of the board are used as an input. One board can have 36 3×3 sub-boards, 25 4×4 sub-boards, 16 5×5 sub-boards, 9 6×6 sub-boards, 4 7×7 sub-boards and 1 8×8 sub-board. 91 sub-boards are used as an input to the feed-forward neural network. The sign of the value indicates whether or not the piece belongs to the player or the opponent. The closer the output of the network is to 1.0, the better the position is. Similarly, the closer the output is to -1.0, the worse the board.

The architecture of the network is fixed and only the weights can be adjusted by evolution. Each individual in the population represents a neural network (weights and biases) that is used to evaluate the quality of the board configuration. Additionally, each neural network has the value of  $K$  and self-adaptive parameters for weights and biases. An offspring  $P'_i, i = 1, \dots, p$  for each parent  $P_i, i = 1, \dots, p$  is created by

$$\begin{aligned}\sigma'_i(j) &= \sigma_i(j) \exp(\tau N_j(0,1)), \quad j = 1, \dots, N_w \\ w'_i(j) &= w_i(j) + \sigma'_i(j) N_j(0,1), \quad j = 1, \dots, N_w\end{aligned}$$

where  $N_w$  is the number of weights and biases in the neural network (here this is 5046),  $\tau = 1/\sqrt{2N_w} = 0.0839$ , and  $N_j(0,1)$  is the standard Gaussian random variable resampled for every  $j$ . In fitness evaluation, each individual chooses five opponents from a population pool and plays games with the players. Fitness increases by 1 for a win while the fitness of an opponent decreases by 2 for a loss. In a draw, the fitness values of both players remain the same. After all the games are played, the fitness values of all players are determined.

In this paper, we utilize a crowding algorithm [4], a popular form of speciation algorithm, for searching for diverse neural networks. In this algorithm, one neural network is selected from two similar individuals based on the result of game played between them (usually, a crowding algorithm uses their fitness but in this case, we cannot use fitness because of the dynamic property of fitness landscape). A crowding algorithm is one of the representative speciation methods that attempt to discover diverse species in a search space. The distance between two neural networks is calculated by using Euclidean distance between their weights. To discover clusters of individuals in the population at the last generation with arbitrary shape, density-based clustering methods have been used. DBSCAN (Density-based Spatial Clustering of Applications with Noise) is one of the algorithms [5]. Representative players from each cluster are chosen by tournament of all players in the same cluster. Moves of combined players are determined using a simple voting of the representative players. It picks the move that has the greatest number of votes. If there is no clear winner, one of the moves that have the greatest votes is selected randomly.

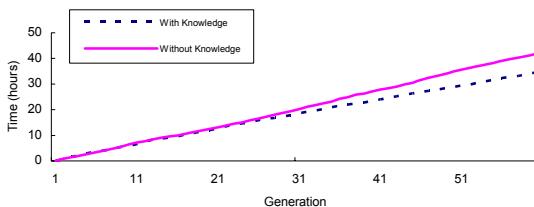
### 2.3 Endgame Stage

The estimated quality of the board is calculated using the evolved neural networks to evaluate the leaf nodes of the tree with min-max algorithm. If the value of  $f$  (estimated goodness of the next moves) is not reliable, we refer to the domain specific knowledge and revise  $f$ . The decision rule for querying the domain knowledge must

be defined previously as follows. **IF** ( $f < 0.75$  and  $f > 0.25$ ) or ( $f < -0.25$  and  $f > -0.75$ ) **THEN** querying the domain knowledge.

### 3 Experimental Results

The non-speciated evolutionary algorithm uses a population size of 15 and limits the run to 60 generations. The speciated evolutionary algorithm sets the population size to 15 and generations to 60. The mutation rate is 0.01 and crossover rate is 1.0. The number of leagues (it is used to select the best player from each species) is 5 (5 means that each player selects 5 players from species randomly and the competition results are used for the selection).



**Fig. 2.** Comparison of running time (Simple evolution)

The Chinook endgame DB (2~6 pieces) is used for revision when the estimated value from the neural network is between 0.25 and 0.75 or between -0.25 and -0.75. Time analysis indicates that the evolution with knowledge takes much less time than that without knowledge in simple evolution (Figure 2). This means that the insertion of knowledge within a limited scope can accelerate the speed of evolutionary algorithm because it can reduce computational requirement for finding optimal endgame sequence by using endgame DB. Table 1 summarizes the competition results between the best individual in the evolution with knowledge and the best individual in the evolution without knowledge for each generation. The knowledge incorporation model can perform better than the one without knowledge. Table 2 shows the competition results in the speciated evolution. Table 3 shows the effect of the stored knowledge (opening and endgame DB) in speciation.

**Table 1.** Experimental results on opening and endgame knowledge incorporation (Win/Lose/Draw) for simple evolution. Evolution with the stored knowledge performs better than that without the knowledge. (Op=Opening knowledge, SGA=Simple GA, E=Endgame knowledge).

Op+SGA+E	SGA	Generations				
		1~14	15~29	30~44	45~59	Total
Red	White	5/0/10	3/3/9	3/0/12	5/3/7	16/6/38
White	Red	4/3/8	4/2/9	5/4/6	4/2/9	17/11/32

**Table 2.** Experimental results on opening and endgame knowledge incorporation (Win/Lose/Draw) for speciated evolution. Evolution with the stored knowledge performs better than that without the knowledge. (S=Speciation).

<b>Op+S+E</b>	<b>Speciated</b>	<b>Generations</b>				
		1~14	15~29	30~44	45~59	Total
Red	White	5/1/9	4/3/8	6/0/9	8/2/5	23/6/31
White	Red	7/3/5	5/2/8	8/4/3	6/2/7	26/11/23

**Table 3.** The competition results between the speciated players using both opening and end-game DB and the speciated player with one of the knowledge

<b>Op+S+E</b>	<b>Op+S</b>	<b>Total</b>
Red	White	6/2/7
White	Red	8/4/3
<b>Op+S+E</b>	<b>S+E</b>	<b>Total</b>
Red	White	5/5/5
White	Red	4/5/6
<b>Op+S</b>	<b>S+E</b>	<b>Total</b>
Red	White	3/6/6
White	Red	2/7/6

## 4 Conclusion and Future Work

The final conclusion of the experiment is SGA < Speciated < Op+S < S+E  $\approx$  Op+S+E (SGA < Speciated is from the results of [2]). The effect of opening knowledge is not so big because they have only the limited sequences. The limited opening knowledge can prevent the player from making a big mistake but it is not much useful when the opponent chooses a move that is not included in the opening sequence. Multiple diverse neural networks can perform better than the single best one but there is always problem of combination and averaging may not work. As a future work, sophisticated combination method should be explored for better performance.

## References

1. Jin, Y.: Knowledge Incorporation in Evolutionary Computation, Springer (2004)
2. Kim, K.-J. and Cho, S.-B.: Evolving speciated checkers players with crowding algorithm. Proc. of the 2002 Congress on Evolutionary Computation (2002) 407-412
3. Fogel, D.B.: Blondie24: Playing at the Edge of AI. Morgan Kaufmann (2001)
4. Mahfoud, S. W.: Niching methods. Handbook of Evolutionary Computation, C6.1, IOP Publishing and Oxford University Press, (1997)
5. Ester, M., Kriegel, H.-P., Sander J. and Xu, X.: A density-based algorithm for discovering clusters in large spatial databases with noise. Knowledge Discovery and Data Mining (1996) 226-231

# Dynamic Game Level Design Using Gaussian Mixture Model

Sangkyung Lee and Keechul Jung<sup>\*</sup>

HCI Lab., School of Media, College of Information Science,  
Soongsil University, Seoul, South Korea  
[{monask, kcjung}@ssu.ac.kr](mailto:{monask, kcjung}@ssu.ac.kr)  
<http://hci.ssu.ac.kr/>

**Abstract.** In computer games, the level design and balance of character attributes are the key features of interesting games. Level designers adjust the attributes of the game characters and opponent behavior to create appropriate levels of difficult, and avoid player frustration. Generally, opponent behavior is defined by a static script, however, this results in repetitive levels and environments, making it difficult to maintain the player's interest. Accordingly, this paper proposes a dynamic scripting method that can sustain the degree of interest intended by the level designer by adjusting the opponent behaviors while playing the game. The player's countermeasure pattern for dynamic level design is modeled using a Gaussian Mixture Model (GMM). The proposed method is applied to a shooter game, and the experimental results maintain the degree of interest intended by the level designer.

## 1 Introduction

In recent years, the quality and complexity of computer games have significantly advanced, making it increasingly difficult to design games and game features that will attract players. Level design is widely recognized as a key feature of successful games and involves a successive change of parameters and opponent behavior to pose new challenges and maintain the interest of the player. Yet, setting a few pre-defined difficulty levels (e.g. beginner, intermediate, and advanced) does not necessarily maintain the degree of enjoyment intended by the level designer [1].

Accordingly, level designers use artificial intelligence (AI) to control individual characters, provide strategic direction to character groups, and dynamically change the parameters to make the game appropriately challenging. Various other approaches have also been used to create dynamic level design, such as neural networks and genetic algorithms [2]. However, despite the ability of AI to respond to changes in human player tactics, AI is still unable to maintain the degree of enjoyment intended by the level designer. Thus, another approach that includes opponent modeling is dynamic scripting [3], which assigns a probability of being picked to each rule. This

---

<sup>\*</sup> Corresponding author.

paper proposes a dynamic scripting method that is able to maintain the degree of enjoyment intended by the level designer using adjustments based on the user's patterns when playing the game.

## 2 Player's Countermeasure Patterns Modeled by GMM

The position of the user in the game is set as the player's countermeasure pattern, and the movement distribution of the user indicates an activity. A Gaussian Mixture Model is used to model the user's patterns. For density estimation, a GMM is usually estimated in an unsupervised manner [4]. A GMM can be modeled using an EM algorithm, which is an iterative optimization method to estimate some unknown parameters  $\Theta (\pi_k, \mu_k, \Sigma_k, k = 1, \dots, K)$ , where  $\pi_k$  means the weight of a component,  $\mu_k$  is the mean of a component.  $\Sigma$  is the covariance matrix of a component, and  $k$  indicates the number of Gaussian components. A GMM is used to model the player's countermeasure patterns. In this paper,  $k = 3$ . A GMM is a probability density function defined by equation (1).

$$G(x|\Theta) = \sum_{i=1}^k p(i) g(x|\theta_i) \text{ where, } \sum_{i=1}^k p(i) = 1, \quad p(i) \geq 0 \quad (1)$$

Where  $x$  denotes the feature vector,  $\Theta$  is  $\{p(i), \theta_i\}_{i=1}^k$ ,  $\theta_i$  indicates the mean and covariance and  $p(i)$  is the prior probability of the component. These parameters are estimated using an EM algorithm with two steps, the expectation step (E-step) and the maximization step (M-step). In each iteration step, the likelihood increases monotonously. Each parameter is set at an initial value and  $p(i)$  is set at  $1/k$ . The  $p(x)$  of each component is calculated in the E-step (Eq. 2).

$$P_i(x) = \frac{p(i)g(x|\theta_i)}{\sum_{i=1}^k p(i)g(x|\theta_i)} \quad (2)$$

Each parameter is maximized in the M-step (Eq. 3).

$$p(i) = \frac{\sum_{j=1}^n P_i(x_j)}{n}, \quad \mu_i = \frac{\sum_{j=1}^n P_i(x_j)x_j}{\sum_{j=1}^n P_i(x_j)}, \quad \Sigma_i = \frac{\sum_{j=1}^n P_i(x_j)(x_j - \mu_i)(x_j - \mu_i)^T}{\sum_{j=1}^n P_i(x_j)} \quad (3)$$

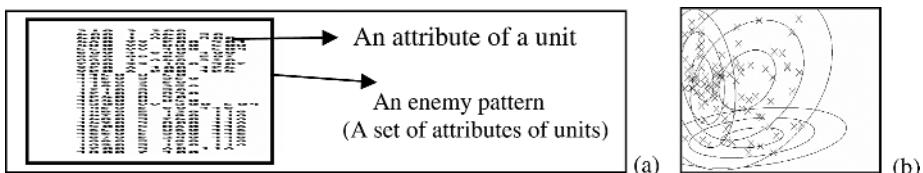
These two steps are reiterated until they converge (Eq. 4).

$$L(\Theta) = \sum_{j=1}^n \log G(x_j|\Theta) \quad (4)$$

## 3 Application to the Shooter Game

A GMM is used to model the user's reaction patterns in a 2D shooter game where the player operates an aircraft to destroy enemy aircraft. The patterns are then analyzed to

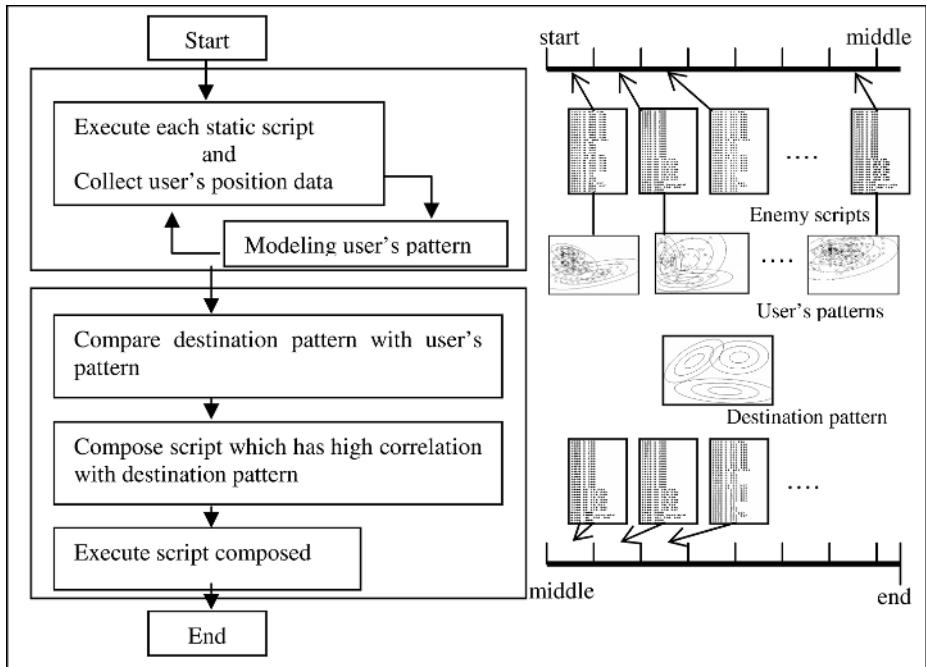
maintain the level intended by the level designer. In shooter games, the player obtains items and defeat enemies until meeting the arch enemy. If the player then destroys the archenemy, that level is completed. The progress is computed through a static script that contains such information as the enemy's movements and attributes. However, a static script can become boring due to its predictability and repetition. As such, a player can quickly learn how to deal with the enemy attack patterns, making the advanced levels easier than intended by the level designer. Thus, with a static script game, it is difficult to maintain interest at the degree of difficulty intended by the level designer. Therefore, in the proposed method, the progress of the game is divided into three steps. In the first step, a static script is used to extract the user's counter-measure patterns. These patterns are then analyzed to maintain the intent of the level designer while the user plays the game. In other words, the system selects scripts that will enable the game to maintain the intent of the level designer, even when played based on analyzing the user pattern results. The second step is constructed using the patterns created in the first step. The activity of the player can be made more dynamic by adjusting the script. The level designer needs to decide what to use as the domain. This paper uses the position of the player's aircraft, which is represented by a point in a 2-dimension coordinate system. Fig. 1(a) shows a script used to operate an enemy aircraft, where the single lines represent the attributes of the enemy aircraft and combine to make an enemy attack pattern within a certain period of time. The user's movement pattern corresponding to the enemy attack pattern is then extracted.



**Fig. 1.** (a) Script used in game, where single lines indicate unit attributes, while set of lines is defined as enemy attack pattern, (b) example of user pattern modeled by GMM

An example of a user's movement pattern is presented in Fig. 1 (b), where the x-axis and y-axis correspond to each axis of the 2-dimension coordinate system in the game. The sign X indicates the position of the user's aircraft within an enemy attack pattern. Each oval represents a Gaussian component, and the center of an oval is the mean of the Gaussian. After modeling the user's patterns, the data is then used to select scripts to create the environment intended by the level designer when playing the game.

Fig. 2 shows the process of the proposed system. A GMM is used to reflect the user's patterns, which represent the distribution of movement corresponding to each enemy pattern, until the middle of the game. The user's patterns are then compared with the destination pattern, reflecting the intention of the level designer, in order to choose enemy scripts that would make the player movement become like the destination pattern. The system then composes a script for use during the second half of the game.



**Fig. 2.** Flow chart of proposed system

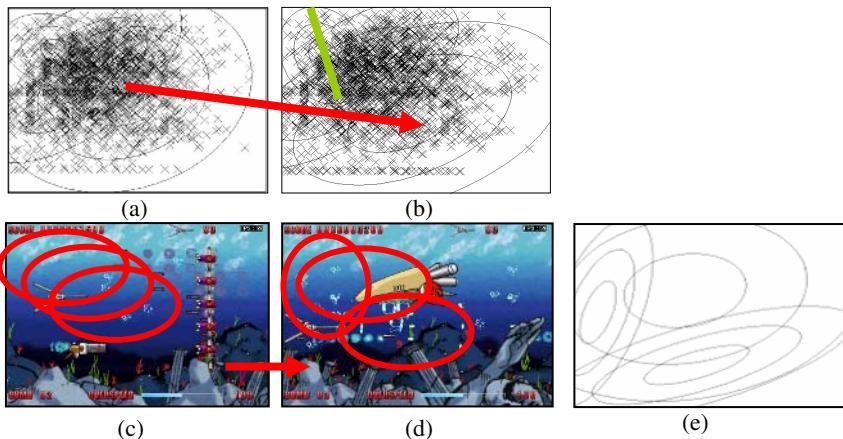
## 4 Experimental Results

The proposed method was applied to an actual shooter game<sup>1</sup>, where the player controls a character that is usually armed to destroy enemies (Fig. 3 (c), (d)).

The purpose of the proposed method is to increase the coherence between the player's movement and the movement intended by the level designer after applying a dynamic script. Seven enemy attack patterns were defined and each enemy attack pattern contained a combination of enemy characters, starting positions, and unit behavior. One enemy attack pattern continued for one minute and the player's position was sampled six times per second.

Fig 3(e) shows the distribution intended by the level designer that would make the game more fun, rather than frustrating. This pattern set is called the destination pattern. Fig. 3(b) shows the end result when using a dynamic script for the game. The user's pattern was observed to approach the destination pattern. The third Gaussian component became lower and larger than the user's position pattern by the middle of the game. The parameters of the component in fig. 3 (b) reflect the intention of the level designer for the overall movement of the player. Fig. 3 (c) and (d) indicate the real game space. The movement modeled by the GMM corresponded to an oval on the game screen, as follows.

<sup>1</sup> This study used the open source game released in 2001.(Title: Final Mission, Developer: Younho Lim).



**Fig. 3.** (a) User's position pattern by the middle of the game, (b) distribution chart of player's movements modeled by the GMM after applying dynamic script, (c) red ellipses indicating principal distribution of the player's movement on screen by the middle of game, (d) red ellipses indicating the principal distribution of the player's movement on screen by end of game after application of dynamic script, and (e) destination pattern indicating intent of level designer

## 5 Conclusion and Future Work

A dynamic scripting method was proposed that can maintain the degree of interest and enjoyment intended by level designers using adjustments based on the user's patterns while playing the game. A typical rule-based system does not consider the user's reaction, which means the system can not anticipate how the user will react to the rules picked by the system. Therefore, a user countermeasure model is needed based on probability. Since the main purpose of the proposed method is to maintain the degree of interest intended by the level designer, this was confirmed when comparing the destination pattern and the final user pattern.

**Acknowledgements.** This work is supported by the Soongsil University Research Fund.

## References

1. Noah Falstein.: Game Developer Magazine. The Flow Channel, (2004)
2. David B, Fogel.: Using Evolutionary Programming to Create Neural Networks that are Capable of Playing Tic-Tac-Toe. Intl. Joint Conference Neural Networks, New York (1993) 875-880
3. Pieter Spronck, Ida Sprinkhuizen-Kuyper and Eric Postma.: Difficulty Scaling of Game AI. In Proceedings of the 5<sup>th</sup> International Conference on Intelligent Games and Simulation, Belgium (2004) 33-37
4. Shental, N., Bar-Hillel, A., Hertz, T., Weinshall, D.: Computing Gaussian Mixture Models with EM using Equivalence Constraints. Advances in Neural Information Processing Systems 16, MIT Press (2004)

# Application Architecture of Data Mining in Telecom Customer Relationship Management Based on Swarm Intelligence

Peng Jin<sup>1,2</sup>, Yunlong Zhu<sup>1</sup>, Sufen Li<sup>1</sup>, and Kunyuan Hu<sup>1</sup>

<sup>1</sup> Shenyang Institute of Automation of the Chinese Academy of Sciences,  
Shenyang, 110016, China

<sup>2</sup> Graduate School of the Chinese Academy of Sciences  
Beijing, 100039, China

{jinpeng, ylzhu, lisufen, hukunyuan}@sia.cn

**Abstract.** Customer relationship management (CRM) is one of the biggest concerns in a telecom company. Application of data mining can support telecom CRM effectively, and a systematical architecture of data mining has been needed to support every aspects of telecom CRM roundly. To solve this problem, the systematical application architecture of data mining in telecom CRM is established and components of five modules in the architecture are specified in this paper. Data mining algorithms based-on swarm intelligence improved by our own have been adopted in these modules. SIMiner, a self-development data mining software system based on swarm intelligence, is applied in this architecture. Finally, an application example is given to illuminate that telecom companies can make marketing strategies roundly and effectively with the support of the application architecture.

## 1 Introduction

With improvement of hardware and network equipment, competition in the telecom service market has transferred to customer service. More and more telecom companies have been paying great concern on customer relationship management (CRM). Telecom companies have collected huge amount of customer data, in which so much information and knowledge are contained. It's important and difficult for telecom companies that how to process and analyze huge data and obtain valuable potential information and knowledge. Data mining is a useful tool to do this. It can acquire knowledge from huge data with automatic or semiautomatic methods for decision support.

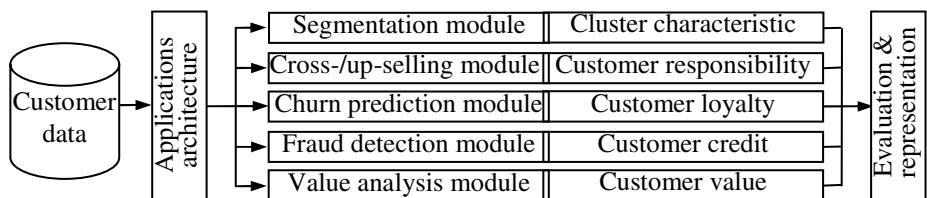
Swarm intelligence is a kind of evolutionary algorithm. In swarm intelligence, swarm consists of simple individuals. Swarm behavior is expressed by interactions among individuals and between individual and environment. Analogously, single customer data can't express a customer mode available, but some approximate customer data can express a valid customer mode which reflects common characteristic of this customer cluster. So we adopt the data mining algorithm based-on swarm intelligence in this research.

Application of data mining has been just beginning in telecom companies [3-5], but existing researches are static and isolated. Each of them is only focus on one aspect of telecom CRM but systematical application architecture of data mining has been needed to support every aspects of telecom CRM roundly. In this paper, the application of data mining in five business areas of telecom CRM is analyzed systematically. The application architecture is established and components of five modules in the architecture are specified to guide business applications of telecom CRM.

## 2 Application Architecture

Data mining can provide answers to many telecom business questions<sup>[1]</sup>, including: identify customer cluster characteristics and adopt market strategies accordingly; predict churning customer and maintain them; find out characteristics of customer fraud and prevent it; evaluate customer value; identify new prospects using demographic data; confirm candidate customer for cross-selling and up-selling; etc.

In this paper, we summarize above-mentioned questions into five business application areas of telecom CRM: customer segmentation, customer churn prediction, customer fraud detection, customer value analysis and cross-selling/up-selling. These five application areas are not isolated. CRM architecture can't be implemented well to give enough decision support if only one of these areas is analyzed. Accordingly, we present application architecture of data mining in telecom CRM, shown in figure 1.



**Fig. 1.** Applications architecture of data mining in telecom CRM

### 2.1 Customer Segmentation Module

In customer segmentation module, we adopted the Ant-Cluster algorithm, a clustering analysis algorithm proposed by our own based-on swarm intelligence. Ant-Cluster algorithm will be discussed in another paper. Ant-Cluster algorithm is implemented with SWARM, agent-based model simulation software, which is integrated in SIMiner developed by our own with Java.

Evaluation and interpretation of clustering result is a key step of customer segmentation. Analyzing value distributions of every attributes in each cluster and comparing them with value distributions of the same attribute in entire data set, then cluster characteristics can be obtained for decision support.

### 2.2 Customer Churn Prediction Module

In customer churn prediction module, we adopted ACO-Miner algorithm, a classification algorithm proposed by our own based on the ant colony optimization. ACO-Miner algorithm will be discussed in another paper.

Customer satisfaction is also involved in churn prediction. It should be noticed that satisfaction is not equal to loyalty. Satisfaction can't be used to predict churn directly, but it can be used as assistant condition to make appropriate strategies for churning customer. For example, customers with lower loyalty and higher satisfaction should be retained as possible as company can.

### **2.3 Customer Fraud Detection Module**

Customer frauds in telecom business mostly mean intended arrearage behaviors. Because both absolute amount and relative amount of customer fraud behavior are small, the proportion of samples should be noticed when sampling from data set. We propose that fraud customer should account for about 10 percent. Chosen samples will be used as input of this module.

Customer fraud detection module adopts ACO-Miner algorithm to distinguish fraud behavior. After training input data set, a set of classification rules is obtained for fraud detection.

### **2.4 Customer Value Analysis Module**

In customer value analysis module, Ant-Cluster algorithm is applied to cluster customer data according to three attributes: current value (c), potential value (p) and credit risk (r).

Evaluation and interpretation of clustering result are similar to the method in customer segmentation. Customer cluster characteristics can be obtained then. According to these characteristics, different strategies can be adopted. For instance, for the cluster which has higher current value and potential value, but lower loyalty or credit, i.e. higher credit risk, favorable strategies should be adopted to promote their loyalty.

### **2.5 Cross-Selling/Up-Selling Module**

The cross-selling/up-selling module adopts Apriori algorithm. The output of cross-selling/up-selling module is association rules and customer responsibility. The minimum confidence of association rules obtained by Apriori algorithm is regarded as customer responsibility.

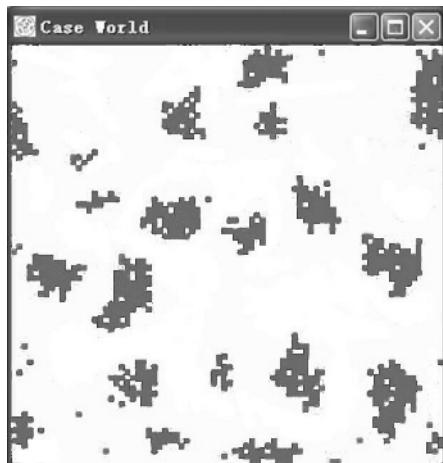
Additionally, the result of customer segmentation can support to make strategies of cross-selling/ up-selling. For example, customers in a certain cluster are almost use “call transfer” service, then telecom company can up-selling this service to the rest customers of this cluster who hasn’t used “call transfer” service.

## **3 Application Example**

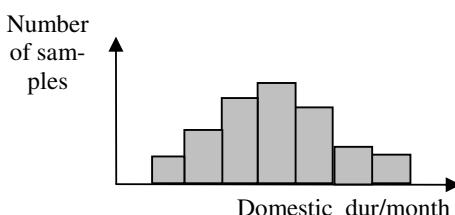
In this section, we will give an example to illuminate the process of applying application architecture of data mining in telecom CRM, using SIMiner, a self-development data mining software system based on swarm intelligence. We chose a telecom company’s customer data as the input of application architecture.

The process of applying application architecture of data mining in telecom CRM with SIMiner is described as follows.

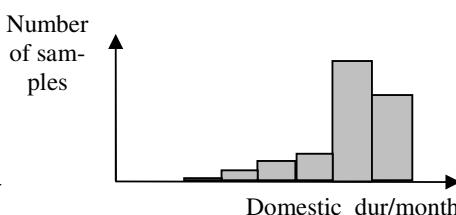
- 1) Data preparation: According to the different demand of each module for input data, customer data is processed and relative attributes are chosen.
- 2) Analysis with data mining modules: Prepared data is inputted into relevant data mining module of application architecture and data mining model are achieved finally. For example, customer segmentation module uses Ant-Cluster algorithm to get customer clusters. The algorithm is implemented with Swarm, which is integrated in SIMiner. Result of customer segmentation is illuminated in figure 2. In figure 2, each cluster figures one customer cluster. Analysis of this result is given in the next step.
- 3) Evaluation and interpretation of results: Evaluation and interpretation of data mining result is a key step of application process. Useful rules and knowledge can be obtained by evaluation and interpretation for decision support. Figure 3 and 4 illuminate distribution of domestic call time attribute of one cluster outputted by customer segmentation module. As is shown in figure 4, domestic call time in this cluster is longer than that in all customer data set shown in figure 3. Therefore, we can draw a conclusion that higher domestic call time is one of characteristics of this cluster. Appropriate marketing strategy should be made according to this.



**Fig. 2.** Result of customer segmentation obtained by Swarm



**Fig. 3.** Distribution of domestic call time in all customer data set



**Fig. 4.** Distribution of domestic call time in a certain customer cluster

This application example shows that the architecture can deal with each area of telecom CRM and support telecom companies to make marketing strategies roundly.

## 4 Conclusion

In this paper, we analyzed applications of data mining in telecom CRM systematically. Because application of data mining in each telecom business area is not isolated, we establish systematical application architecture of data mining in telecom CRM and specify components of each module in the architecture. The methods based on swarm intelligence are adopted in each module and implemented in SIMiner, a self-development data mining software system based on swarm intelligence. Finally, an application example is given and shows that the architecture can support decision-making roundly.

## References

1. Baragoin, C.: Mining Your Own Business in Telecoms Using DB2 Intelligent Miner for Data. IBM Corporation (2001)
2. Zhu, Y. L., Nan, L., Wang, F. D.: CRM Concepts, Methods, and Solution. Tsinghua University Press, Beijing (2004)
3. Wei, C. P., Chiu, I. T.: Turning telecommunications call details to churn prediction: a data mining approach. *Expert Systems with Applications* 23 (2002) 103-112
4. Daskalaki, S., Kopanas, I., Goudara, M., Avouris, N.: Data mining for decision support on customer insolvency in telecommunications business. *European Journal of Operational Research* 145 (2003) 239-255
5. Hwang, H., Jung, T., Suh, E.: An LTV model and customer segmentation based on customer value: a case study on the wireless telecommunication industry. *Expert Systems with Applications* 26 (2004) 181-188
6. Kim, E., Kim, W., Lee, Y.: Combination of multiple classifiers for the customer's purchase behavior prediction. *Decision Support Systems* 34(2002) 167-175
7. Shaw, M. J., Subramaniam, C., Tan, G. W., Welge, M. E.: Knowledge management and data mining for marketing. *Decision Support Systems* 31(2001) 127-137
8. Gerpott, T. J., Rams, W., Schindler, A.: Customer retention, loyalty and satisfaction in the German mobile cellular telecommunications market. *Telecommunications Policy* 25 (2001) 249-269
9. Wu, B., Shi, Z. Z.: A clustering algorithm based on swarm intelligence. In: Proceedings of the International Conferences on Info-tech and Info-net, Beijing (2001) 58-66
10. Parpinelli, R. S., Lopes, H. S., Freitas, A. A.: Data mining with an ant colony optimization algorithm. *IEEE Transactions on Evolutionary Computation* 6 (2002) 321-332

# Mining Image Sequence Similarity Patterns in Brain Images

Pan Haiwei<sup>1</sup>, Xiaoqin Xie<sup>1</sup>, Zhang Wei<sup>2</sup>, and Jianzhong Li<sup>2</sup>

<sup>1</sup> Dept. of Computer Science, Harbin Engineering University, Harbin, P.R. China  
[{panhaiwei\\_xiexiaoqin}@hrbeu.edu.cn](mailto:{panhaiwei_xiexiaoqin}@hrbeu.edu.cn)

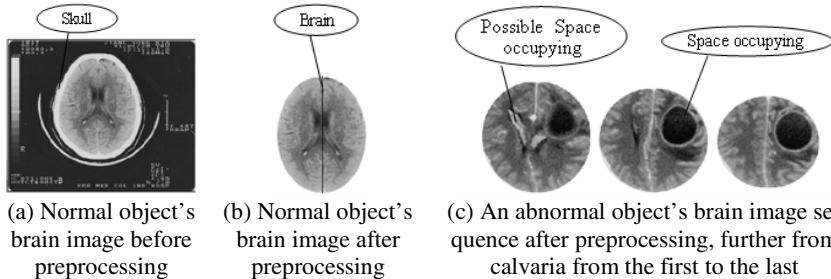
<sup>2</sup> Dept. of Computer Science, Harbin Institute of Technology, Harbin, P.R. China  
[{lijzh\\_wzhang74}@hit.edu.cn](mailto:{lijzh_wzhang74}@hit.edu.cn)

**Abstract.** The high incidence of brain disease, especially brain tumor, has increased significantly in recent years. It is becoming more and more concerned to discover knowledge through mining medical brain image to aid doctors' diagnosis. In this paper, we introduce a notion of image sequence similarity patterns (ISSP) for medical image database. These patterns are significant in medical images because it is the similarity of objects each of which has an image sequence that is meaningful. We design the new algorithms with the guidance of the domain knowledge to discover ISSP for similarity retrieval. Our experiments demonstrate that the results of similarity retrieval are satisfying.

## 1 Introduction

Advances in image acquisition and storage technology have led to tremendous growth in very large and detailed image databases. A vast amount of image data is generated in our daily life and each field. These images involve a great number of useful and implicit information that is difficult to discover. Image mining can automatically discover these implicit information and patterns and is rapidly gaining attention in the field of data mining. Image mining is more than just an extension of data mining to image domain. It is an interdisciplinary endeavor that draws upon image processing, machine learning, artificial intelligence, database and data mining, etc. While some of individual fields in themselves may be quite matured, image mining, to date, is still at an experimental stage. Research in image mining can be broadly classified to two main directions: (1) domain-specific applications; (2) general applications [2]. The focus in the first direction is to extract the most relevant image features into a form suitable for data mining [3] and the latter is to generate image patterns that may be helpful in understanding of the interaction between high level human perceptions of image and low level image features [4]. Data mining in medical images belongs to the first direction.

During these years, the incidence of brain disease (especially brain tumor) has increased significantly and the early diagnosis of brain diseases is becoming more and more crucial. Computerized Tomography (CT) is one of the most important techniques that are used to diagnose by doctors. Brain CT scan of each patient (as an object below) is an image sequence in which each one is an image of a layer every a few millimeters from calvaria. There exists a certain spatial relationship between images in the sequence.



**Fig. 1.** An example of normal and abnormal brain image

At present, the main work of data mining on medical images [7] has two characteristics: (1) Research content is the images in the medical image database, not the objects with medical images. (2) Research method is to extract features from images to form feature attributes and use data mining on these attributes. These work paid little attention to the guidance effect of domain knowledge to data mining. Our research content is objects each of which contains a series of images and the images from different objects maybe have different intensity. The images from one object have the same structure but come from the different layers of a brain. They have similar pixel density distribution and it is possible for two spatial-adjacent images to be similar in one object, see figure 1(c). Each image contains some Possible Space-Occupying Lesions (PSOs) that also have a spatial relationship in an image. Therefore, the objects with the similar image sequence patterns (ISP) should have similar clinical manifestations. It is very helpful to assist medical doctors to make a diagnosis.

The rest of the paper is organized as follows: Pre-processing is presented in section 2. Section 3 presents Similarity Retrieval Based On ISSP. Section 4 reports the results of our experiments. Conclusions are presented in section 5.

## 2 Pre-processing

In preprocessing, our work firstly uses domain knowledge effectively to remove the noisy data. A brain CT image mainly consists of three parts: noisy data, skull and cerebrum. The noisy data includes the black background and patient's name etc. These information is not only helpless but revealing patient's privacy. We are only interested in cerebrum. So we use domain knowledge (for short DK1) and cropping technique of image processing to gain it.

**DK1** --- human's brain skull has the highest density and surrounds the cerebrum.

That is, the skull is a cricoid area in the image with the whitest pixels that separate cerebrum from the noisy data, see figure 1(a). It becomes easy to remove the noisy data by using cropping technique and keep the interesting region with the guidance of DK1. All the objects are formed as follows, see table 1. Each object has a unique identification (ID) and its image part (IM) is a preprocessed image sequence where every code is composed of id, the image sequence number and the arrow represents the spatial relationship.

After image cropping, a method based on pixel's clustering [14] are utilized to detecting the whole PSOs in an image.

**Table 1.** Each object is formed as a record in the table after its images are preprocessed

ID	IM
001	001.01→001.02→001.03→...→001.07
002	002.01→002.02→002.03→...→002.13
...	.....
n	n.01→n.02→n.03→...→n.10

### 3 Similarity Retrieval Based On ISSP

In this section, we will: (1) discover the image sequence pattern (ISP) of one object; (2) discover the image sequence similarity patterns (ISSP) of two objects.

We denote each PSO as the following:  $\langle H(L), (x_i, y_i), (x_{a1}, x_{a2}), (y_{b1}, y_{b2}) \rangle$ , where  $H(L)$  represents the high (low) bound GS,  $(x_i, y_i)$  is the coordinate of the center of of this PSO,  $(x_{a1}, x_{a2})$  and  $(y_{b1}, y_{b2})$  are the max and min x and y coordinate of the PSO. They are computed by the following formula:

$$x_i = \frac{1}{k} \sum_{j=1}^k x_j \quad (1) \quad y_i = \frac{1}{k} \sum_{j=1}^k y_j \quad (2) \quad x_{a1} = \max_{j=1}^k (x_j) \quad (3)$$

$$x_{a2} = \min_{j=1}^k (x_j) \quad (4) \quad y_{b1} = \max_{j=1}^k (y_j) \quad (5) \quad y_{b2} = \min_{j=1}^k (y_j) \quad (6)$$

We take the center of the brain as the origin of coordinates. If  $x_i \leq 0$ , then this PSO is in  $IM(L)$ . Otherwise, it is in  $IM(R)$ .

**Definition 1.**  $PSO_k$  is prior to  $PSO_j$  if for  $PSO_k = \langle H(L), (x_k, y_k), (x_{ka1}, x_{ka2}), (y_{kb1}, y_{kb2}) \rangle$  and  $PSO_j = \langle H(L), (x_j, y_j), (x_{ja1}, x_{ja2}), (y_{jb1}, y_{jb2}) \rangle$ , they satisfy one of the three prior conditions: (a)  $x_k < x_j$ ; (b)  $x_k = x_j$  and  $y_k > y_j$ ; (c)  $x_{ka1} \leq x_{ja1}$  and  $x_{ka2} \geq x_{ja2}$ ; (d)  $y_{kb1} \leq y_{jb1}$  and  $y_{kb2} \geq y_{jb2}$ . We denote it as  $PSO_k >> PSO_j$ .

According to this priority, we describe PSO pattern (PSOP) of each image as the following form:  $PSOP(IM_i) = \langle L_{i1}, L_{i2}, \dots, L_{im}, R_{i1}, R_{i2}, \dots, R_{in} \rangle$ . Where  $L_{im}$  and  $R_{in}$  represent a PSO in  $IM_i(L)$  and  $IM_i(R)$  respectively.

**Definition 2.** PSOP of two images is complete similar if  $|PSOP(IM_i)| = |PSOP(IM_j)|$ , and the corresponding  $L_{ik}$  and  $L_{jk}$  (or  $R_{ik}$  and  $R_{jk}$ ) have the same bound GS, locate the same part of the image ( $IM(L)$  or  $IM(R)$ ) and satisfy the same prior condition. That is, if  $L_{ik}$  and  $L_{i(k+1)}$  satisfy the prior condition (a), then  $L_{jk}$  and  $L_{j(k+1)}$  must satisfy (a).

**Definition 3.** PSOP of two images is incomplete similar if cut one or more discontinuous PSO from one or both of these two images, then  $|PSOP(IM_i)| = |PSOP(IM_j)|$ , and the corresponding  $L_{ik}$  and  $L_{jk}$  (or  $R_{ik}$  and  $R_{jk}$ ) have the same bound GS, locate the same part of the image ( $IM(L)$  or  $IM(R)$ ) and satisfy the same prior condition.

For each image of one object, it has a PSOP and maybe is different from the PSOP of the adjacent image. We compare the PSOP of all adjacent images and get a pattern sequence in which two adjacent patterns are not same. This pattern sequence is called image sequence pattern (ISP). The algorithm of discovering ISP is as follows.

**DISP Algorithm**

**Input:** m images of one object

**Output:** Image sequence patterns (ISP)

1. Initialization:  $j=1, n_j=1, k=1$ ;
2. For  $i = 1$  to  $m$  {
3. Compare the pattern PSOP( $IM_i$ ) and PSOP( $IM_{i+1}$ ) of  $i_{th}$  and  $(i+1)_{th}$  image;
4. If complete similar, Then record the pattern as  $<PSOP(IM_k), n_j = n_j + 1>$ ;
5. Else if  $j = i$
6. Then  $k=i$  and record the pattern as  $<PSOP(IM_k), n_j = 1>; j=j+1$ ;
7. Else  $k=i+1$  and record the pattern as  $<PSOP(IM_k), n_j = 1>; j=j+1$ ;

For  $ISP(S_i)$  and  $ISP(S_j)$ , ISSP of two objects refers to the longest similar and continuous sub-patterns that belongs to the ISP of each object. Since there are spatial relationship between all images in  $S_i$ , that is, for any  $IM_{i1}, \dots, IM_{in}, IM_{i(j+1)}$  must be farther from calvaria than  $IM_{ij}$ , it is not necessary to retrieve the whole PSOPs of one object to find the similar pattern of a given  $M_i$ . For example,  $M_1$  is PSOP of the farthest image from calvaria of one object and  $M_p$  is PSOP of the nearest image of another different object, it is not meaningful to compare  $M_1$  and  $M_p$  to find whether they are similar or not because  $M_1$  and  $M_p$  show the different parts of the brain. According to this, two rules are introduced to reduce the retrieval space. For two objects  $S_i$  and  $S_j$ , assumed that  $ISP(S_i)=m$  and  $ISP(S_j)=n$ , (1) if  $m=n$ , that is, the number of the PSOPs in these two objects is equal, then we only need to retrieve  $M_{i-1}, M_i$  and  $M_{i+1}$  in  $ISP(S_i)$  to discover the similar patterns of  $M_i$  in  $ISP(S_i)$ . (2) If  $m < n$ , then we only need to retrieve  $M_i, \dots, M_{i+n-m}$  in  $ISP(S_i)$  to discover the similar patterns of  $M_i$  in  $ISP(S_i)$ . We define two objects are similar if there exists ISSP in two objects.

**DISSP Algorithm**

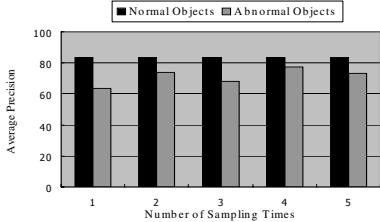
**Input:** ISP of two objects

**Output:** Image sequence similarity patterns (ISSP)

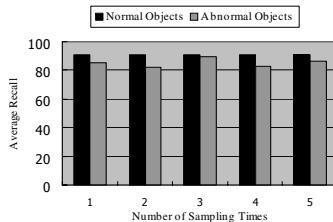
1. Assumed that  $ISP(S_i)=m$  and  $ISP(S_j)=n$ , if  $m \leq n$ , start the following steps:
2. For  $i = 1$  to  $m$  {
3. if  $m=n$ , then goto step (5); if  $m < n$ , then goto step(7);
4. Compare  $M_i$  in  $ISP(S_i)$  with  $M_{i-1}, M_i$  and  $M_{i+1}$  in  $ISP(S_j)$ . If discover a complete similar PSOP,  $C=C \cup (M_i \rightarrow M_j)$ . If there is no PSOP to compare in  $ISP(S_j)$ , return step (3) to start the next iteration. If there is no complete similar PSOP to be discovered, goto step (6);
5. Compare  $M_i$  in  $ISP(S_i)$  with  $M_{i-1}, M_i$  and  $M_{i+1}$  in  $ISP(S_j)$ . If discover a incomplete similar PSOP,  $NC=NC \cup (M_i \rightarrow M_j)$ . If there is no PSOP to compare in  $ISP(S_j)$ , return step (3) to start the next iteration;
6. Compare  $M_i$  in  $ISP(S_i)$  with  $M_i, M_{i+1}, \dots, M_{i+n-m}$  in  $ISP(S_j)$ . If discover a complete similar PSOP,  $C=C \cup (M_i \rightarrow M_j)$ . If there is no PSOP to compare in  $ISP(S_j)$ , return step (3) to start the next iteration. If there is no complete similar PSOP to be discovered, goto step (8);
7. Compare  $M_i$  in  $ISP(S_i)$  with  $M_i, M_{i+1}, \dots, M_{i+n-m}$  in  $ISP(S_j)$ . If discover a incomplete similar PSOP,  $NC=NC \cup (M_i \rightarrow M_j)$ . If there is no PSOP to compare in  $ISP(S_j)$ , return step (3) to start the next iteration; }
8. Order the whole patterns in  $C \cup NC$  by the spatial relationship of  $M_i$  and use the exhausting method to discover the ISSP;

## 4 Experiments

The dataset in our experiments was real data from hospital including 11 normal and 92 abnormal objects data.



**Fig. 2.** Average Precision when the Normal and Abnormal Objects as Targets



**Fig. 3.** Average Recall when the Normal and Abnormal Objects as Targets

The average precision of the abnormal targets is more than 60% but not very high. All retrieved objects are certainly the abnormal objects and the image sequence of them is similar to that of the target. The reason why not very high is that their concrete tumor kind is not the same as the target and the medical doctors made a different detailed diagnosis. But this demonstrates that our similarity retrieval algorithm based on ISSP can gain the similar image sequence of the target. The average recall, however, is very high. This illustrates that the results of our similarity retrieval method based on ISSP can include most of the similar objects with the targets.

## 5 Conclusions

In this paper, we introduce image sequence similarity patterns (ISSP) for medical image database and design a new algorithm with the guidance of the domain knowledge to generate ISSP for similarity retrieval. Our experiments demonstrate that the results of similarity retrieval are meaningful.

## References

- [1] WYNNE HSU, MONG LI LEE, JI ZHANG. Image Mining: Trends and Developments. *Journal of Intelligent Information Systems*, 19:1, 7–23, 2002.
- [2] Osmar R. Zaiane, Maria-Luiza Antonie, Alexandru Coman. Mammography Classification by an Association Rule-based Classifier. *Proceedings of the Third International Workshop on Multimedia Data Mining (MDM/KDD'2002)*.
- [3] Kitamoto, A. (2001). Data Mining for Typhoon Image Collection. In *Second International Workshop on Multimedia Data Mining (MDM/KDD'2001)*.
- [4] Ordonez, C. and Omiecinski, E. (1999). Discovering Association Rules Based on Image Content. In *IEEE Advances in Digital Libraries Conference*.
- [5] Pan Haiwei, Jianzhong Li, Zhang Wei, Fang Fang, Detecting Brain Tumor Based on Pixel's Clustering, *17th International EURASIP Conference BIOSIGNAL*, 2004

# Weightily Averaged One-Dependence Estimators

Liangxiao Jiang<sup>1</sup> and Harry Zhang<sup>2</sup>

<sup>1</sup> Faculty of Computer Science, China University of Geosciences  
Wuhan, Hubei, P.R. China, 430074

ljiang@cug.edu.cn

<sup>2</sup> Faculty of Computer Science, University of New Brunswick  
P.O. Box 4400, Fredericton, NB, Canada E3B 5A3  
hzhang@unb.ca

**Abstract.** NB(naive Bayes) is a probabilistic classification model, which is based on the attribute independence assumption. However, in many real-world data mining applications, this assumption is often violated. Responding to this fact, researchers have made a substantial amount of effort to improve NB's accuracy by weakening its attribute independence assumption. For a recent example, Webb et al.[1] propose a model called Averaged One-Dependence Estimators, simply AODE, which weakens the attribute independence assumption by averaging all models from a restricted class of one-dependence classifiers. Motivated by their work, we believe that assigning different weights to these one-dependence classifiers can result in significant improvement. Based on this belief, we present an improved algorithm called Weightily Averaged One-Dependence Estimators, simply WAODE. We experimentally tested our algorithm in Weka system[2], using the whole 36 UCI data sets[3] selected by Weka[2], and compared it to NB, SBC[4], TAN [5], NBTree[6], and AODE[1]. The experimental results show that WAODE significantly outperforms all the other algorithms used to compare.

**Keywords:** naive Bayes, Bayesian networks, AODE, WAODE, classification.

## 1 Introduction

Classification is one of the most important tasks in data mining. Learning Bayesian classifiers is a process of constructing a special classifier from a given set of training instances with class labels. Assume  $A_i, i = 1, 2, \dots, n$ , are  $n$  attributes which take values  $a_i, i = 1, 2, \dots, n$  respectively. These attributes will be used collectively to predict the value  $c$  of the class  $C$ . Thus, the Bayesian classifier can be defined as:

$$\arg \max_{c \in C} P(c) P(a_1, a_2, \dots, a_n | c) \quad (1)$$

Assume all attributes are independent given the class, then the resulting classifier is called *naive Bayes*:

$$\arg \max_{c \in C} P(c) \prod_{i=1}^n P(a_i | c) \quad (2)$$

In naive Bayes, each attribute node has the class node as its parent, but does not have any parent from attribute nodes. A central assumption made by the naive approach is that attributes are independent within each class, which harms the classification process when it is violated.

In order to relax this assumption effectively, an appropriate language and efficient machinery to represent and manipulate independence assertions are needed[5]. Both are provided by *Bayesian networks*[7]. Unfortunately, however, it has been proved that learning an optimal Bayesian networks is NP-hard[8]. In order to avoid the intractable complexity for learning Bayesian networks, learning improved naive Bayes has attracted much attention from researchers. Related work can be broadly divided into three approaches: 1) Selecting attributes subsets in which attributes satisfy the attribute independence assumption, for example, SBC[4]; 2) Extending the structure of naive Bayes to represent the dependencies among attributes, for example, TAN[5]; 3) Employing the principle of local learning to extend naive Bayes, for example, NBTree[6].

The most recent work on improving NB is called *Averaged One-Dependence Estimators*, simply AODE[1]. In AODE, an aggregate of one-dependence classifiers are learned and the prediction is produced by averaging the predictions of all these qualified one-dependence classifiers. For simplicity, a one-dependence classifier is firstly built for each attribute, in which the attribute is set to be the parent of all other attributes. Then, AODE directly averages the aggregate consisting of many special tree augmented naive Bayes.

AODE classifies a test instance using Equation 3.

$$\arg \max_{c \in C} \left( \frac{\sum_{i=1, AF(a_i) \geq m}^n P(a_i, c) \prod_{j=1, j \neq i}^n P(a_j | a_i, c)}{numParent} \right) \quad (3)$$

where  $F(a_i)$  is a count of the number of training instances having attribute-value  $a_i$  and is used to enforce the limit  $m$  that they place on the support needed in order to accept a conditional probability estimate.  $n$  is the number of attributes.  $unmParent$  is the number of the root attributes, which satisfy the condition that the training instances contain more than  $m$  examples with the value  $a_i$  for the parent attribute  $A_i$ . In the current research they use  $m = 30$ .

In addition to, AODE estimates the base probabilities  $P(a_i, c)$  and  $P(a_j | a_i, c)$  using the Laplace estimate as follows:

$$P(a_i, c) = \frac{F(a_i, c) + 1}{N + v_i * k} \quad (4)$$

$$P(a_j | a_i, c) = \frac{F(a_j, a_i, c) + 1}{F(a_i, c) + v_j} \quad (5)$$

where  $F(\bullet)$  is the frequency with which a combination of terms appears in the training data,  $N$  is the number of training instances,  $v_i$  is the number of values of the root attribute  $A_i$ ,  $v_j$  is the number of values of the leaf attribute  $A_j$ , and  $k$  is the number of classes.

Except for having good classification performance, AODE[1] retains the simplicity and direct theoretical foundation of NB while alleviating the limitations

of its conditional independence assumption. AODE only has a training time complexity of  $O(Nn^2)$  and classification time complexity of  $O(kn^2)$ .

The rest of the paper is organized as follows. In Section 2, we present an improved algorithm simply called WAODE. In Section 3, we describe the experimental setup and results in detail. In Section 4, we draw conclusions and outline our main directions for future research.

## 2 Weightily Averaged One-Dependence Estimators

Just as discussed above, for each attribute, a special tree augmented naive Bayes is built in AODE, in which the attribute is the parent of all other attributes. The attribute is called root attribute. AODE averages the aggregate consisting of these special tree augmented naive Bayes. In the classification of AODE, each tree augmented naive Bayes is treated equally. That essentially means that each attribute is treated equally. However, in many real world applications, attributes do not play the same role in classification. Some of them are more important than others. Thus, a natural way to extend AODE is to assign each attribute different weight. More precisely, weight each tree augmented naive Bayes differently. We believe that doing this could result in significant improvement. This is the key idea of our new algorithm called *Weightily Averaged One-Dependence Estimators*, simply WAODE.

Similar to AODE, a tree augmented naive Bayes is built for each attribute, in which the attribute is set as the root attribute. But each tree augmented naive Bayes is assigned a weight. WAODE classifies an instance using Equation 6.

$$\arg \max_{c \in C} \left( \frac{\sum_{i=1}^n W_i P(a_i, c) \prod_{j=1, j \neq i}^n P(a_j | a_i, c)}{\sum_{i=1}^n W_i} \right) \quad (6)$$

where  $W_i$  is the weight of the tree augmented naive Bayes for attribute  $A_i$ .

There are two major approaches to learn weights. One is to conduct a search process to find the weights that maximize the performance of the resulting model. Usually, this approach leads to good weight assignment, but it needs significant amount of time for the search. Another more efficient way is to directly compute the weights. In classification, the importance of an attribute can be measured by the correlation between it and the class variable. The mutual information  $I_P(A_i; C)$  between the attribute  $A_i$  and the class variable  $C$  can be naturally used for this purpose. In information theory, mutual information is formally defined as follows.

Let  $X, Y$  are two variables, the mutual information between  $X$  and  $Y$  is defined by the following Equation.

$$I_P(X; Y) = \sum_{x,y} P(x, y) \log \frac{P(x, y)}{P(x)P(y)}. \quad (7)$$

In WAODE, we use  $I_P(A_i; C)$  as the weight of attribute  $A_i$  (that is  $W_i$  in Equation 6.). In our implementation of WAODE, the base probabilities  $P(a_i, c)$

and  $P(a_j|a_i, c)$  are estimated using the M-estimation as follows:

$$P(a_i, c) = \frac{F(a_i, c) + 1.0/(v_i * k)}{N + 1.0} \quad (8)$$

$$P(a_j|a_i, c) = \frac{F(a_j, a_i, c) + 1.0/v_j}{F(a_i, c) + 1.0} \quad (9)$$

Compared to AODE, WAODE needs to compute the mutual information  $I_P(A_i; C)$  between the root attribute  $A_i$  of the tree and the class  $C$ . The time complexity for computing these weights only is  $O(knv)$ , where  $v$  the average number of values for an attribute. So, WAODE only has a training time complexity of  $O(Nn^2 + knv)$ . Since  $kv$  is much less than  $Nn$  in reality, the training time complexity is still  $O(Nn^2)$ , the same as AODE. That means WAODE is almost as efficient as AODE. Our experiments also verify this fact. The classification time complexity of WAODE is  $O(kn^2)$ , the same as AODE too.

### 3 Experimental Methodology and Results

We ran our experiments on 36 UCI data sets[3] selected by Weka[2], which represent a wide range of domains and data characteristics. In our experiments, we adopted the following three preprocessing steps.

1. Missing values: We used the unsupervised filter *ReplaceMissingValues* in Weka to replace the missing values in each data set.
2. Discretization of numeric attributes: We used the unsupervised 10-bin filter *Discretize* in Weka to discretize all the numeric attributes.
3. Removal of useless attributes: Apparently, if the number of values of an attribute is almost equal to the number of instances in a data set, it does not contribute useful information to classification. Thus, we removed this type of attributes using the unsupervised filter *Remove* in Weka.

We conduct our experiments to compare WAODE measured by classification accuracy with NB, SBC[4], TAN[5], NBTree[6], and AODE[1]. In our all experiments, the classification accuracy of each classifier on each data set was obtained via 10 runs of ten-fold cross validation. Runs with the various classifiers were carried out on the same training sets and evaluated on the same test sets.

**Table 1.** Summary of experimental results: accuracy comparisons

	NB	SBC	TAN	NBTree	AODE
SBC	11/23/2				
TAN	12/20/4	5/24/7			
NBTree	12/24/0	8/28/0	7/27/2		
AODE	13/22/1	11/23/2	10/23/3	5/26/5	
WAODE	17/18/1	12/23/1	13/23/0	10/24/2	9/27/0

Table 1 shows the compared results of two-tailed t-test with a 95% confidence level between each pair of algorithms, each entry  $w/t/l$  in Table 1 means that the classifier at the corresponding row wins in  $w$  data sets, ties in  $t$  data sets, and loses in  $l$  data sets, compared to the classifier at the corresponding column. Our experiments show that WAODE significantly outperforms all the other algorithms used to compare.

## 4 Conclusions and Future Work

In this paper, we propose an improved classification model called Weightily Averaged One-Dependence Estimators by modifying most recent work on improving naive Bayes called Averaged One-Dependence Estimators. Basically, WAODE assigns different weights to different tree augmented naive Bayes in the aggregate of AODE according to the mutual information between the root attribute of the tree and the class. Our experiments show that WAODE significantly outperforms all the other algorithms used to compare. In addition, WAODE almost has the same time complexity as AODE. Thus, WAODE could be a good classification algorithm for various data mining applications.

In defining and learning an WAODE, how to learn the weights is crucial. Currently, we use mutual information to estimate the weights directly from data, which is efficient but quite simple. We believe that the use of more sophisticated methods to determine the weights could further improve the performance of the current WAODE and make its advantage stronger. This is one direction for our future research.

## References

1. G. I. Webb, J. Boughton, and Z. Wang. Not so naive bayes: Aggregating one-dependence estimators. *Machine Learning*, Vol. 58, 5-24, 2005.
2. Witten, I. H., Frank, E.: data mining-Practical Machine Learning Tools and Techniques with Java Implementation. Morgan Kaufmann (2000). <http://prdownloads.sourceforge.net/weka/datasets-UCI.jar>
3. Merz,C., Murphy,P. and Aha,D., UCI repository of machine learning databases. In Dept of ICS, University of California, Irvine. <http://www.ics.uci.edu/mlearn/MLRepository.html>, 1997.
4. Langley, P., Sage, S. Induction of selective Bayesian classifiers. in Proceedings of the Tenth Conference on Uncertainty in Artificial Intelligence, 1994, pp. 339-406.
5. Friedman, Geiger, and Goldszmidt. "Bayesian Network Classifiers", *Machine Learning*, Vol. 29, 131-163, 1997.
6. Kohavi, R.: Scaling Up the Accuracy of Naive-Bayes Classifiers: A Decision-Tree Hybrid. *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96)*. AAAI Press (1996) 202-207
7. Pearl, J. (1988). Probabilistic Reasoning in Intelligent Systems. San Francisco, CA: Morgan Kaufmann.
8. Chickering, D. M. (1996). Learning Bayesian networks is NP-Complete. In Fisher, D. and Lenz, H., editors, *Learning from Data: Artificial Intelligence and Statistics V*, pages 121-130. Springer-Verlag.

# SV-kNNC: An Algorithm for Improving the Efficiency of k-Nearest Neighbor

Anantaporn Srisawat, Tanasanee Phienthrakul, and Boonserm Kijisirikul

Department of Computer Engineering, Faculty of Engineering, Chulalongkorn University,  
Bangkok 10330 Thailand  
anantaporn.s@student.chula.ac.th, tanasanee@yahoo.com,  
boonserm.k@chula.ac.th

**Abstract.** This paper proposes SV-kNNC, a new algorithm for k-Nearest Neighbor (kNN). This algorithm consists of three steps. First, Support Vector Machines (SVMs) are applied to select some important training data. Then, k-mean clustering is used to assign the weight to each training instance. Finally, unseen examples are classified by kNN. Fourteen datasets from the UCI repository were used to evaluate the performance of this algorithm. SV-kNNC is compared with conventional kNN and kNN with two instance reduction techniques: CNN and ENN. The results show that our algorithm provides the best performance, both predictive accuracy and classification time.

## 1 Introduction

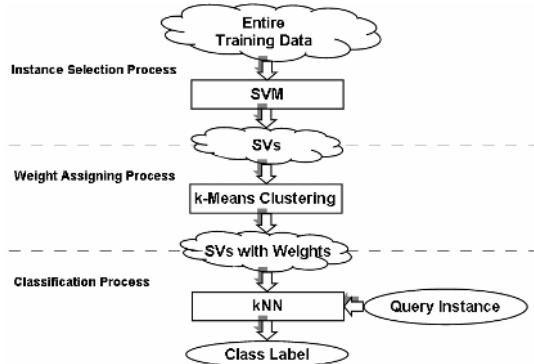
In supervised learning, kNN is one of the most popular choices due to its simplicity [1]. The advantages of kNN include the ability to model complex target functions by a collection of less complex local approximation. Moreover, information presented in the training examples is never lost [2].

However, the main practical difficulties of kNN are the amounts and characteristics of training data. kNN methods are based on a distance function for all pairs of a new query instance and training data. If there are a lot of training data, kNN will take a lot of time in the classification process. In addition, the training data have strong influence on the target output. Noisy data may also reduce the performance of kNN. Hence, if unimportant data and noisy data are eliminated, both classification time and error should be reduced.

This paper proposes SV-kNNC that applies SVM [3] and k-mean clustering [4] to improve the efficiency of kNN. In this approach, SVM are used to reduce the training data of kNN for improving the classification time. Moreover, the accuracy of kNN can also be improved by using k-means clustering to assign weight for each training data.

## 2 SV-kNNC

In this section, we propose SV-kNNC that consists of three processes: instance selection process, weight assigning process, and classification process. The model of SV-kNNC is illustrated in Fig. 1.

**Fig. 1.** SV-kNNC Model

## 2.1 Instance Selection

A main difficulty of kNN is the classification time that increases with the number of training data. If unimportant training data are discarded, the classification time can be reduced. Conventional data reduction methods such as CNN [5], IB3, DROP [6, 7], and ENN [8] gradually add or remove training data. Since these methods do not consider all training data at the same time, they may not yield the actual optimal training set.

We notice that the support vector machine uses only support vectors (SVs), examples that are closest to the hyperplane, to classify unseen data. Therefore, the obtained SVs are representative training data that should be sufficient to represent all training data.

For instance selection process, entire training data are fed into SVM and a set of SVs is produced as the output of SVM training. These SVs are then stored in memory. This process also can eliminate redundant instances of each class in the feature space.

## 2.2 Weight Assigning

While SVs are good representative data for each class in the feature space, some SVs may not be suitable as training data for kNN. This is because the distance between instances is changed when these SVs are retransformed to the input space. The class label of an instance may differ from its surrounding nearest neighbor instances. These instances may be the cause of misclassification.

To avoid this problem, re-checking the contribution of SVs to the classification in the input space before using them to classify a query instance is considered. Weight is then assigned to determine the contribution of each SV. The instance which has a higher weight is more credible, and thus it will have more impact on kNN classification.

There are 2 steps in weight assigning process. First, the k-means algorithm is used to partition all SVs into pre-defined  $k$  clusters. If there are more than one class label appearing in one cluster, the instances with the majority class will be more credible than the instances with the minority classes. After data are clustered, the class labels of all SVs are considered. In this step, each instance in a cluster is assigned weight

according to the proportion of the instance class labels in that cluster. This weight of sample  $x_i$ , denoted as  $w_i$ , is shown in equation (1).

$$w_i = \frac{n(\text{class}(x_i))}{\text{Total}} \quad (1)$$

where  $i=1,\dots,m$ , for  $m$  is the number of all SVs.  $n(\text{class}(x_i))$  is the number of samples in the cluster that have the same class with the sample  $x_i$ .  $\text{Total}$  is the number of total samples in the cluster.

### 2.3 Classification

In classification process, unseen instances are classified by the kNN algorithm. Only weighted SVs, which are calculated in the previous process, are used as training data of kNN. When a new query instance is entered, kNN finds a set of  $k$  nearest instances from a set of SVs. Weights of SVs with the same class label are summed up, and the class label with the maximum weight is produced as the class label of the query instance.

## 3 Experiments and Results

In order to verify the performance of our approach, fourteen datasets from the UCI repository [9] are tested using 5-folds cross validation. Each dataset contains two classes. When SVM, CNN, and ENN were run, the training data were selected differently by these algorithms. Percentages of reduced data are showed in Table 1. We found that SV-kNNC has the highest ability on data reduction. It reduced data more than CNN and ENN on 6 datasets. Moreover, SV-kNNC reduced training examples up to 71.67% on the BreastCancer dataset.

**Table 1.** Percentage of reduced data

Datasets	CNN	ENN	SV-kNNC
Checkers	9.11	9.38	<b>10.16</b>
Spiral	<b>7.65</b>	0.00	<b>14.43</b>
LiverDisorders	7.32	<b>17.61</b>	14.71
IndiansDiabetes	10.09	<b>14.75</b>	11.46
ThreeOfNine	<b>29.64</b>	10.74	0.00
TicTacToe	<b>35.49</b>	0.00	1.46
BreastCancer	66.24	1.86	<b>71.67</b>
ParityBits	8.76	<b>12.04</b>	0.61
ClevelandHeart	17.69	10.00	<b>24.63</b>
Australian	<b>25.36</b>	0.25	15.76
Rand	<b>10.58</b>	0.27	2.77
German-org	<b>13.80</b>	9.95	8.03
Ionosphere	22.65	4.70	<b>49.79</b>
Sonar	15.50	0.96	<b>22.48</b>

In Table 2, the accuracies of SV-kNNC are compared with the conventional kNN and kNN with two instance reduction techniques: CNN and ENN. These results show that the accuracies of SV-kNNC are statistically significantly higher than the

conventional kNN with significance level better than 0.05 on all datasets except for datasets *Spiral* and *ThreeOfNine*. The SV-kNNC also provides the best accuracies on all datasets when compared with the other data reduction techniques, i.e. CNN and ENN.

**Table 2.** Comparison of accuracies on various algorithms

Datasets	kNN	CNN+kNN	ENN+kNN	SV-kNNC
Checkers	88.00	85.94	87.50	<b>93.23 ***</b>
Spiral	<b>100.00</b>	98.46	<b>100.00</b>	<b>100.00</b>
LiverDisorders	66.09	65.51	66.67	<b>74.20 ***</b>
IndiansDiabetes	76.82	76.95	77.21	<b>79.03 ***</b>
ThreeOfNine	<b>100.00</b>	96.48	92.57	<b>100.00</b>
TicTacToe	97.29	93.63	97.29	<b>99.27 ***</b>
BreastCancer	97.14	96.57	96.85	<b>98.14 *</b>
ParityBits	53.90	56.15	49.70	<b>68.75 ***</b>
ClevelandHeart	84.44	84.81	83.70	<b>87.78 ***</b>
Australian	87.97	87.39	87.97	<b>90.00 ***</b>
Rand	51.83	50.67	51.73	<b>54.93 ***</b>
German-org	75.90	74.90	75.00	<b>76.90 **</b>
Ionosphere	86.33	84.36	85.77	<b>90.04 ***</b>
Sonar	85.56	82.68	85.09	<b>89.87 ***</b>
Average	82.23	81.04	81.22	<b>85.87</b>

Statistical significance for the difference from kNN at level: \* 0.05, \*\* 0.025, \*\*\* 0.01

From the experimental results, we found that SV-kNNC enhances the performance of kNN, both classification time and predictive accuracy. Furthermore, the data reduction ability of SV-kNNC is better than CNN and ENN. This is because CNN and ENN keep or remove an instance depending on its nearest instances. Besides, both CNN and ENN are especially sensitive to noise, and noisy instances may be retained. Hence, there are some instances that are still misclassified by CNN and ENN. On the other hand, SVM selects a set of good representative examples of each class by inspecting whole instances at the same time. Thus, some noisy and redundant data may be removed.

## 4 Conclusions

This paper proposes an approach for data reduction to enhance performance of kNN, called SV-kNNC. This algorithm is divided into three steps. First, SVs from SVM learning process are used to be the training examples. Then, these SVs are assigned weights by the ratio of each class label on a cluster from k-mean clustering. Finally, SVs with weights are used to classify the query instances by kNN classification process.

The experimental results showed that SV-kNNC has the ability to reduce data (more than 70% in some dataset). Thus, the classification time of SV-kNNC is less than kNN. In addition, the accuracy of SV-kNNC is better than the conventional kNN, and kNN with two instance reduction techniques (CNN and ENN) on the UCI benchmarks.

SV-kNNC achieves the best performance because training data are analyzed twice before classification process. First, SVM eliminates redundant data and selects the instances that are nearest to a decision surface in the feature space. Then, these SVs

are re-checked and are assigned weights. Therefore, the obtained training data are more effective than the conventional techniques.

## Acknowledgement

The authors acknowledge the financial support provided by the Thailand Research Fund and the Royal Golden Jubilee Ph.D. Program.

## References

1. Cover, T., Hart, P.: Nearest Neighbor Pattern Classification. *IEEE Transactions on Information Theory*. 13 (1967) 21-27
2. Mitchell, T.M.: Machine Learning. The McGraw-Hill, New York (1997)
3. Vapnik, V.N.: Statistical Learning Theory. John Wiley and Sons, New York (1998)
4. MacQueen, J.B.: Some Methods for classification and Analysis of Multivariate Observations. *Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability*, Berkeley, University of California Press, 1 (1967) 281-297
5. Hart, P.E.: The Condensed Nearest Neighbor Rule. *IEEE Transactions on Information Theory*. 14 (1968) 515-516
6. Wilson, D.R., Martinez, T.R.: Reduction Techniques for Instance-Based Learning Algorithms. *Machine Learning*, 38 (2000) 257-286
7. Jankowski, N., Grochowski, M.: Comparison of Instance Selection Algorithms I. Algorithms Survey. *ICAISC* (2004) 598-603
8. Wilson, D.L.: Asymptotic Properties of Nearest Neighbor Rules using Edited Data. *IEEE Transactions on Systems, Man, and Cybernetics*. 2 (1972) 408-421
9. Blake, C.L., Merz, C.J.: UCI Repository of machine learning databases [<http://www.ics.uci.edu/~mlearn/MLRepository.html>]. Irvine, CA: University of California, Department of Information and Computer Science (1998)

# A Novel Support Vector Machine Metamodel for Business Risk Identification

Kin Keung Lai<sup>1,2</sup>, Lean Yu<sup>2,4</sup>, Wei Huang<sup>3</sup>, and Shouyang Wang<sup>1,4</sup>

<sup>1</sup> College of Business Administration, Hunan University, Changsha 410082, China

<sup>2</sup> Department of Management Sciences, City University of Hong Kong,

Tat Chee Avenue, Kowloon, Hong Kong

{msyulean, mskklai}@cityu.edu.hk

<sup>3</sup> School of Management, Huazhong University of Science and Technology,

1037 Luoyu Road, Wuhan 430074, China

<sup>4</sup> Institute of Systems Science, Academy of Mathematics and Systems Science,

Chinese Academy of Sciences, Beijing 100080, China

{yulean, sywang}@amss.ac.cn

**Abstract.** In this study, support vector machine (SVM) is used as a metamodeling technique to design a business risk identification system. First of all, a bagging sampling technique is used to generate different training sets. Based on the different training sets, different SVM models with different parameters, i.e., base models, are then trained to formulate different classifiers. Finally, a SVM-based metamodel (i.e., metaclassifier) can be produced by learning from all base models. For illustration the proposed metamodel is applied to a real-world business insolvency risk classification problem.

## 1 Introduction

Business risk identification is an important and critical step in business risk management. Almost all financial organizations, such as banks and credit institutions, need this kind of information in order to enable them to take either preventive or corrective action. Therefore it is important to identify possible business risk.

Originally, identifying business risk started with empirical methods (e.g., the “three A” method, the “five C” method, and the “credit-men” method) proposed by large banks in USA. Then, the financial ratios methodology [1] was developed for business risk identification. Subsequently, logit regression [2] and probit analysis model [3] are also utilized to improve the classification accuracy. However, the generalization of these methods resulted in several critical studies, see [2, 4] for more details.

Recently, some emerging intelligent methods such as artificial neural networks (ANN) [5] and support vector machines (SVM) [6] have been developed to identify business risk. However, these unitary intelligent techniques are not perfect. For example, neural networks are easy to trap into local minima and SVM may suffer from overfitting in some situations [7]. In order to overcome these drawbacks, a novel metamodeling technique is introduced. In this study, SVM is used as a metamodeling tool. The generic idea consists of three phases. First, an initial data set is transformed into several different training sets. Based on the different training sets, different SVM

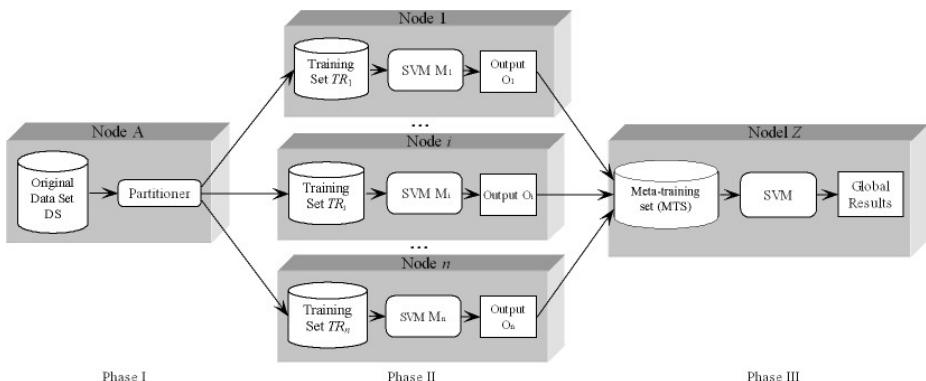
models, i.e., base models, are then trained to formulate different classifiers. Finally, a SVM-based metamodel (i.e., metaclassifier) can be produced by learning from all base models. In this sense, metamodel is actually a model of a model.

The rest of this study is organized as follows. In the next section, a triple-phase SVM-based metamodeling process is described in detail. In Section 3, a business risk identification experiment is performed for illustration. Section 4 concludes the paper.

## 2 The SVM-Based Metamodelling Process

SVM was originally introduced by Vapnik [8] in the 1990s. The basic idea of SVM is to use linear model to implement nonlinear class boundaries through some nonlinear mapping the input vector into the high-dimensional feature space. In this study, we use SVM for pattern classification, i.e., support vector classification (SVC) [8].

Although the SVM has proven to be a very suitable tool for classification problem, but later studies also found overfitting problem in SVM [7]. That is, the results of SVM are unstable. To overcome the limitation, a SVM-based metamodel is proposed. In the proposed metamodel, original data sets are firstly used to generate different training sets, and then these training sets are trained with different SVM models. Finally, the results from different SVM models are collected to formulate a meta-training set; another SVM model is used as a combiner to produce a global model or a metamodel. Fig. 1 shows a generic metamodeling process, in which a metamodel is obtained on *Node Z*, starting from the original data set *DS* stored on *Node A*.



**Fig. 1.** The generic SVM-based metamodeling process

As can be seen from Fig. 1, the generic metamodeling process consists of three phases, which can be described as follows.

**Phase 1:** on *Node A*, original data set *DS* are partitioned into some different training subsets  $TR_1, TR_2, \dots, TR_n$  via certain partitioning algorithm. Then  $TR_1, TR_2, \dots, TR_n$  are moved from *Node A* to *Node 1, Node 2, ..., Node n*.

**Phase 2:** on each *Node i* ( $i = 1, 2, \dots, n$ ) the SVM models  $M_i$  with different parameters is deployed different nodes and is trained with different  $TR_i$ . Then the output of each  $M_i$  is moved from *Node i* to *Node Z*.

**Phase 3:** on *Node Z*, the output of the different models  $M_1, M_2, \dots, M_n$  can form a new training set namely meta-training set *MTS*. Based on the *MTS*, another SVM model is trained and thus global results can be obtained.

## 2.1 Original Data Partition

Data partition is the most important step in designing a metamodel. In this study, the bootstrap aggregating (bagging) proposed by Breiman [9] is used. Bagging [9] is a widely used data sampling method in the machine learning. Given that the size of the original data set *DS* is  $P$ , the size of new training data is  $N$ , and the number of new training data items is  $m$ , the bagging algorithm can be shown in Fig. 2.

```

Input: original data set DS
Output: The generated new training subsets ( $TR_1, TR_2, \dots, TR_m$ )
For  $t = 1$  to  $m$ 
    For  $i = 1$  to  $N$ 
         $RandRow = P * rand()$ 
        If  $RandRow \leq P$ 
             $P_t(i, AllColumns) = DS(RandRow, AllColumns)$ 
        End If
        Next  $i$ 
    Next  $t$ 
Output the final training subsets ( $TR_1, TR_2, \dots, TR_m$ )

```

Fig. 2. The bagging algorithm

## 2.2 Unitary Model Generation

With the work about bias-variance trade-off [10], a metamodel consisting of diverse models (i.e., base models) with much disagreement is more likely to have a good performance. Therefore, how to generate the diverse model is the key to an effective metamodel. For the SVM model, there are several methods for generating diverse models: (1) using different kernel functions, such as polynomial function and Gaussian function; (2) varying the SVM model parameters, such as margin parameter  $C$  and kernel parameter  $\sigma^2$ ; (3) utilizing different training data sets. This method is done by the first phase. In this study, the individual SVM models with different training data are therefore generated as base models  $M_1, M_2, \dots, M_n$ , as illustrated in Fig. 1.

## 2.3 Metamodel Formulation

As Fig. 1 illustrated, the initial data set is first divided into different training sets, and then these training sets are input to the different individual SVM models which could be executed concurrently. These individual models are called “base models”. In this phase, the main task is to generate a metamodel that explores the relationship between base models. In this study, another SVM is used to perform this task to generate a

metamodel. That is, we use a SVM model to learn the relationship between based models by taking the outputs of all base models as input. By training and learning, a metamodel can be formulated.

### 3 Experiment Study

The data used in this study is about UK firms from the Financial Analysis Made Easy (FAME) database which can be found in the Appendix of [16]. It contains 30 failed and 30 non-failed firms with 12 variables. In this empirical test, 40 firms are randomly drawn as the training sample. Due to the sample scarcity, we make the number of good firms equal to the number of bad firms in both the training and testing samples, so as to avoid the embarrassing situations that just two or three good (or bad, equally likely) firms in the testing sample. Thus the training sample includes 20 data of each class. Its aim is to minimize the effect of such factors as industry or size that in some cases can be very important. Except from the above learning sample, the testing sample was collected using a similar approach. The testing sample consists of 10 failed and 10 non-failed firms, which is used to test the effectiveness of metamodel.

For constructing a SVM-based metamodel, 20 training sets are generated by bagging algorithm. For comparison, several commonly used models, such as individual SVM model, artificial neural network (ANN) [5], linear discriminant analysis (LDA) [1] and logit analysis [2], are used. Finally, the identification performance is evaluated by hit ratio. Accordingly, the best results of different models are reported in Table 1.

**Table 1.** The performance comparison of different models (hit ratio: %)

Data	Metamodel	SVM	ANN	LDA	Logit
Training data	98.86	96.12	89.63	74.36	76.69
Testing data	93.45	90.69	82.56	70.25	72.34

From Table 1, the SVM-based metamodel outperforms the individual SVM, ANN, LDA and Logit in terms of both training data and testing data. By McNemar test [12], the SVM-based metamodel outperforms individual SVM and ANN model at 10% significant level and LDA and Logit at 5% statistical significance level, respectively.

### 4 Conclusions

In this study, we have developed a triple-phase SVM-based metamodeling system for business risk identification. Through the practical data experiment, we have obtained good classification results and meantime demonstrated the SVM-based metamodel outperforms all the benchmark models. These advantages imply that the proposed SVM-based metamodeling technique can provide a promising solution to business insolvency risk identification problem.

## Acknowledgements

This work is partially supported by National Natural Science Foundation of China; Chinese Academy of Sciences and Strategic Research Grant of City University of Hong Kong (SRG No. 7001806).

## References

1. Altman, E.I.: *Corporate Financial Distress and Bankruptcy*. John Wiley, New York (1993)
2. Ohlson, J.A.: Financial ratios and the probabilistic prediction of bankruptcy. *Journal of Accounting Research* 3 (1980) 109-131
3. Zmijewski, M.E.: Methodological issues related to the estimation of financial distress prediction models. *Journal of Accounting Research* (1984) 59-82
4. Eisenbeis, R.A.: Pitfalls in the application of discriminant analysis in business and economics. *The Journal of Finance* 32 (1977) 875-900
5. Lee, K., Booth, D., Alam, P.: A Comparison of Supervised and Unsupervised Neural Networks in Predicting Bankruptcy of Korean Firms. *Expert Systems with Applications* 29 (2005) 1-16
6. Shin, K.S., Lee, T.S., Kim, H.J.: An Application of Support Vector Machines in Bankruptcy Prediction Model. *Expert Systems with Applications* 28 (2005) 127-135
7. Tay, F.E.H., Cao, L.J.: Application of support vector machines in financial time series forecasting. *Omega* 29 (2001) 309-317
8. Vapnik, V.N.: *The Nature of Statistical Learning Theory*. Springer, New York (1995)
9. Breiman, L. Bagging Predictors. *Machine Learning* 26 (1996) 123-140
10. Lai, K.K., Yu, L., Wang, S.Y., Huang, W.: A bias-variance-complexity trade-off framework for complex system modeling. *Lecture Notes in Computer Science* 3980 (2006) 518-527
11. Beynon, M.J., Peel, M.J.: Variable precision rough set theory and data discretisation: an application to corporate failure prediction. *Omega* 29 (2001) 561-576
12. Cooper, D.R., Emory, C.W.: *Business Research Methods*. Irwin, Chicago (1995)

# Performing Locally Linear Embedding with Adaptable Neighborhood Size on Manifold

Guihua Wen<sup>1,3</sup>, Lijun Jiang<sup>1</sup>, Jun Wen<sup>2</sup>, and Nigel R. Shadbolt<sup>3</sup>

<sup>1</sup> South China University of Technology, Guangzhou 510641, China  
crghwen@scut.edu.cn, cssturat@sohu.com

<sup>2</sup> Hubei Institute for Nationalities, HuBei Ensi 445000, China  
wenj\_64@163.com

<sup>3</sup> University of Southampton, Southampton SO17 1BJ, United Kingdom  
nrs@ecs.soton.ac.uk

**Abstract.** Locally linear embedding approach (LLE) is one of most efficient nonlinear dimensionality reduction approaches with good representational capacity for a broader range of manifolds and high computational efficiency. However, LLE and its variants are based on the assumption that the whole data manifold is evenly distributed so that they fail to nicely deal with most real problems that are unevenly distributed. This paper first proposes an approach to judge whether the manifold is even or not, and then logically divides the unevenly distributed manifold into many evenly distributed sub-manifolds, where the neighbourhood size for each sub-manifold is automatically determined based on its structure. It is proved, by visualization and classification experiments on benchmark data sets, that our approach is competitive.

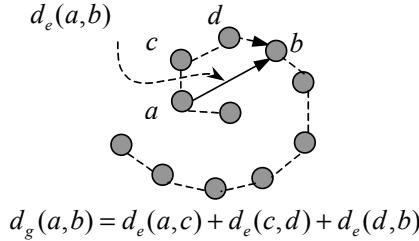
## 1 Introduction

LLE is very efficient to perform dimensionality reduction on nonlinear data manifolds[1]. It tries to find an optimal reconstruction of each point from a linear composition of its neighbors and then applies the same reconstruction to generate the low-dimensional configuration. Accordingly, LLE is computationally simpler and can give useful results on a broader range of manifolds [2]. LLE has many variants, such as Laplacian eigenmaps [3] and Hessian eigenmaps(HLLE) [4], incremental LLE [7], supervised LLE [10], integrated LLE with classic PCA or LDA [8], integrated LLE with SOM [11] etc. These approaches determine the neighborhood for all points with the same neighborhood size so that they often fail to nicely deal with unevenly distributed manifolds. To solve this problem, this paper proposes an improved LLE with adaptable neighborhood size for each point on manifold based on the structure of the manifold.

## 2 Proposed Approach

LLE largely depends on the neighborhood structure. In presence of noise or when the data is sparsely sampled, it has to take the small neighborhood size to

avoid generating short-circuit edges [6,12]. Simultaneously, in terms of principle of LLE, the neighborhood size should be as large as possible as long as those determined neighbors lie on the same local linear subspace. This puts LLE in a dilemma when dealing with unevenly distributed manifolds. It needs to be improved by varying the number of neighbors per point based on the structure of manifold so as to give consistently faithful embedding.



**Fig. 1.** Difference between Euclidean distance and geodesic distance

For sufficiently dense sampling, geodesic distance can be estimated approximately from an direct Euclidean distance graph connecting all local neighborhoods of points [5]. As shown as Figure 1, when the manifold is curved, the geodesic distance is not equal to Euclidean distance. Inspired by stress formula of Sammon, we define the curvature stress of the manifold as follows

$$\lambda_s(M) = \sum_{1 < j}^N \frac{(d_e(X_i, X_j) - d_g(X_i, X_j))^2}{d_g(X_i, X_j)} / \sum_{1 < j}^N d_g(X_i, X_j)$$

The  $\lambda_s(M)$  reflect the intrinsic structure of the manifold, by which we can logically divide the whole manifold into disjoint sub-manifolds and then assign different neighborhood size for each submanifold. Suppose that the manifold  $M$  is divided into two logic equal parts  $M_1$  and  $M_2$ , the neigborhood sizes  $k_1$  and  $k_2$  for each submanifold can be determined as follows:

$$k_1 = k \times \lambda_s(M_2) / (\lambda_s(M_1) + \lambda_s(M_2)) \quad (1)$$

$$k_2 = k \times \lambda_s(M_1) / (\lambda_s(M_1) + \lambda_s(M_2)) \quad (2)$$

Generally for unevenly distributed manifold,  $k_1 \neq k_2$ . Because  $k$  is a parameter, to align  $k_1$  and  $k_2$  with the  $k$ , we perform the following linear transformation:

$$k_1 = k_1 + (k - \max(k_1, k_2)) \quad (3)$$

$$k_2 = k_2 + (k - \max(k_1, k_2)) \quad (4)$$

Similarly, we can logically decompose the sub-manifold  $M_1$  and  $M_2$  recursively so that the neighborhood size for each sub-manifold can be calculated. Based on this idea, a new algorithm VKLLE is proposed to improve the LLE.

**Algorithm.** VKLLE( $M, k_g, k, m$ )

**Input parameters:**  $M = \{X_i \in R^n\}$  is the high dimensional data with  $N$  points,  $k_g$  is for the geodesic distance,  $k$  is the neighborhood size for LLE, and  $m$  is the number of sub-manifolds of  $M$  to be divided logically. **Output:** it gives  $N$  points  $Y_i \in R^d$ , where  $d < n..$

**Step 1.** Logically divide  $M$  into  $m$  sub-manifolds and then determine the neighborhood size  $k_i$  for each sub-manifold  $M_i (1 \leq i \leq m)$ .

**Step 2.** Suppose  $X_i \in M_i$ , its neighborhood  $N(X_i)$  consists of  $k_i$  nearest neighbors from  $M$  in terms of Euclidean distance.

**Step 3.** The following function is minimized to calculate the optimal weights  $W_{ij}$  for the reconstruction of  $X_i$ , where each  $W_{ij}$  stands for the contribution of the  $X_j$  to the reconstruction of  $X_i$  and the constraints are  $\sum_{X_{ij} \in N(X_i)} W_{ij} = 1$ .

$$\varepsilon = \sum_{i=1}^N \|X_i - \sum_{X_{ij} \in N(X_i)} W_{ij} X_{ij}\|^2 \quad (5)$$

**Step 4.** The weights  $W_{ij}$  are fixed and the following cost function is minimized to generate the low-dimensional embedding:

$$\varepsilon = \sum_{i=1}^N \|Y_i - \sum_{Y_{ij} \in N(Y_i)} W_{ij} Y_{ij}\|^2 \quad (6)$$

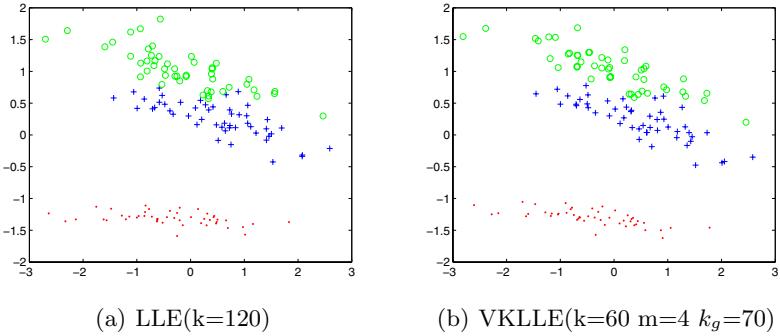
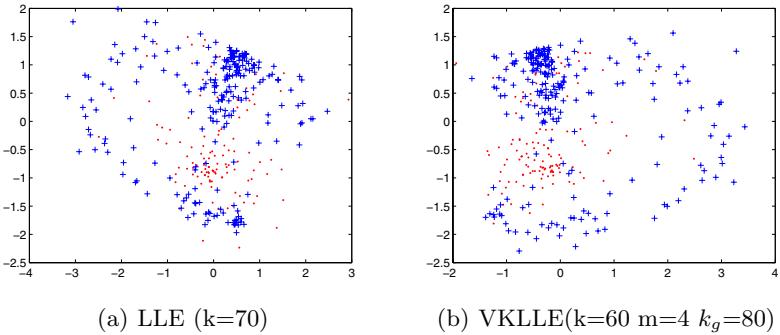
subject to two constraints:  $\sum_{j=1}^N Y_j = 0$  and  $\sum_{j=1}^N Y_j Y_j^T = I$  where  $I$  is the  $d \times d$  identity matrix.

### 3 Experimental Results

Dimensionality reduction can be applied to remove the irrelevant noise but keep necessary information for classification so that classification accuracy can be improved[9]. We did experiments on 6 benchmark data sets from UCI Machine Learning Repository listed as Table 1(Left). Each data set is projected onto its

**Table 1.** Data sets and Classification error of two approaches

Dataset	Size	Attributes	Classes	LLE	VKLLE
Dematology	358	34	6	0.0417	0.0250
Diabetes	768	8	2	0.2109	0.2109
Iris	150	4	3	0.0612	0.0000
Glass	214	9	6	0.2817	0.2676
Segmentation	210	19	7	0.1000	0.0857
Ionosphere	351	34	2	0.1167	0.0833

**Fig. 2.** Visualization on Iris dataset**Fig. 3.** Visualization on Ionosphere

half dimensional space and is then split into the training set (70%) and test set(30%) for KNN classifier, where parameter  $k=10\dots50$ ,  $m=2\dots8$ ,  $k_g=2\dots8$ , and another parameter for KNN classifier is selected from 3...40. It can be seen from Table 1(Right) that VKLLE averagely outperforms LLE by 2.33%. The maximum gap on data set no.3 is up to 6.12%. Except on data set no.2, VKLLE performs better than LLE on all data sets.

Visualization experiments are performed on Iris and Ionosphere where  $k=10\dots100$  with step 10,  $m = 1\dots10$ , and  $k_g=10\dots100$  with step 10. It can be observed from Figure 2 that VKLLE performs better on Iris data set than LLE, as it has not only preserved the inter-cluster structures better, for example, it separates the class denoted by symbol + from that denoted by symbol O better, but also captured more details of intra-cluster and inter-point distributions. Similarly, from Figure 3 on Isonosphere, it can be concluded that VKLLE also outperforms LLE, as it reduces the overlap among points from two different classes. This can be obviously observed when the pictures are enlarged.

## 4 Conclusion

This paper presents a new locally linear embedding approach to deal with unevenly distributed manifolds. It is validated by visualization and classification experiments on benchmark data sets. One of shortcomings of this approach is that computation of geodesic distance is required. To avoid this, we are working on finding alternative ways.

## Acknowledgement

This work is supported by Natural Science Foundation of China(60003019), Hubei Science and Technology Project (2005AA101C17),and UK Engineering and Physical Sciences Research Council under grant number GR/N15764/01.

## References

1. S.T. Roweis and L.K. Saul: Nonlinear Dimensionality Reduction by Locally Linear Embedding. *Science*, **290**(2000)2323–2326
2. L. K. Saul and S.T.Roweis: Think Globally, Fit Locally: Unsupervised Learning of Low Dimensional Manifolds. *Journal of Machine Learning Research*, **4**(2003)119-155
3. M. Belkin and P. Niyogi: Laplacian Eigenmaps for Dimensionality Reduction and Data Representation. *Neural Computing*, **15**(2003)1373–1396
4. D.L. Donoho and C. Grimes: Hessian eigenmaps: Locally linear embedding, techniques for high-dimensional data. *Proc.Natl.Acad. Sci. U. S. A.*, **100**(2003)5591-5596
5. J.B. Tenenbaum and V. de Silva and J.C. Langford: A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science*,**290**(2000)2319–2323
6. M. Balasubramanian and E.L. Schwartz: The Isomap Algorithm and Topological Stability. *Science*,**295**(2002)7–7
7. Kouropteva, Olga, Okun, Oleg, Pietikainen, Matti: Incremental locally linear embedding. *Pattern Recognition*,**38**(2005) 1764-1767
8. Abusham EE, Ngo D, Teoh A: Fusion of locally linear embedding and principal component analysis for face recognition (FLLEPCA). *Lecture Notes in Computer Science*,**3687**(2005) 326-333
9. X.Geng and D.C.Zhan and Z.H.Zhou: Supervised Nonlinear Dimensionality Reduction for Visualization and Classification. *IEEE Transactions on Systems, Man and Cybernetics*, **35**(2005)1098–1107
10. de Ridder D, Kouropteva O, Okun O, et al.: Supervised locally linear embedding. *Lecture Notes in Artificial Intelligence*,**2714**(2003) 333-341
11. Xiao J, Zhou ZT, Hu DW, et al. Self-organized locally linear embedding for nonlinear dimensionality reduction. *Lecture Notes in Computer Science*,**3610**(2005) 101-109
12. Yang L: Building k edge-disjoint spanning trees of minimum total length for isometric data embedding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **27**(2005) 1680-1683

# Stroke Number and Order Free Handwriting Recognition for Nepali

Santosh K.C.<sup>1</sup> and Cholwich Nattee<sup>2</sup>

School of Information and Computer Technology,  
Sirindhorn International Institute of Technology, Thammasat University,  
131 Moo 5 Tiwanont Rd., Bangkadi, Muang,  
PathumThani 12000, Thailand  
[santosh@siit.tu.ac.th](mailto:santosh@siit.tu.ac.th), [cholwich@siit.tu.ac.th](mailto:cholwich@siit.tu.ac.th)

**Abstract.** This paper utilizes *structural* properties of those alphanumeric characters, which have variable writing units. Writing units reveal number, shape, size, order of stroke, and speed in writing. It uses a string of pen tip's positions and tangent angles of every consecutive point as a feature vector sequence of a stroke. We constructed a prototype recognizer that uses the “Dynamic Time Warping” (DTW) algorithm to align handwritten strokes with stored stroke templates and determine their similarity. Separate system is trained for original and preprocessed writing samples and achieved recognition rates of 85.87% and 88.59% respectively. This introduces novel real time handwriting recognition on Nepalese alphanumeric characters, which are independent of number of strokes, as well as their order.

## 1 Introduction

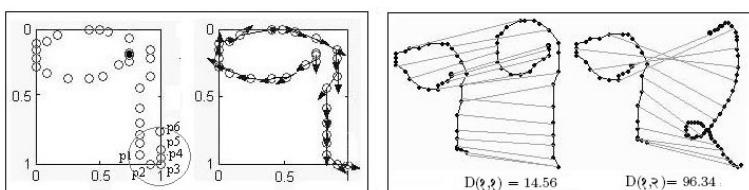
Devanagari script is originated from Sanskrit. Script used in Nepali is Devanagari. In Nepali, there are 33 consonants (vyanjan) with half forms, 13 vowels (svar), 16 modifiers and 10 numerals, from which common letters are used in this task. Handwriting analysis and recognition have received extensive attention in both the academic and commercial fields. However, Information Technology has not reached this demand on local languages, like Nepali. Connell et al, [3] focused on both local and global features, where a number of classifiers are used in addition to use traditional Hidden Markov Model. Dynamic Programming has been enjoyed advantages from the last few decades and has a wide range of applications from speech to character recognition [2]. Very few works have been done on online handwriting recognition in Devanagari. However, some efforts on the recognition of Devanagari script [1], [4] have done. To the best our knowledge, this is the first time real-time stroke number and order free Nepalese handwriting recognition system is being reported. In Nepali, many classes of alphanumeric characters are similar to each other, that make confusions in recognition. In addition, number, order, shape and size, direction of stroke and speed are varied from time to time writing even within the same user and same class of character, but the writing samples are readable. This is why this script is different from

western scripts and is often called cursive script. The recognition system, which consists of collection of temporal information, pre-processing, feature extraction and DTW based agglomerative hierarchical clustering is explained in the following sections. The proposed recognizers yield competitive, encouraging and analogical results.

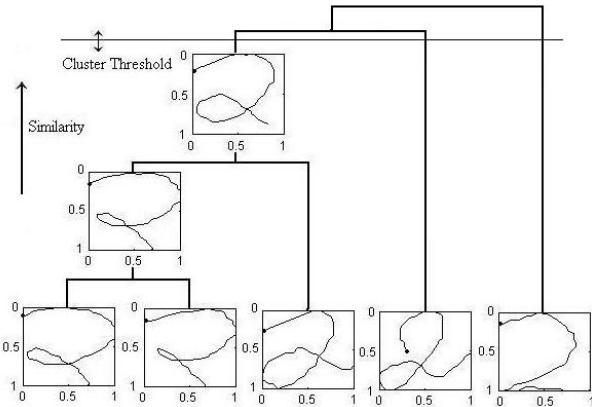
## 2 Recognition System Design

### 2.1 Data Pre-processing and Feature Extraction

Wide ranges of writing styles with variable stroke number and order are collected from Nepalese natives. Handwriting consists of a time sequence of strokes. Pre-processing techniques are varying with the script used from one system to another. Generally, real time data are noisy and incomplete. In Nepali, it is very difficult to identify and eliminate the noisy sequences. However, this paper eliminates cusps or undesirable hooks at ascenders and descenders, based on the nature of slopes. This is checked by the use of angles among 8-10 two-dimensional coordinates if the angle changes drastically (approx. 90 degrees is considered). In addition, size normalization is needed because of the variable size of writing samples (sometimes very big and sometimes very small). Size normalization is done to change the variable sized alphanumeric character to the standard size. The new window size is designated as,  $x_{new} = \left(\frac{x - x_{min}}{x_{max} - x_{min}}\right)(x_{max'} - x_{min'}) + x_{min'}$  and  $y_{new} = \left(\frac{y - y_{min}}{y_{max} - y_{min}}\right)(y_{max'} - y_{min'}) + y_{min'}$  where,  $x_{max}$ ,  $y_{max}$  and  $x_{min}$ ,  $y_{min}$  are the maximum and minimum coordinate points of all strokes within a letter.  $x_{max'} = 1$ ,  $x_{min'} = 0$  and  $y_{max'} = 1$ ,  $y_{min'} = 0$  give the size of new window where all writing samples are restored. A set of strokes after pre-processing is applied to the feature extraction coordinator individually. Consider an alphanumeric character  $C_i$ , consisting of a set of  $m$  strokes from  $i^{th}$  user is,  $S_i = [S_{i,1}, S_{i,2}, \dots, S_{i,m}]$ . Let's take  $j^{th}$  stroke, then,  $S_{i,j} = [p_{i,j,1}, p_{i,j,2}, \dots, p_{i,j,l}]$ ,



**Fig. 1.** A sample of size normalization and cusp elimination is done on a string of 2D coordinates (numeral '8') before extracting its feature and a technique used in extracting feature of the stroke is demonstrated in the left box. The encircled string of coordinates represents a sample of a cusp, formed by the local points: p2, p3 and p4. The drastic change in the angle from p3 gives a birth of a cusp. The right box illustrates the similarity measure between two pairs of numerals using DTW, where each black dot represents a feature event.



**Fig. 2.** A sample of hierarchy in clustering based on similarity is demonstrated using numeral class  $\mathfrak{R}$  with a significant example of a stopping threshold

where  $p_{i,j,k} = (x_{t(i,j,k)}, y_{t(i,j,k)})$ . The complete feature of  $j^{th}$  stroke is,  $F_{i,j} = [(p_{i,j,1}, f(p_{i,j,1}, p_{i,j,2})), (p_{i,j,2}, f(p_{i,j,2}, p_{i,j,3})), \dots, (p_{i,j,l-1}, f(p_{i,j,l-1}, p_{i,j,l}))]$ , where,  $f(p, q) = \arctan(\frac{y_q - y_p}{x_q - x_p})$ . Figure 1 demonstrates graphically the stroke pre-processing and its feature extracting technique by taking a numeral class ‘?’.

## 2.2 Clustering

Two steps: firstly, similarity measure and secondly, single-linkage agglomerative hierarchical clustering, based on similarity complete the clustering. Distance finding technique between the features of the strokes is a common approach to estimate the similarity. Consider two feature vector sequences:  $A$  and  $B$  of length  $N$  and  $M$  respectively.  $D(n, m)$  is the global distance up to  $(n, m)$ ,  $D(n, m) = \min[D(n-1, m-1), D(n-1, m), D(n, m-1)] + d(n, m)$  with an initial condition  $D(1, 1) = d(1, 1)$ , where  $d(n, m)$  is an element of the local distance between the events  $e_n^A$  and  $e_m^B$  i.e.,  $d(n, m) = |e_n^A - e_m^B|$ . The global distance between  $A$  and  $B$  is,  $Dist.(A, B) = D(N, M)$ . This is often called DTW-matching score in recognition process. Fig. 1 shows two pairs of numerals ((?,?) and (?,?)) associated in matching process for similarity determination. Two feature vector sequences of strokes are said to be similar if the matching-score in between them is smaller than other pairs. We merge these two clusters and find a new cluster, which is the average of two merged clusters. Here, a cluster represents a feature vector sequence of a stroke. The representative of the merged clusters is computed by averaging clusters' members pair wise via the use of warping path along the diagonal DTW-matrix [5]. This process is repeated until it reaches stopping threshold (cluster threshold). The value of the threshold gives the number of cluster representatives after clustering. Cluster threshold is equal to the number of complete letters employed because writing pattern of every drawing is different from one to another. Clustering is done for each class of alphanumeric character separately.

### 2.3 Recognition Process

The recognition of stroke starts with a series of feature matching from test stroke with every template using DTW, resulting DTW-matching scores. The template with the lowest DTW-matching score is said to be the most similar feature with the test feature, which is the recognized stroke. A complete letter is recognized once all the components (strokes) are identified, i.e. the sum of the lowest DTW-matching scores from every matching is minimum from the frame of same test letter. Consider a 2-stroke consonant ‘क’ as a test data. Both test stroke matched with every stored template separately. The recognizer determines the lowest DTW-matching score from every frame from both matching. To classify, the sum of two lowest DTW-matching scores should be the smallest score from the frame ‘क’ as compared to other frames, otherwise misrecognized.

## 3 Experiments and Discussions

No directions, constraints and limitations were given to the users but they were encouraged to write alphanumeric characters with variable stroke number and order along the horizontal line. 25 users were employed to write two times for each letter, (i.e. 25 users×2 times×46 classes). Training data was composed of 60% of data from every class and remaining 40% were for testing. The dataset was comprised of natural handwritings. We trained separate system using original and pre-processed writing samples. Table 1 reveals the experimental results. Recognition improvement is a little bit higher in pre-processed writing samples. Nevertheless difficulties arise because improvement of one class of letter often leads to the deterioration of another during pre-processing. For example, cusp elimination at the ascender and descender sometimes changed the shape of the writing: भ→म, ध→घ, थ→य, न→त, ठ→ट etc., but it corrected the reverse confusions significantly. Similarity in drawings, such as, क↔फ, च↔थ, छ↔ध, य↔प, स↔झ, ठ↔द, इ↔ड, ठ↔ट, ०↔८, ८→८, ८→ट, ७→८, ६→छ, २→र etc. holds high error rate. Some of the confusion pairs are not readable for humans too. Diminished and / or very long ascenders or descenders in strokes are confusion due to the change in shape. Writing two times the same stroke (re-writing) at the end of writing to complete previous stroke are misrecognized. In addition, tremor handwritings also affect the recognition rate. Table 2 explores error analysis of this task. It is noted that letters under miswriting error type do not give any information of complete letter.

**Table 1.** Error rates for both original and preprocessed strokes

Dataset	Char.'s Type (Class)	Test Char.s	Original Strokes		Pre-processed Strokes	
			Char.s (Err)	Avg. Err	Char.s (Err)	Avg. Err
Training	Consonant (31)	930	26 (02.79%)		22 (02.36%)	
	Vowel (5)	150	08 (05.34%)	02.75%	07 (04.67%)	02.31%
	Numerical (10)	300	04 (01.34%)		03 (01.00%)	
Test	Consonant (31)	620	95 (15.32%)		81 (12.58%)	
	Vowel(5)	100	17 (17.00%)	14.13%	13 (13.00%)	11.41%
	Numerical (10)	200	18 (03.00%)		11 (02.00%)	

**Table 2.** Error Type (Test Data)

Error Type	Original Strokes (Err)	Pre-processed Strokes (Err)
Similar Features	77 (08.15%)	56 (07.28%)
Diminished or very long Ascenders and Descenders	16 (01.73%)	13 (01.41%)
Miswriting	18 (01.95%)	18 (01.95%)
Rewriting Strokes	13 (01.41%)	13 (01.41%)
Miscellaneous	06 (00.65%)	05 (00.64%)

## 4 Conclusion and Future Work

We have presented recognition of *stroke number and order free* Nepalese handwritings by using *structural approach*. We have evaluated our automatic systems with overall accuracies of 85.87% and 88.59% using both original and pre-processed writing samples respectively. We plan to extend up to syllable level recognition with the use of positional relation among the strokes and further improvements.

## Acknowledgment

The author would like to thank to all co-operative people, participated in writing alphanumeric characters for both training and test data.

## References

1. Bansal V., Sinha R.M.K.: On how to Describe Shapes of Devanagari Characters and use them for Recognition. In Proceedings of 5th International Conference Document Analysis and Recognition (1999) 410–413
2. Breuel T. M.: Segmentation of Handprinted Letter Strings using a Dynamic Programming Algorithm. In proceedings of 6th International Conference on Document Analysis and Recognition (2001) 821–826
3. Connell S. D., Sinha R. M. K., Jain A. K.: Recognition of Unconstrained On-Line Devanagari Characters. In Proceedings of 15th International Conference on Pattern Recognition (2000) 368–371
4. Sinha R.M.K., Mahabala H.N.: Machine Recognition of Devanagari Script. IEEE Transactions System, Man and Cybernetics, Vol. 9 No. 8, (1979) 435–441
5. Somervuo P., Kohonen T.: Self Organizing Maps and Learning Vector Quantization for Feature Sequences. Neural Processing Letters, Vol. 10 No. 2, (1999) 151–159

# Diagnosis Model of Radio Frequency Impedance Matching in Plasma Equipment by Using Neural Network and Wavelets

Byungwhan Kim<sup>1</sup>, Jae Young Park<sup>2</sup>, Dong Hwan Kim<sup>3</sup>, and Seung Soo Han<sup>4</sup>

<sup>1</sup> Department of Electronic Engineering, Sejong University,  
Seoul, Korea

kbwhan@sejong.ac.kr

<sup>2</sup> Department of Electronic Engineering, Kwangwoon University,  
Seoul, Korea  
jaepark@kw.ac.kr

<sup>3</sup> School of Mechanical Design and Automation Engineering,  
Seoul National University of Technology, Seoul, Korea  
dhkim@sntu.ac.kr

<sup>4</sup> Department of Information Engineering, Myongji University,  
Yongin, Korea  
shan@mju.ac.kr

**Abstract.** A new calibration model for plasma diagnosis was constructed by combining radio frequency impedance match data, wavelet, and neural network. A total of 30 fault symptoms were simulated with the variations in the four process parameters. Both discrete wavelet transformation (DWT) and continuous wavelet transformation (CWT) were utilized to filter the sensor information. Three types of diagnosis models (raw-, DWT-, and CWT-based models) were constructed. The comparisons revealed that the improvement in the prediction performance of DWT and CWT data models over the raw data model were about 42% and 30%, respectively.

## 1 Introduction

To improve device yield or equipment throughput, abnormal states in plasma should be timely detected and accurately diagnosed. An ex-situ film measurement-based neural network model was reported to identify a plasma fault [1]. In-situ optical emission spectroscopy data were also utilized to construct neural network models for plasma diagnosis and control [2-5]. Apart from the diagnosis, intelligent techniques were successfully applied to predict various types of plasma process data [6-9]. Another useful source for plasma control is the radio frequency (RF) match network. As an important component in plasma etch equipment, the match network plays a role of balancing impedances between the equipment plasma and RF generator. The match process is typically conducted by controlling two mechanical motors. Their steady positions can vary considerably with the variations in process parameters or in circuit components comprising the match network. The position variations were examined to

monitor plasma states [10]. Meanwhile, the RF match data can be utilized to construct a calibration model for plasma diagnosis.

In this study, a calibration model was constructed by combining RF match network data, wavelet, and neural network. The monitor system [10] was used to collect real-time match data for the simulated faults. The wavelets were used to examine the effect of filtering match sensor information on the prediction capability of diagnosis model. Wavelet-based models are also compared to non-wavelet-based models.

**Table 1.** Experimental process parameters and ranges involved in generating fault data

Process Factors	Ranges	Increments	Units
Source Power	500-1800	100 (200)	W
Pressure	7.5-20	2.5	mTorr
O <sub>2</sub> Flow Rate	50-100	10	sccm
Ar Flow Rate	50-190	20	sccm

## 2 RF Impedance Matching Data

The real-time monitoring system for collecting RF impedance matching variables was detailed in Ref. [10]. To construct a calibration model, plasma faults were simulated with the variations in the process parameters as shown in Table 1. For the variations in the source powers ranging between 500 W and 1000 W, as shown in Table 1, an increment of 100 W was employed. For the source powers larger than 1000 W, the increment changed to 200 W. 10 cases were therefore generated for variations in source power. In this way, 6, 8, and 6 cases were simulated for the variations in other pressure, Ar flow rate, and O<sub>2</sub> flow rate, respectively. As a result, a total of 30 fault patterns could be prepared. 20 patterns of them were used to train the BPNN. The trained BPNN was then tested with the remaining 10 patterns. For each case, impedance matching variables were collected, and they were two electrical match positions and one reflected power.

## 3 Results

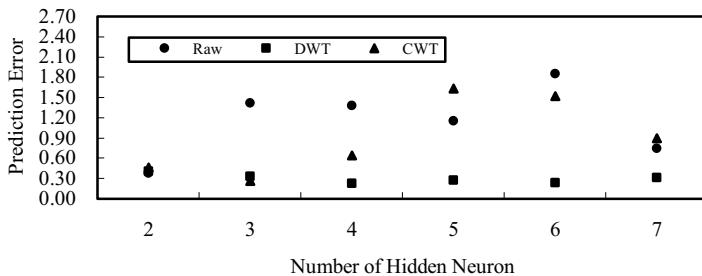
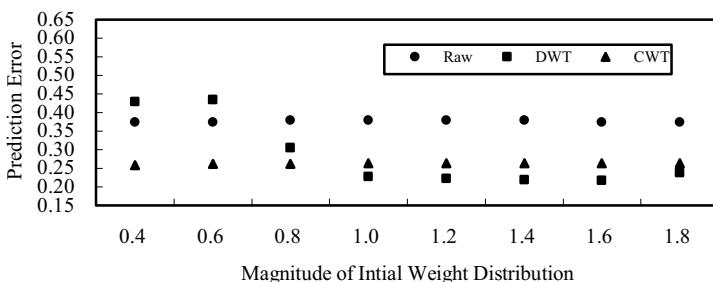
### 3.1 Wavelet Data Conversion

For wavelet transformation, 32 data elements were sampled from the entire range of collected data. Two wavelets, discrete wavelet transformation (DWT) and continuous wavelet transformation (CWT) [11], were used to filter the match data. Each wavelet has two control parameters that determine the efficiency of data transformation. For the DWT, they are the type of daubechies function (TDF) and the scale level (SL). Meanwhile, the CWT efficiency is governed by the dilation parameter (DP) and translation parameter (TP). To optimize the set of control parameters, two metrics were introduced and they are percent sensitivity (PS) and total percent sensitivity (TPS) [12]. As conducted earlier, the effects of transformation factors on the TPS were optimized. The resulting optimized factors are shown in Table 2.

**Table 2.** Wavelet control factors determined for the optimization of TPS

Process Parameters	DWT		CWT	
	SL	TDF	DP	TP
Source Power (W)	5	8	2	1
Pressure (mTorr)	4	7	2	1
O <sub>2</sub> flow rate (sccm)	4	7	2	1
Ar flow rate (sccm)	4	7	2	1

With setting the control parameters to those shown in Table 2, the original 32 sampled data elements were transformed using the wavelets. Then, the statistical mean and standard deviation of raw and wavelet filtered data were computed for the phase and impedance positions. For the remaining reflected power, the steady raw data were used. The network input vector consisted of these five quantities. The network output vector was composed of four elements, each corresponding to one of the process parameters.

**Fig. 1.** RMSE variation of diagnosis models as a function of NHN**Fig. 2.** RMSE variation of diagnosis model as a function of MIWD

### 3.2 Comparison of Diagnosis Models

With the three types of match data, BPNN diagnosis models were constructed. As the predictor, the backpropagation neural network (BPNN) [13] was adopted. Two training factors, the number of hidden neurons (NHN) and magnitude of initial weight

distribution (MIWD) were varied in the range 2-7 and  $\pm 0.4\text{-}\pm 1.8$ , respectively. As the termination criterion, the training tolerance was set to 0.1. The training factors stated earlier were detailed in our previous work [14]. The prediction errors quantified in the root mean squared error (RMSE) were calculated with the 10 test experiments prepared earlier. The RMSEs are shown in Fig. 1 as a function of NHN. As shown in Fig. 1, the RMSE variations for DWT data vary little with the NHN. In contrast, the other raw and CWT data are seen to considerably vary. The smallest RMSE for the raw data model is obtained at 2 hidden neurons and is about 0.380. For the DWT and CWT data models, the smallest RMSEs are achieved at 4 and 3, respectively. They are 0.228 and 0.264 for the DWT and CWT data models, respectively. Compared to the raw data model, both DWT and CWT data models demonstrate an improvement of about 40% and 30%, respectively. It should be noted that the DWT data model yielded more improved prediction over the CWT data model. After setting each diagnosis model to their optimized NHN, the RMSE was more investigated as a function of the MIWD. The results are shown in Fig. 2. The RMSE variations of the raw and CWT data models vary little with the MIWD. Actually, the RMSEs for the raw and CWT data model vary a few around 0.375 and 0.217, which are almost comparable to those determined in Fig. 1. In contrast, the RMSE for the DWT varies appreciably with the MIWD. For the DWT data model, the smallest RMSE was obtained at  $\pm 1.6$ , and is equal to about 0.217. Compared to the corresponding RMSE identified in Fig. 1, the improvement for the DWT data is about 5%. The improvement of this model over the raw data model is about 42%. The comparisons reveal that the MIWD variation was effective only to the DWT-based model. From all comparisons, it is identified that wavelet-filtered data were effective to improve the prediction capability of plasma diagnosis models. This feature is greatly advantageous in a model-based real-time diagnosis of plasma equipment.

## 4 Conclusion

In this study, a new diagnosis model was constructed by predicting wavelet-filtered impedance match data using neural network. The fault symptoms were characterized by two important match variables (impedance and phase position) and one reflected power. Three types of diagnosis models were compared from the standpoint of the prediction performance as a function of the training factors. Wavelet data models demonstrated a considerable improvement over the raw data model. This indicates that DWT is an effective means to improve the prediction performance of neural network diagnosis model. This feature can generally be applied in constructing any neural network diagnosis models of equipment sensor, or in-situ plasma sensing data.

## Acknowledgements

This work was supported by the Seoul Research and Business Development Program (Grant No.10583), and by the MIC (Ministry of Information and Communication), Korea, under the ITRC (Information Technology Research Center) support program supervised by the IITA (Institute of Information Technology Assessment).

## References

1. Kim, B., May, G.S.: Real-Time Diagnosis of Semiconductor Manufacturing Equipment Using a Hybrid Neural Network Expert System. *IEEE Trans Comp Packag Manufact Technol* 20(1) (1997) 39-47
2. Hong, S.J., May, G.S.: Neural Network-Based Real-Time Malfunction Diagnosis of Reactive Ion Etching Using In Situ Metrology Data. *IEEE Trans Semicond Manufact* 17(3) (2004) 408-421
3. Kim, B., Bae, J. Hong, W.S.: Plasma Control Using Neural Network and Optical Emission Spectroscopy. *J Vac Sci Technol A* 23(2) (2005) 355-358
4. Kim, B., Kim, S.: Plasma Diagnosis by Recognizing In-Situ Data Using a Modular Back-propagation Network. *Chemometr Intell Lab Syst* 65(2) (2003) 231-240
5. Kim, B, Kim, S.: Partial Diagnostic Data to Plasma Etch Modeling Using Neural Network, *Microelectron Eng* 75(4) (2004)
6. Kim, B., Park, J.H.: Qualitative Fuzzy Logic Model of Plasma Etching Process. *IEEE Trans Plasma Sci.* 30(2) (2002) 673-678
7. Kim, B., Han, S.S., Kim, T.S., Kim, B.S., Shim, I.J.: Modeling refraction characteristics of silicon nitride film deposited in a SiH<sub>4</sub>/NH<sub>3</sub>/N<sub>2</sub> plasma using neural network. *IEEE Trans Plasma Sci* 31(3) (2003) 317-323
8. Kim, B, Kwon K.H., Kwon, S.K., Park, J.M., Yoo, S.W., Park K.S., You I.K., Kim, B.W.: Modeling etch rate and uniformity of oxide via etching in a CHF<sub>3</sub>/CF<sub>4</sub> plasma using neural networks. *Thin Solid Films* 426(1-2) (2003) 8-15
9. Kim, B., Kim, S., Kim, K.: Modeling of Plasma Etching Using a Generalized Regression Neural Network. *Vacuum* 71(4) (2003) 497-503
10. Kim, B., Lee, C.: Monitoring Plasma Impedance Match Characteristics in a Multipole Inductively Coupled Plasma for Process Control. *J Vac Sci Technol A* 18(1) (2000) 58-62
11. S. G. Mallat: A theory for multiresolution signal decomposition: The wavelet representation *IEEE Trans. Pattern Anal. Mach. Intell.* Vol. 11, No. 7 (1989) 674-693.
12. Kim, B., Choi W.: Using wavelet filtering for monitoring plasma conditions. *Solid State Technol.* 44(11) (2001) 73-81
13. Rummelhart, D.E., McClelland, J.L.: *Parallel Distributed Processing*. MIT Press, Cambridge (1986)
14. Kim, B., Park, S.: An Optimal Neural Network Plasma Model: A Case Study. *Chemometr Intell Lab Syst* 56(1) (2001) 39-50

# Program Plagiarism Detection Using Parse Tree Kernels

Jeong-Woo Son, Seong-Bae Park\*, and Se-Young Park

Department of Computer Engineering

Kyungpook National University

702-701 Daegu, Korea

{jwson, sbpark, separk}@sejong.knu.ac.kr

**Abstract.** Many existing plagiarism detection systems fail in detecting plagiarism when there are an abundant garbage in the copied programs. This is because they do not use the structural information efficiently. In this paper, we propose a novel plagiarism detection system which uses parse tree kernels. By incorporating parse tree kernels into the system, it efficiently handles the structural information within source programs. A comparison with existing systems such as SID and JPlag shows that the proposed system can detect plagiarism more accurately due to its ability of handling structural information.

## 1 Introduction

Parker and Hamblett [1] defined a software plagiarism as *a program which has been produced from another program with a small number of routine changes*. By using another students' program, one can produce a program without understanding what the source code does. Consequently, the detection of plagiarism has been an essential task for a professor.

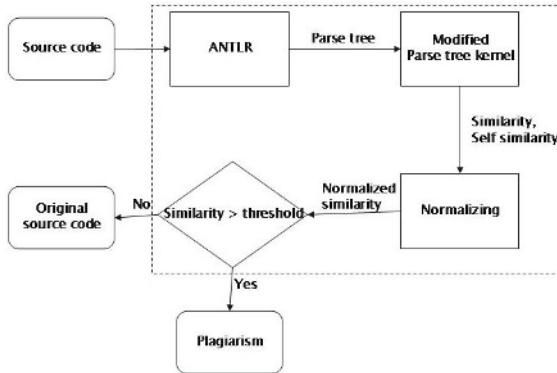
The program plagiarism detection systems are divided into two categories. The first one is an attribute-counting-metric system. This system represents a program as a vector of which elements are the number of operators, operands, unique operators, and unique operands. Halstead first proposed a plagiarism detection system using a software science metric [2]. However, these systems shows relatively poor performance compared with those in the second type.

The second type of plagiarism detection systems not only used attribute-counting metrics, but also compared structure of source codes. Prechelt et al. proposed the JPlag system [3] based on the greedy string tiling algorithm. Chen et al. suggested the SID system [4]. Even if they show good performance and use structural information partially, the structural information is not fully reflected within their metric. It results in poor performance when plagiarists insert large number of useless codes into the source code.

To efficiently express the structural information in the program source code, the system should use the parse tree which has structural information of a source code. The effective way for this purpose is using a kernel method. Kernel methods

---

\* Corresponding author.



**Fig. 1.** The structure of the proposed system

can handle complex data in high dimensional space easily and represent the inner product of the vector representations as a similarity. Collins and Duffy proposed a parse tree kernel [5] derived from the convolution kernel.

In this paper, we modify a parse tree kernel to be applicable to the plagiarism detection. With the modified parse tree kernel, we propose a novel plagiarism detection method for Java language. We evaluated the proposed method with 4 variations of a Java source code. The experimental results prove that the proposed method can successfully detect plagiarism. Especially, the high performance in detecting the variation with a structural attack implies that the method can handle structural information efficiently.

## 2 Plagiarism Detection System

The proposed plagiarism detection system detects plagiarism using the parse tree of a source code. Figure 1 illustrates the proposed system which operates in three steps.

### 2.1 Extracting Parse Trees

To use structural information of source code, parse trees are extracted with ANTLR [6]. ANTLR is a language tool that provides a framework for constructing compiler, translator from grammatical descriptions. It also has tree parser to translate source code into a parse tree.

### 2.2 Parse Tree Kernel

Let  $subtree_1, subtree_2, \dots$  be all of subtrees in a parse tree. Then, the parse tree  $T$  can be represented as a vector

$$V_T = (\#subtree_1(T), \#subtree_2(T), \dots, \#subtree_n(T))$$

where  $\#subtree_i(T)$  is a frequency of  $subtree_i$  in parse tree  $T$ . The inner product of two trees,  $T_1$  and  $T_2$  can be defined:

$$\begin{aligned} < V_{T_1}, V_{T_2} > &= \sum_i \#subtree_i(T_1) \cdot \#subtree_i(T_2) \\ &= \sum_i \left( \sum_{n_1 \in N_{T_1}} I_{subtree_i}(n_1) \right) \cdot \left( \sum_{n_2 \in N_{T_2}} I_{subtree_i}(n_2) \right) \\ &= \sum_{n_1 \in N_{T_1}} \sum_{n_2 \in N_{T_2}} C(n_1, n_2) \end{aligned}$$

where  $N_{T_1}$  and  $N_{T_2}$  are all of nodes in tree  $T_1$  and  $T_2$ . The indicator function  $I_{subtree_i}(n_1)$  is to be 1 if  $subtree_i$  is seen rooted at node  $n$ .  $C(n_1, n_2)$  is a function which is defined as

$$C(n_1, n_2) = \sum_i I_{f_i}(n_1) \cdot I_{f_i}(n_2)$$

This function can be calculated with the recursive definition [5].

### 2.3 Modified Parse Tree Kernel

Even though the parse tree kernel can handle tree data efficiently, it is not adequate for plagiarism detection. The first problem is that the similarity value increases too fast to handle. The scale of similarity should be reduced by using a logarithm.

Second, the value of the kernel between two different trees is typically much smaller than the value between same trees. To tackle this problem, the depth of subtrees is limited to a threshold value determined empirically. By restricting the depth, the modified kernel can calculate both the similarity between codes and the self similarity without influence of small changes.

### 2.4 Detecting Plagiarism

In the proposed system, a source code is decided as plagiarism when the similarity is larger than a threshold value. However, the similarity obtained via the kernel does not have any upper bound, so it needs to be normalized. Let  $S_1, S_2$  be source codes to be compared and a function  $K(S_1, S_2)$  returns the similarity using the modified parse tree kernel. Then, the normalizing function  $N(S_1, S_2)$  is defined as

$$N(S_1, S_2) = \frac{K(S_1, S_2)}{E(S_1, S_2)} \times 100,$$

where a function  $E(S_1, S_2)$  is normalizing factor defined as

$$E(S_1, S_2) = \frac{K(S_1, S_1) + K(S_2, S_2)}{2}.$$

**Table 1.** Experimental result of the proposed method on simple attacks

Source Code	Group1	Group2	Group3	Group4	Source Code	Group1	Group2	Group3	Group4
1	27	<b>100</b>	32	34	9	25	32	<b>100</b>	31
2	29	<b>55</b>	33	35	10	26	33	<b>47</b>	32
3	27	<b>45</b>	29	32	11	23	29	<b>70</b>	30
4	28	<b>45</b>	31	33	12	25	31	<b>58</b>	30
5	<b>100</b>	27	25	30	13	30	34	31	<b>100</b>
6	<b>47</b>	27	25	30	14	31	34	29	<b>49</b>
7	<b>60</b>	25	23	25	15	25	31	28	<b>62</b>
8	<b>51</b>	25	23	25	16	29	33	30	<b>64</b>

### 3 Experiments

#### 3.1 Data Set

The data set for experiments are prepared by 4 programmers. Each programmer writes an original source code, and then modifies his code with following two types of attacks.

1. Simple attack type
  - (a) Modify variable, function or class names.
  - (b) Change expressions such as ‘for’, ‘do~while’, and ‘if’ to the synonymous expressions.
  - (c) Merge or Separate functions and classes.
2. Structural attack type
  - (a) Add useless 80 lines into a source code.
  - (b) Add useless 120 lines into a source code.
  - (c) Add useless 660 lines into a source code.

#### 3.2 Experimental Result

In all the experiments, the depth of subtrees is limited to 5 and set the similarity threshold to 40. Table 1 represents the experimental results of the proposed system with simple attack type data. In this table, the bold item indicates the source codes that are recognized as plagiarism. Not only the proposed method but also all other plagiarism detection systems such as SID and JPlag also succeeded in finding the plagiarized programs. All the compared systems showed 100.00% of accuracy. This result implies that most plagiarism detection systems can find a plagiarism which is simply modified from an original source code.

Table 2 shows that the proposed method uses the structural information more efficiently than other systems. As more useless lines are added, the similarity gets down slowly but is still within the threshold. JPlag absolutely fails in detecting plagiarism when many useless lines are inserted. Even though SID is not affected by the added useless lines, the similarity values are too low, which makes it difficult to make a decision of plagiarism. In conclusion, the more useless codes are added, the better performance is shown by the proposed system.

**Table 2.** Performance on structural attacks

Number of added lines	Proposed Method	SID	JPlag
80	65.9	38	73.30
160	58.1	16	67.80
660	42.5	34	15.40

## 4 Conclusions

In this paper, we proposed a new system to detect program plagiarism by using the parse tree kernel to efficiently reflect structural information in a source code. The similarity between parse trees extracted by a ANTLR parser is measured by a parse tree kernel which is modified for the program plagiarism.

In the experiments of two attack types, the proposed method outperformed other systems. The method achieved 100% of accuracy for the simple attack, and it is not much affected by the structural attack while others are fatal to the structural attack.

An advantage of the proposed method is that it is independent of programming languages. The only requirement for other languages such as C, C++, and Pascal is the parsers for the languages.

## Acknowledgements

This work was supported by grant No. (R01-2006-000-11196-0) from the Basic Research Program of the Korea Science & Engineering Foundation.

## References

1. A. Parker and J. Hamblen, “Computer Algorithms for Plagiarism Detection,” *IEEE Transactions on Education*, Vol. 32, No. 2, pp. 94–99, 1989.
2. M. Halstead, *Elements of Software Science*, Elsevier, 1977.
3. L. Prechelt, G. Malphol, and M. Philippson, “Finding Plagiarisms among a Set of Programs with JPlag,” *Journal of Universal Computer Science*, Vol. 8, No. 11, pp. 1016–1038, 2002.
4. X. Chen, B. Francia, M. Li, B. McKinnon, and A. Seker, “Shared Information and Program Plagiarism Detection,” *IEEE Transactions on Information Theory*, Vol. 50, No. 7, pp. 1545–1551, 2004.
5. M. Collins and N. Duffy, “Convolution Kernels for Natural Language,” In *Proceedings of the 14th Neural Information Processing Systems*, 2001.
6. T. Parr and R. Quong, “ANTLR: A Predicated-LL( $k$ ) Parser Generator,” *Journal of Software Practice & Experience*, Vol. 25, No. 7, 1995.

# Determine the Optimal Parameter for Information Bottleneck Method

Gang Li<sup>1</sup>, Dong Liu<sup>2</sup>, Yangdong Ye<sup>2</sup>, and Jia Rong<sup>1</sup>

<sup>1</sup> School of Engineering and Information Technology, Deakin University,  
221 Burwood Highway, Vic 3125, Australia

<sup>2</sup> School of Information Engineering, Zhengzhou University, Zhengzhou, China  
`gang.li@deakin.edu.au`, `ielsld1@gs.zzu.edu.cn`, `yeyd@zju.edu.cn`,  
`jron@deakin.edu.au`

**Abstract.** A natural question in *Information Bottleneck* method is how many “groups” are appropriate. The dependency on prior knowledge restricts the applications of many *Information Bottleneck* algorithms. In this paper we aim to remove this dependency by formulating the parameter choosing as a model selection problem, and solve it using the minimum message length principle. Empirical results in the documentation clustering scenario indicates that the proposed method works well for the determination of the optimal parameter value for information bottleneck method.

## 1 Introduction

Given a joint distribution  $p(x, y)$ , the *Information Bottleneck* method [1] constructs a new representation  $T$  that defines partitions over the values of  $X$  that are informative about  $Y$ . The compactness of the representation is then determined by the mutual information between  $T$  and  $X$ ,  $I(T; X)$ , while the accuracy of the representation is measured by  $I(T; Y)$ . The *Information Bottleneck* method is proposed to find a stochastic mapping, parameterized by probability  $p(t|x)$  that maximizes:

$$I(T; Y) - \beta I(T; X) \quad (1)$$

for some prespecified  $\beta > 0$ , which balances the tradeoff between the compression and preservation. For applications in which the motivation is to group the data, it is tempting to set  $\beta$  to be  $\infty$ . In this case, the equation (1) becomes the maximization of  $I(T; Y)$ , and the optimal IB solution tends to become crisp, i.e., the  $p(t|x)$  approaches zero or one almost everywhere. If the cardinality value of  $T$  is given, there exist several information bottleneck algorithms which aim to produce this crisp representation  $T$ , as shown in [2] and [3].

It would be attractive if an IB algorithm has a build-in mechanism which could automatically determine this parameter. This is in general a question of model selection. In this paper, we introduce the “*Minimum Message Length*” (MML) principle into IB method, and show that one can give this question a formal optimal solution that is also based on information theoretical considerations.

## 2 Choosing the Optimal Cardinality Value Using the MML principle

The basic idea of the *Minimum Message Length* (MML) [4]principle is to find the hypothesis  $H$  which leads to the shortest message. In the case of determining the optimal cardinality value for IB method, it requires the estimate of the optimal encoding code for the description of the IB solution and the original data  $D$ . The whole message consists of 2 parts:

1. The message costed to describe the IB solution. This part of message can be further divided into two sub-messages.
  - (a) message encoding the cardinality value  $k$ .
  - (b) message encoding the IB solution  $T_k$ .
2. The message costed to describe the original data set  $D$ , under the assumption that the IB solution was the optimal one.

Given a cardinality value  $k$ , the IB solution returned from any IB algorithm is  $k$  groups of objects. The encoding of this model shall consist of the following two terms:

1. Encoding of the cardility value  $k$ : The encoding length of this description can be estimated as  $\log^*(k)$ , where  $\log^*$  is the universal code length for integers [5].
2. Encoding of the IB solution  $T_k$ : To encode the IB solution, we can code the index in some agreed enumeration of all possible groupings. As the number of possible groupings is exactly the definition of Stirling numbers of the second kind [6], the number of bits needed to code the  $T_k$  can be estimated as  $\log S_2(n, k)$ .

Thus, the encoding length of  $k$  and the  $T_k$  can be estimated as:

$$MsgLen(k) + MsgLen(T_k) = \log^*(k) + \log S_2(n, k) \quad (2)$$

where  $k$  is the cardinality value, and  $n$  is the number of objects.

In the case of encoding the original data set  $X$  when the IB solution  $T_k$  is known, the encoding length can be estimated as  $n$  multiplies the  $H(X|T_k)$ , i.e.

$$\begin{aligned} MsgLen(D|k, T_k) &= n \times H(X|T_k) \\ &= n \times (H(X) - I(X; T_k)) \end{aligned} \quad (3)$$

where  $I(X; T_k)$  is the mutual information between  $X$  and  $T_k$ , and  $H(X)$  is the entropy of  $X$ .

Therefore, the total message length for the IB model and the original data set  $D$  can be estimated as

$$MsgLen = \log^*(k) + \log S_2(n, k) + n \times (H(X) - I(X; T_k)) \quad (4)$$

where  $k$  is the cardinality value,  $n$  is the number of objects in  $D$ .

## 2.1 Automatical SIB Algorithm — ASIB

Based on the encoding length formula (4), we can design an iterative algorithm as described in Algorithm 1: Starting with a range  $[min_k, max_k]$  of possible cardinality values (by default,  $min_k = 1$  and  $max_k = n$ ), the algorithm runs through each possible  $k$  value, and a sequential IB algorithm is called to get the IB solution  $T_k$ , then the encoding length of the IB model is calculated. Finally the  $k$  which results in the minimum encoding message length will be returned as the optimal cardinality value.

---

**Algorithm 1.** Automatical SIB algorithm

---

**Require:** a joint distribution  $p(x, y)$ , the range of possible cardinality values  $[min_k, max_k]$  (optional).

**Ensure:** the optimal cardinality value  $bestK$ , and the feature grouping result  $T_{bestK}$

$T \leftarrow$  an empty array with length  $max_k - min_k + 1$   
 $\{T\}$  will store the IB solutions corresponding to different  $k$

$MsgLen \leftarrow$  an empty array with length  $max_k - min_k + 1$   
 $\{MsgLen\}$  will store the encoding length corresponding to different  $k$

**for** each  $k \in [min_k, max_k]$  **do**

$T_k \leftarrow sIB(k)$

$MsgLen(k) \leftarrow \log^*(k) + \log S_2(n, k) + n \times (H(X) - I(X; T_k))$

**if**  $msgLen(k) < msgLen(bestK)$  **then**

$bestK = k$

**end if**

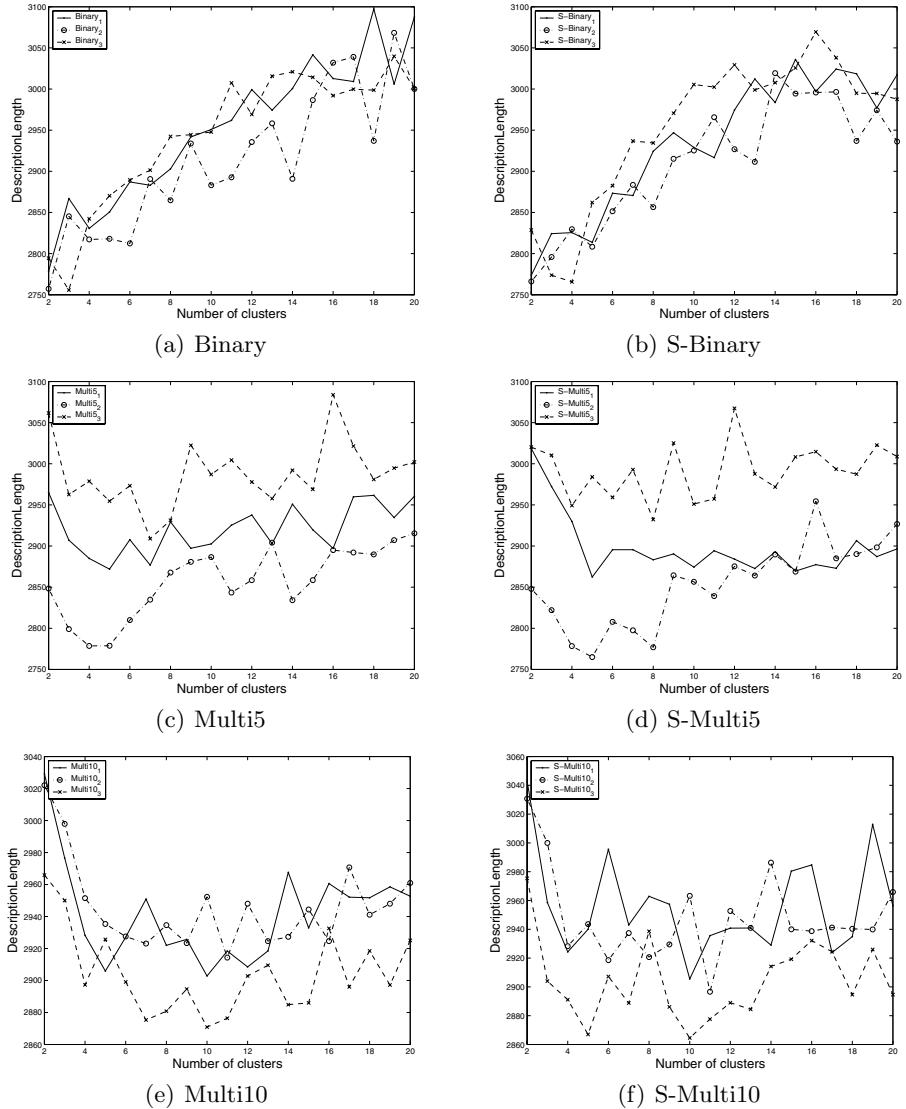
**end for** {Iterate each candidate feature set to choose the optimal one}  
 returns  $bestK$  and  $T_{bestK}$

---

## 3 Experiment Design and Results Analysis

In this section, we evaluate the performance of the proposed *ASIB* algorithm under the document clustering scenario, as in the paper [2]. The datasets used in our test are 9 original and 9 shuffled document-words counts datasets [2]. In the *ASIB* algorithm, we set  $min_k = 2$  and  $max_k=20$ . The *ASIB* algorithm is then called to automatically determine the correct number of document categories, and cluster the documents.

Left column of the Figure 1 shows the general shape of the total cost versus the number of possible categories. In Figure 1(a), it is interesting to note that the encoding length increases as the increasing of cluster number, and data sets *Binary-1* and *Binary-2* arrive at their minimum encoding length when the  $k = 2$ . The data set *Binary-3* arrived at its minimum encoding length when  $k = 3$ , which is very close to its correct number of categories 2. Figure 1(c) and 1(e) present a similar results, in which we can find that the *ASIB* algorithm determined the correct number of clusters for *Multi5-1*, *Multi10-1* and *Multi10-3*, and located very close results for data sets *Multi5-2*, *Multi5-3* and *Multi10-2*. From the figure, it is evident that the *ASIB* algorithm successfully determined



**Fig. 1.** Cost versus number of Categories

the correct number of categories in 5 out of 9 data sets, and located very close values for the other 4 data sets with an error just around 1.

The *ASIB* algorithm is then called to automatically determine the correct number of document categories on 9 shuffled data sets. Right column of the Figure 1 shows the general shape of the total cost versus the number of possible categories. In Figure 1(b), it is interesting to note that the encoding length also increases as the increasing of  $k$ , and data sets *S-Binary-1* and *S-Binary-2* arrive at their minimum encoding length when  $k = 2$ . The data set *S-Binary-3* arrived

at its minimum encoding length when  $k = 3$ , which is very close to its correct categories number 2. Figure 1(d) and 1(f) present a similar results, in which we can find that the *ASIB* algorithm determined the correct number of clusters for *S-Multi5-1*, *S-Multi10-1* and *S-Multi10-3*, and located a very close results for data sets *S-Multi5-2*, *S-Multi5-3* and *S-Multi10-2*.

These results are of special interest taking into account that no prior knowledge of the number of categories is provided to the *ASIB* algorithm. More over, there is strong empirical evidence suggesting that the proposed method could identify an appropriate parameter value for IB method automatically.

## 4 Conclusion

In this paper, we designed an encoding scheme for IB solutions corresponding to different cardinality values, and developed a MML-based algorithm — *ASIB* to automatically determine the best cardinality value based on the data set itself. The experimental results on document clustering showed that the proposed *ASIB* algorithm is capable of recovering the true category number, and this indicates that the proposed parameter determining mechanism is promising.

## References

1. N. Tishby, F.P., Bialek, W.: The information bottleneck method. In: Proc. 37th Allerton Conference on Communication and Computation. (1999)
2. Slonim, N.: The Information Bottleneck: Theory and Applications. PhD thesis, the Senate of the Hebrew University (2002)
3. Slonim, N., Tishby., N.: Agglomerative information bottleneck. Advances in Neural Information Processing Systems (NIPS) 12 (1999) 617–623
4. Wallace, C., Freeman, P.R.: Estimation and inference by compact coding. Journal of the Royal Statistical Society **49** (1987) 223–265
5. Rissanen, J.: Universal Prior for Integers and Estimation by Minimum Description Length. Annals of Statistics **11** (1983) 416–431
6. Eric W. Weisstein: Stirling Number of the Second Kind. (From MathWorld – A Wolfram Web Resource) <http://mathworld.wolfram.com/StirlingNumberoftheSecondKind.html>.

# Optimized Parameters for Missing Data Imputation\*

Shichao Zhang<sup>1,2</sup>, Yongsong Qin<sup>1</sup>, Xiaofeng Zhu<sup>1</sup>, Jilian Zhang<sup>1</sup>, and Chengqi Zhang<sup>2</sup>

<sup>1</sup> Department of Computer Science, Guangxi Normal University, China

<sup>2</sup> Faculty of Information Technology, University of Technology Sydney

P. O. Box 123, Broadway NSW 2007, Australia

{zhangsc, chengqi}@it.uts.edu.au, ysqin@mailbox.gxnu.edu.cn,  
zhu0011@21cn.com, zhangjilian@yeah.net

**Abstract.** To complete missing values, a solution is to use attribute correlations within data. However, it is difficult to identify such relations within data containing missing values. Accordingly, we develop a kernel-based missing data imputation method in this paper. This approach aims at making optimal statistical parameters: mean, distribution function after missing-data are imputed. We refer this approach to *parameter optimization method* (POP algorithm, a random regression imputation). We experimentally evaluate our approach, and demonstrate that our POP algorithm is much better than deterministic regression imputation in efficiency of generating an inference on the above two parameters. The results also show our algorithm is computationally efficient, robust and stable for the missing data imputation.

## 1 Introduction

Missing value imputation is an actual yet challenging issue confronted by machine learning and data mining fields [1,3,4,5,8,9,12,13,14,16,17,18]. There are many approaches to deal with missing values [1,6,17,18], mainly including (a) Ignore objects containing missing values; (b) Fill the missing value manually; (c) Substitute the missing values by a global constant or the mean of the objects; and (d) Get the most probable value to fill in the missing values. The first approach usually will lose too much useful information, whereas the second one is time-consuming and expensive in cost, so it is infeasible in many applications. The third approach assumes that all missing values are replaced with the same value, probably leading to considerable distortion for data distribution. The fourth approach, missing value filling, is the most effective way to deal with the missing value problem [6]. This means that missing data imputation is a reasonable strategy. There exist many methods for missing data imputation (we'll adopt the term 'imputation' in this paper from statistics domain).

Traditional missing value imputation techniques can be roughly classified into parametric imputation (e.g., the linear regression) and non-parametric imputation

---

\* This work is partially supported by Australian large ARC grants (DP0449535 DP0559536 and DP0667060), a China NSFC major research Program (60496327), a China NSFC grant (60463003) and a grant from Overseas Outstanding Talent Research Program of Chinese Academy of Sciences (06S3011S01).

(e.g., the Nearest Neighbor method or non-parametric regression imputation). The parametric regression imputation must correctly specify the parametric model for the data set. However, in many real applications, it is almost impossible to know what distribution can capture the attribute correlations within real data (i.e., setting the exact relation between the feature attributes and a target attribute is very difficult). An ill-specified model may result in highly biased and miscalculated results. An alternative approach for missing value imputation could be the use of non-parametric techniques. Using a non-parametric method is beneficial when the form of relationship between input and output data is not known apriori [21]. Kernel method, as one of the non-parametric techniques, which is widely used in machine learning and pattern recognition, is an effective method to deal with missing data because of its computationally efficient, robust and stable [22].

Most commonly used imputation methods include deterministic imputation and random imputation [15]. Wang and Rao [19, 20] show that the deterministic imputation method performance well in making inference for the mean of  $Y$ . Qin and Rao [11] have showed that one must use random imputation method in making inference for distribution functions and quantiles of  $Y$ . However, the most appropriate way to handle missing or incomplete data will depend upon how data is missing. Little and Rubin [9] classified missing data mechanisms into three categories as follows: Missing Completely at Random (MCAR), Missing at Random (MAR) and Nonignorable. In this paper, Different from above methods for imputation missing value, we propose a random regression imputation method (named as POP algorithm) which construct a kernel based nonparametric estimator to impute the missing values based on the missing mechanism MAR as well as MCAR, then we evaluate the performance of the POP in making inference for the parameters of the response variable  $Y$  about the mean and the distribution function.

The rest of this paper is organized as follows. In section 2, we give related work in missing value completion. In Section 3, we present our POP algorithm in detail. Experimental results are given in Section 4. Conclusions and future work are in Section 5.

## 2 Related Work

Currently, there are two kinds of methods that are used in the research topic of missing value completion: Machine learning based methods include decision tree imputation, case-wise deletion, lazy decision tree; Statistics-based methods include linear regression, robust Bayesian estimator and a new family of reconstruction problems for multiple images from minimal data [7].

But these methods are not completely satisfactory ways to handle missing value problems because these methods usually only handled with the discrete value in machine learning, such as C4.5. Continuous attributes are discretized before it is processed, which may be lost the true characteristic during converting process from the continuous values to discrete ones. [10] has reviewed the main missing data techniques (MDTs). They revealed that statistical methods have been mainly developed to manage survey data and proved to be very effective in many situations. However, the main problem of these techniques is the need of strong model

assumptions. That is, an assumption that the data follows a specific distribution is needed. However, these assumptions cannot well describe the data in many cases in practice, sometimes these assumptions are even misleading for modeling the data. An alternative approach for missing value imputation could be the use of non-parametric techniques for it needs fewer assumptions about the data distribution. In many real world applications, it is very common that the user has no idea about the data being discussed. As a consequence, non-parametric methods should be adopted for this kind of task of missing value imputation. Our POP algorithm is a non-parametric kernel method for missing value imputation, which is presented in section 3.

### 3 The POP Algorithm

Let  $X$  be an  $n \times d$ -dimensional vector and let  $Y$  be a variable influenced by  $X$ , we denote  $X, Y$  as attributes values and target attribute (class label) respectively, and  $X$  and  $Y$  are continuous. We assume that  $X$  has no missing values, while only  $Y$  has. To simplify the discussion, the dataset is denoted as  $(X_i, Y_i, \delta_i), i = 1, \dots, n$ , where  $\delta_i$  is an indicator of missing, i.e.,  $\delta_i = 0$  if  $Y_i$  is missing and  $\delta_i = 1$  if  $Y_i$  is not missing. In a real world database, we suppose that  $X$  and  $Y$  satisfy:

$$Y_i = m(X_i) + \varepsilon_i, i = 1, \dots, n. \quad (1)$$

Where  $m(\cdot)$  is an unknown function,  $\varepsilon_i$  is a random error with mean 0 and variance  $\sigma^2$ . In other words, we assume that  $Y$  has relation with  $X_i$ . The relation may be linear or nonlinear, but we cannot know it exactly. In this paper, we use

$$\hat{m}(x) = \frac{\sum_{i=1}^n \delta_i Y_i K\left(\frac{x - X_i}{h}\right)}{\sum_{i=1}^n \delta_i K\left(\frac{x - X_i}{h}\right) + n^{-2}} \quad (2)$$

Where  $\hat{m}(X)$  is the kernel estimate of  $m(X)$  and  $n^{-2}$  is introduced to avoid the case that the denominator vanishes, and  $h$  refers to bandwidth. We use  $\hat{m}(X_i)$  as imputed the missing value of  $Y_i$  in this paper.

In Equation 2,  $K(\cdot)$  is a kernel function. There are many commonly used forms of kernel functions, such as the Gaussian kernel, and the uniform kernel. In practice, there is not any significant difference using these kernel functions under the MAR and MCAR assumptions. In this paper, we adopt the widely used Gaussian kernel function. Silverman [23] turns out that the choice of bandwidth is much more important than the choice of kernel function. Small value of  $h$  make the estimate look ‘wiggly’ and show spurious features, whereas too big values of  $h$  will lead to an estimate which is too smooth in the sense that it is too biased and may not reveal structural features. There is no generally accepted method for choosing the window widths. Methods currently available include ‘subjective choice’ and automatic

methods such as the ‘‘plug-in’’, ‘cross-validation’ (CV), and ‘penalizing function’ approaches. In this paper, we use the method of cross-validation to choose  $c$ .

Let  $y_i^{(D)}$  and  $y_i^{(R)}, i \in s_m$  be values imputed for the missing data using deterministic and random imputation methods, respectively. Deterministic imputation [19, 20] uses  $\hat{m}_n(x)$  as the imputed value, i.e.  $y_i^{(D)} = \hat{m}_n(x), i \in s_m$ ; POP imputation uses  $y_i^{(R)} = \hat{m}_n(X_i) + \varepsilon_i^* = Y_i^{(D)} + \varepsilon_i^*, i \in s_m$ , as the imputed values, where  $\{\varepsilon_i^*\}$  is a simple random sample of size  $m$  with replacement from  $\{Y_j - \hat{m}_n(X_i), j \in s_r\}$ .

## 4 Analysis of Performance

In order to show the effectiveness of the proposed method, extensive experiments were done on simulation models as well as real dataset using a DELL Workstation PWS650 with 2G main memory, 2.6G CPU, and WINDOWS 2000.

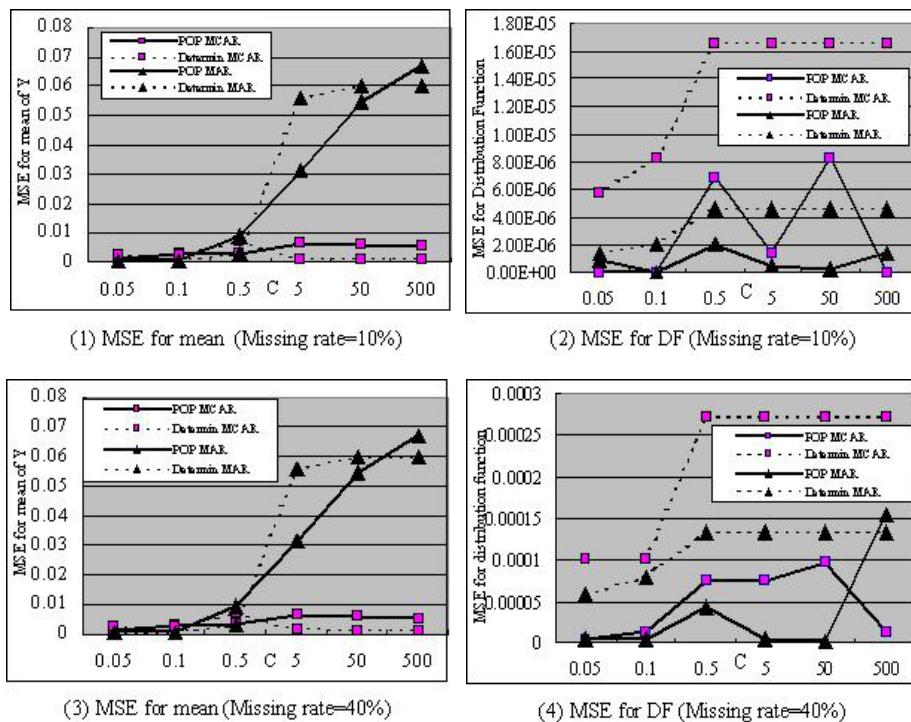
The dataset we adopted is the *Abalone* that downloaded from UCI machine learning repository [2], which contains 4177 instances in total and 9 attributes for each instance (non missing attribute values). These attributes give the *length*, *diameter*, *height* and other features of the abalone, and the last attribute *Rings* gives the age in years. For interest of space, we only select the attribute *length* as  $X$ , and the attribute *Rings* as  $Y$ . The detailed statistics of these two attributes are listed in Table 1.

**Table 1.** Statistics for attributes *Length* and *Rings* of *Abalone* dataset

Statistics	<i>Length</i>	<i>Rings</i>
Min	0.075	1
Max	0.815	29
Mean	0.524	9.934
Standard deviation	0.120	3.224

We use the missing mechanisms MCAR and MAR on  $Y$  at different missing rates of 10% and 40%. Then POP algorithm is utilized to fill the missing values of  $Y$ . The estimators of parameters of  $Y$  are also evaluated after imputation. Note that in real world applications, the missing values need to be imputed only once, i.e., repeat time = 1. In this paper, we do not present the performance for the estimators of quantile  $\theta_q$  as the estimators with different imputations are almost identical to the one with original data set. The reason is that the missing values of  $Y$  cause little effect for the estimator of quantiles in this case. We report the performance on  $\mu$  and  $\sigma$  in Figures (1)-(4).

We can see from Figures (1) and (3) that the POP and the deterministic one work well on the mean of  $Y$  under the real dataset, as the MSE values of both imputation methods are small and very close to each other. Figures (2) and (4) shows that the MSE for distribution function ( $y=10$ ) based on the POP are significantly smaller than those based on deterministic imputation, i.e. the POP performs much better than the deterministic counterpart on distribution function of  $Y$  under the missing mechanisms MCAR and MAR.



The POP still performs reasonably well under the circumstance of high missing rate, as the MSE for estimators of  $\mu$  and  $\theta$  remains small when missing rate of  $Y$  is changed from 10% to 40%. This trend can be seen by comparing Figures (1), (2) with (3), (4).

## 5 Conclusions and Future Work

Because learning from incomplete data is a challenging and practical issue, we have designed a kernel based missing data imputation for the problem in this paper. In nonparametric regression settings, we have shown that our POP algorithm (random regression imputation) works well in making inference on all parameters. We have also illustrated that deterministic regression imputation only works for the mean of  $Y$ , but not for the distribution function and quantiles of  $Y$ . That is, when we want to make inference on the mean of  $Y$ , we can use deterministic imputation or random imputation; when we want to make inference on the distribution function, we must use random imputation. Our experiments have also demonstrated that our POP algorithm generates an inference on the above parameters much better than deterministic regression imputation.

In future work, we plan to use the POP algorithm to make inference for nonparametric model with fixed design points  $X_i$ , and the partial linear model,

$$Y_i = \beta' X_i + m(T_i) + \varepsilon_i.$$

## References

1. Batista, G.A., et al. (2003). An Analysis of Four Missing Data Treatment Methods for Supervised Learning. *Applied Artificial Intelligence*, 17(5-6): 519-533.
2. Blake, C. L., and Merz, C. J (1998). *UCI Repository of machine learning database*. [<http://www.ics.uci.edu/~mlearn/MLRespository.html>] Irvine, CA: university of California, Department of Information and Computer Science.
3. Chen, SM, Chen HH (2000). Estimating null values in the distributed relational databases environments. *Cybernetics and Systems: An International Journal*, 31: 851-871
4. Chen, SM, Huang, CM (2003). Generating weighted fuzzy rules from relational database systems for estimating null values using genetic algorithms. *IEEE Transactions on Fuzzy Systems*, 11:495-506
5. Gessert G. (1991). Handling Missing Data by Using Stored Truth Values. *SIGMOD Record*, 20(3): 30-42.
6. Han, J., Kamber, M. (2000). *Data Mining: Concepts and Techniques*, Morgan Kaufmann.
7. Kahl, F., Heyden, A., Quan, L. (2001). Minimal Projective Reconstruction Including Missing Data. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(4): 418-424.
8. Lakshminarayanan, K, Steven A. Harp, Robert P. Goldman, Tariq Samad (1996). Imputation of Missing Data Using Machine Learning Techniques. *KDD-1996*: 140-145.
9. Little, R.J.A. and Rubin, D.A. (1987). Statistical analysis with missing data. New York: John Wiley and Sons.
10. Magnani M. (2004). *Techniques for Dealing with Missing Data in Knowledge Discovery Tasks*, In: <http://magnanim.web.cs.unibo.it/index.html>.
11. Qin, Y. S. and Rao, J. N. K. (2004). *Confidence intervals for parameters of the response variable in a linear model with missing data*. Technical Report, School of Math and Statistics, Carleton University.
12. Pawlak, W., (1993). Kernel classification rules from missing data. *IEEE Transactions on Information Theory*, 39(3): 979-988.
13. Pesonen, E., Matti Eskelinen and Martti Juhola (1998). Treatment of missing data values in a neural network based decision support system for acute abdominal pain. *Artificial Intelligence in Medicine*, 13(3): 139-146.
14. Ramoni, M., et al. (2001). Robust Learning with Missing Data. *Machine Learning*, 45(2): 147-170.
15. Rao, J. N. K. (1996). On variance estimation with imputed survey data. *J. Amer. Statist. Assoc.*, 91: 499-520.
16. Schafer, J.L., and Graham, J.W., (2002). Missing Data: Our View of the State of the Art. *Psychological Methods 2002*, Vol. 7, No. 2, 147-177.
17. Shichao Zhang, Chengqi Zhang and Qiang Yang (2004). Information Enhancement for Data Mining. *IEEE Intelligent Systems*, 19(2): 12-13.
18. Shichao Zhang, Zhenxing Qin, Charles X. Ling and Shengli Sheng (2005). "Missing is useful": missing values in cost-sensitive decision trees. *IEEE Transactions on Knowledge and Data Engineering*, Vol. 17 No. 12: 1689-1693.

19. Wang, Q., Rao, J. N. K. (2002a). Empirical likelihood-based inference in linear models with missing data. *Scand. J. Statist.*, 29: 563-576.
20. Wang, Q., Rao, J. N. K. (2002b). Empirical likelihood-based inference under imputation for missing response data. *Ann. Statist.* 30: 896-924.
21. Lall, U., and A. Sharma. (1996). A nearest-neighbor bootstrap for resampling hydrologic time series. *Water Resource. Res.* 32: 679-693.
22. John, Shawe-Taylor and Nello Cristianini (2004). *Kernel Methods for Pattern Analysis*. Cambridge Press.
23. Silverman, BW. (1986). *Density Estimation for Statistics and Data Analysis*. Chapman and Hall, New York.

# Expediting Model Selection for Support Vector Machines Based on an Advanced Data Reduction Algorithm

Yu-Yen Ou<sup>1</sup>, Guan-Hau Chen<sup>1</sup>, and Yen-Jen Oyang<sup>2</sup>

<sup>1</sup> Graduate School of Biotechnology and Bioinformatics,  
Department of Computer Science and Engineering,  
Yuan-Ze University, Chung-Li, Taiwan

<sup>2</sup> Department of Computer Science and Information Engineering,  
National Taiwan University, Taipei, Taiwan

**Abstract.** In recent years, the support vector machine (SVM) has been extensively applied to deal with various data classification problems. However, it has also been observed that, for some datasets, the classification accuracy delivered by the SVM is very sensitive to how the cost parameter and the kernel parameters are set. As a result, the user may need to conduct extensive cross validation in order to figure out the optimal parameter setting. How to expedite the model selection process of the SVM has attracted a high degree of attention in the machine learning research community in recent years. This paper proposes an advanced data reduction algorithm aimed at expediting the model selection process of the SVM. Experimental results reveal that the proposed mechanism is able to deliver a speedup of over 70 times without causing meaningful side effects and compares favorably with the alternative approaches.

## 1 Introduction

The support vector machine (SVM) was first proposed by Vapnik [1] and has since attracted a high degree of interest in the machine learning research community. However, it has also been observed that, for some data sets, the classification accuracy delivered by the SVM is very sensitive to how the cost parameter and the kernel parameters are set. As a result, the user may need to conduct extensive cross validation in order to figure out the optimal parameter setting. This process is commonly referred to as model selection [2].

One practical issue concerning model selection with the SVM is that this process could be very time-consuming if the data set is large and an extensive search in the parameter space is conducted. In recent years, several studies have been conducted to address this issue and all these studies share a common ground aimed at reducing the search space of parameter combinations [3,4,5].

In this paper, we propose a novel approach to expedite the model selection process of the SVM based on an advanced data reduction mechanism. The study has been motivated by the observation that the learning algorithm of the SVM

is aimed at figuring out the boundary that separates the two different classes of training instances in a hyperspace, subject to certain optimization criteria. Accordingly, the exact profile of the boundary can be figured out by observing only the training instances located in the proximity of the boundary. In other words, those training instances that are far away from the boundary essentially play no role in determining the boundary and can be removed. However, we must also consider the side effect of data reduction. That is, it may cause degradation of classification accuracy if some essential training instances are mistakenly removed. Fortunately, the experiments that we have conducted show that extensive data reduction can be applied on the data set fed into the model selection process of the SVM without causing meaningful side effect on classification accuracy. In our first stage of study on this issue [6], we employed a conventional data reduction mechanism and observed an average reduction rate of over 80% of the input instances removed for the three datasets in Table 1. This level of reduction in turn leads to an average speedup of the model selection process by over 20 times. On the other hand, we observed no degradation of classification accuracy in two out of the three datasets and only a degradation of 0.6% in the third dataset. In our follow-up study, we have developed an advanced data reduction mechanism based on the efficient kernel density estimation algorithm that we have recently proposed [7]. With the kernel density estimation (KDE) based data reduction mechanism, the average reduction rate for the three datasets in Table 1 is raised to 90% of the input instances being removed, which in turn leads to another 3 times of speedup in the model selection process. Since it is generally believed that the time complexity of the model selection process of the SVM is over  $O(n^2)$ , the additional level of reduction rate achieved could mean a much higher level of speedup for large datasets. The experiments conducted to evaluate the effects of the KDE based data reduction mechanism again show no meaningful side effect on classification accuracy.

## 2 The KDE Based Data Reduction Mechanism

The design of the data reduction based mechanism proposed in this paper for expediting the model selection process of the SVM is motivated by the observation that the learning algorithm of the SVM is aimed at figuring out the boundary that separates the two different classes of training instances in a hyperspace subject to certain optimization criteria. Accordingly, the profile of the boundary can be accurately figured out by the training instances located in the proximity of the boundary.

As mentioned earlier, in our first stage of study on this issue, we employed a conventional data reduction mechanism to expedite the model selection process of the SVM. The conventional data reduction mechanism operates by first sorting the instances in the training dataset in the descending order according to the distance of each instance to its *nearest enemy* [8], which is defined to be the nearest neighbor belonging to a different class. With the sorted list, the data reduction algorithm then examines the instances one by one. An instance is

considered as redundant and is removed if the instance and all its  $k$  nearest neighbors in the residue dataset belong to the same class. Here,  $k$  is a parameter to be set and the residue dataset refers to the dataset generated by removal of the redundant instances that have been examined earlier. In the following part of this paper, this conventional data reduction mechanism is referred to as the NE (nearest enemy) based data reduction algorithm.

Experimental results show that with parameter  $k$  set to 3, the NE based data reduction mechanism is capable of removing over 80% of the training instances in the three datasets listed in Table 1 and reduces the execution time of the model selection process by over 95% without causing any meaningful side effect on classification accuracy. In our follow-up study, we have developed an advanced data reduction mechanism based on the efficient kernel density estimation (KDE) algorithm [7]. Also, the detail of KDE algorithm can be found in [7]. In the following part of this paper, the advanced data reduction mechanism is referred to as the KDE based data reduction algorithm.

The efficient kernel density estimation algorithm treats each class of training instances as a set of random samples taken from a probability distribution and construct an approximate probability density function accordingly. Since the approximate probability density function is continuous and smooth, it can be expected that the function values at the instances in the outer regions of the distribution are generally lower than the function values of the instances in the inner regions of the distribution. Accordingly, the function values at the instances can be exploited to sort the instances in the descending order.

With the instances in the dataset sorted, the KDE based data reduction algorithm then examines the instances one by one for their redundancy. In the KDE based data reduction mechanism, two conditions are employed to determine whether an instance is redundant and an instance is considered as redundant if either one condition is met. The first condition is that the instance and all its  $k$  nearest neighbors in the residue dataset belong to the same class. The second condition is that the value of the approximate probability density function at the instance is higher than the average function value multiplied by a factor  $r$ .

As far as the time complexity of the KDE based data reduction mechanism is concerned, for each of the  $n$  training instance, we need to compute the function value of the approximate probability density function. If the kd-tree structure is employed, then the average time complexity of this process is  $O(n \log n)$  [9]. Then, we need to sort the functional values at the training instances and the sorting can be carried out in  $O(n \log n)$ . Finally, we need to remove a training instance in the kd-tree if the instance is redundant. With the kd-tree, the removal of an instance can be carried out in  $O(\log n)$ . Therefore, the average time complexity of the data reduction mechanism is  $O(n \log n)$  if  $k$  is treated as a constant [7].

### 3 Experimental Results

Tables 1-3 show how the KDE based data reduction mechanism performs in comparison with the alternative approaches in various aspects. As Table 1 reveals,

**Table 1.** Execution time of the model selection process with alternative mechanisms in seconds

Dataset	The original time in seconds	The two-line method time in seconds	% of the original time	The NE based method time in seconds	% of the original time	The KDE based method time in seconds	% of the original time
satimage	29052	3244	11.1%	1791	6.2%	632	2.2%
letter	192358	15716	8.2%	14528	7.6%	3915	2.0%
shuttle	252018	7887	3.1%	63	0.2%	10.7	0.04%
Average			7.5%		4.6%		1.4%
Speedup		1		13.3		21.7	
							71.4

**Table 2.** Generalization accuracy with alternative mechanisms

Dataset	original method	two-line method	NE based method	KDE based method
satimage	91.8%	91.55%	91.2%	90.95%
letter	97.82%	96.54%	97.82%	97.16%
shuttle	99.92%	99.81%	99.92%	99.92%
Average	96.51%	95.97%	96.31%	96.01%

**Table 3.** The numbers of training instances left after data reduction is applied

Dataset	The original # of training instances	SVM		The NE based method		The KDE based method	
		# of SV	% of original training data	# of instances	% of original training data	# of instances	% of original training data
satimage	4435	1689	38.0%	1167	26.3%	737	16.6%
letter	15000	8931	59.5%	4027	26.8%	2002	13.3%
shuttle	44500	287	0.64%	272	0.61%	81	0.18%
Average			32.7%		17.9%		10.0%

all three mechanisms can significantly reduce the execution time taken to carry out the model selection process, while all three cause slight degradation of classification accuracy. An overall observation is that both the KDE based data reduction mechanism and the NE based data reduction mechanism deliver higher levels of speedup and cause less degree of loss of classification accuracy than the two-line approach. In addition, the KDE based mechanism is able to deliver a higher level of speedup than the NE based mechanism. Table 3 presents further insight about the effects of the KDE based mechanism. As shown in Table 3, with the KDE based mechanism applied, the average number of training instances remaining in the data set is about one third of the average number of training instances remaining after the NE based mechanism has been applied.

## 4 Discussion and Conclusion

This paper proposes a novel approach to expedite the model selection process of the SVM based on an advanced data reduction mechanism. Experimental results reveal that the proposed KDE based mechanism is able to deliver a significant

speedup without causing meaningful sideeffects and compares favorably with the alternative approaches. In particular, in the experiments with the three benchmark datasets employed in this paper, the KDE based mechanism achieves an average speedup of over 70 times. Since it is generally believed that the time complexity of the model selection process of the SVM is over  $O(n^2)$ , a higher level of speedup could be observed if the dataset involved were larger than the benchmark datasets employed in the experiments. rate achieved could mean a much higher level of speedup for large datasets. As far as the execution time of the KDE based data reduction mechanism is concerned, the average time complexity is  $O(n \log n)$ , where  $n$  is the number of instances in the input dataset.

## References

1. Cortes, C., Vapnik, V.: Support-vector network. *Machine Learning* **20** (1995) 273–297
2. Hsu, C.W., Lin, C.J.: A comparison of methods for multi-class support vector machines. *IEEE Transactions on Neural Networks* **13** (2002) 415–425
3. Chapelle, O., Vapnik, V., Bousquet, O., Mukherjee, S.: Choosing multiple parameters for support vector machines. *Machine Learning* **46** (2002) 131–159
4. DeCoste, D., Wagstaff, K.: Alpha seeding for support vector machines. In: Proceedings of International Conference on Knowledge Discovery and Data Mining (KDD-2000). (2000)
5. Keerthi, S.S.: Efficient tuning of SVM hyperparameters using radius/margin bound and iterative algorithms. *IEEE Transactions on Neural Networks* **13** (2002) 1225–1229
6. Ou, Y.Y., Chen, C.Y., Hwang, S.C., Oyang, Y.J.: Expediting model selection for support vector machines based on data reduction. In: Proc. IEEE International Conference on Systems, Man and Cybernetics (SMC2003). (2003) 786–791
7. Oyang, Y.J., Hwang, S.C., Ou, Y.Y., Chen, C.Y., Chen, Z.W.: Data classification with radial basis function networks based on a novel kernel density estimation algorithm. *IEEE Transactions on Neural Networks* **16** (2005) 225 – 236
8. Wilson, D.R., Martinez, T.R.: Reduction techniques for instance-based learning algorithms. *Machine Learning* **38** (2000) 257–286
9. Bentley, J.: Multidimensional binary search trees used for associative searching. In: *Communications of the ACM*. (1975) 18(9):509–517

# Study of the SMO Algorithm Applied in Power System Load Forecasting

Jingmin Wang and Kanzhang Wu

Department of Economy and Management, North China Electric power University,  
Baoding 071003  
jingminwang126@126.com, changcheng2004-9@163.com

**Abstract.** A new methodology on the algorithm of sequential minimal optimization (SMO) for power system load was presented. In order to solve the problem that support vector machines (SVM) can not deal with large scale data, this paper introduces the modified algorithm of SMO to increase operational speed by use of a single threshold value. Adopting the actual data from the distribution network of a certain domestic city, and the load is forecasted by use of support vector regression (SVR) which is based on the modified SMO algorithm and proper kernel function. The forecasted results are compared with those SVR employing quadratic programming (QP) optimization algorithm and BP artificial neural method, and it is shown that the presented forecasting method is more accurate and efficient.

**Keywords:** Support vector machine; Sequential minimal optimization; kennel function; load forecasting.

## 1 Introduction

STLF (Short-term Load Forecasting) plays an import role in the reliability of electric power system and the operation of economy [1]. Especially as the development of the electric market, people attach more and more importance to STLF. The researching stuff both from domestic and abroad have been making large amounts of studies and have put forward various kinds of forecasting method such as Time Series Method, Neural Network method, expert system and Artificial Neural Method etc. But because Load of electric power system is a multidimensional non-linear chaotic system, the above-mentioned methods have limiting factors in their astringency or applicability.

Vapnik [2] and his co-workers proposed SVM which is considered as a novel method of machine learning. It overcomes the inherent drawback of neural network , such as local minimal point, over learning and architecture selection and type over-depending on experience. The SMO algorithm proposed by Platt [3] is for SVM classifier design. Because SMO uses a sub-problem of size two, each sub-problem has an analytical solution. Thus SVM could be optimized without a QP solver. The fine computational speed and simple implementation are the significant features of the SMO algorithm. And then Alex Smola generalized SMO algorithm to recursive problems, which provided possibility for resolving the huge data recursive questions by

support vector machines. Many scholars revised Smola's SMO algorithm [4,5]. In this paper, we introduce a feasible SMO arithmetic method.

## 2 Support Vector Regression (SVR)

Consider a set of data points,  $\{(x_i, y_i)\}$  ( $i = 1, 2, \dots, m$ ), such that  $x_i \in \Re^d$  is an input and  $y_i$  is a target output. The basic idea of support vector regression is to map the input vector into high dimensional feature space by nonlinear mapping function and then to perform linear regression in the feature space. This transformation is realized by Kernel function. It can be written as follows:

$$f(x, w) = w\Phi(x) + b \quad (1)$$

The coefficients and are estimated by minimizing

$$\begin{aligned} \text{Minimize: } & E(w) = C \sum_{i=1}^n (\xi_i + \xi_i^*) + \frac{1}{2} \|w\|^2 \\ \text{Subjected to: } & \begin{cases} y_i - f(x_i, w) \leq \varepsilon + \xi_i \\ f(x_i, w) - y_i \leq \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0 \end{cases} \end{aligned} \quad (2)$$

Slack variables  $\xi_i$  and  $\xi_i^*$  can be introduced when data can't be estimated by the function  $f$  under the precise  $\varepsilon$ .

Introducing Lagrange multipliers and the model output is given :

$$f(x, \alpha) = \sum_{i=1}^N (\alpha_i^* - \alpha_i) K(x_i, x) + b \quad (3)$$

## 3 SMO Algorithm for Regression

### 3.1 SVR Based on SMO

If  $\lambda_i = \alpha_i^* - \alpha_i$ ,  $|\lambda_i| = \alpha_i^* + \alpha$ , the new value for  $\lambda_i$  will obey the box constraint  $-C < \lambda_i < C$ ,  $i = 1, 2, \dots, N$ . Substitute  $\lambda_i$  and  $|\lambda_i|$  into (3) and (4), then they can be written as follows:

$$\text{Minimize: } L_p(\lambda) = \varepsilon \sum_{i=1}^N |\lambda_i| - \sum_{i=1}^N \lambda_i y_i + \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \lambda_i \lambda_j k_{ij}, \quad (5)$$

$$\text{Subject to: } \sum_{i=1}^N \lambda_i = 0 \quad -C < \lambda_i < C$$

$$\text{And } f(x, \lambda) = \sum_{i=1}^N \lambda_i K(x_i, x) + \lambda_0, \quad (6)$$

Our goal is to express analytically the minimum of (5) as a function of two parameters. Let these two parameters have indices  $a$  and  $b$  so that  $\lambda_a$  and  $\lambda_b$  are the unknowns. Note that a superscript \* indicate that values are computed with the old parameter values. This means that these portions of the expression will not be a function of the new parameters.

Because (5) can be expressed analytically as a function of two parameters, we can get the analytic solution. However (5) has absolute value sign when computing partial derivative. Nevertheless, if we take  $d|x|/dx = \text{sgn}(x)$ , the resulting derivative is algebraic.

$$\lambda_b = \lambda_b^* + \frac{1}{\eta} [y_b - y_a + f_a^* - f_b^* + \epsilon(\text{sgn}(\lambda_a) - \text{sgn}(\lambda_b))] \quad (7)$$

where  $\eta = (k_{bb} + k_{aa} - 2k_{ab})$ . (7) is  $\lambda_b$  new rule.

The Computation of the Threshold b:

If  $\lambda_a$  is satisfied with the constraint, then  $f_a = y_a$ ,

$$\lambda_0^a = y_a - f_a^* + (\lambda_a^{old} - \lambda_a^{new})k_{aa} + (\lambda_b^{old} - \lambda_b^{new})k_{ab} + \lambda_0^* \quad (8)$$

If  $\lambda_b$  is satisfied with the constraint, then  $f_b = y_b$ ,

$$\lambda_0^b = y_b - f_b^* + (\lambda_a^{old} - \lambda_a^{new})k_{ab} + (\lambda_b^{old} - \lambda_b^{new})k_{bb} + \lambda_0^* \quad (9)$$

Otherwise:  $\lambda_0^{new} = 0.5(\lambda_0^a + \lambda_0^b)$

$$\lambda_i = 0 \Leftrightarrow |y_i - f_i| \leq \epsilon$$

$$\begin{aligned} \text{KKT Conditions: } & -C < \lambda_i \neq 0 < C \Leftrightarrow |y_i - f_i| \approx \epsilon \\ & |\lambda_i| = C \Leftrightarrow |y_i - f_i| \geq \epsilon \end{aligned} \quad (10)$$

If one or both multipliers violate the KKT condition, the algorithm will go on operating. When no parameter violates any KKT condition, the global minimum can be achieved within machine precision.

### 3.2 SMO Algorithm of Completion

1. Select the first multiplier: The external loop first iterates over the non-bound training subset. If a sample violates the KKT conditions, it is qualified for immediate optimization. If there are no such samples, then iterates over the entire training set.
2. Choose the sample with maximum of  $|(\mathbf{E}_1 - \mathbf{E}_2)/\eta|$  as the second multiplier;
3. Compute  $\lambda_b^{raw}$ : Set  $\lambda_a^{new} = \lambda_a^{old} + \lambda_b^{old} - \lambda_b^{new}$
4. Updating threshold: if  $f_a = y_a$ , set  $\lambda_0^{new}$  as equation (8); if  $f_b = y_b$ , set  $\lambda_0^{new}$  as equation (9); otherwise,  $\lambda_0^{new} = 0.5(\lambda_0^a + \lambda_0^b)$

## 4 Short-Term Load Forecasting Based on SVR

There are great differences on the load between workdays and weekends. Also the weather influences the load of forecast day, so the training simple should include weather data. The kennel function uses the Radial Base Function. In the SVR algorithm, parameter  $c$  and  $\varepsilon$  influence the generalized performance of SVR. The value of parameter  $C$  is selected between 10 and 100, when the  $C$  value is over 100, it can result in the owed-study phenomenon; The bigger  $\varepsilon$  is, the less the number of SVM is and the lower the estimated precision of the function is. So the value is generally chosen between 0 and 1. In this paper,  $C$  is 50 and  $\varepsilon$  is 0.011.

### 4.1 The Process of Load Forecasting

1. Smooth pretreatment of historical data;
2. Establish forecast sample, including historical load data, temperature data, date type attribute etc. and establish the system model.
3. Calculate  $\lambda_i, \lambda_0$ , using the improvement SMO algorithm
4. Substitute  $\lambda_i, \lambda_0$  into (5) and forecast the load data in some time of the near future making use of forecasting samples.

### 4.2 Computational Example

Power load data in Shijiazhuang region is used to prove the effectiveness of the model. The power load data from 0:00 at 5/8/2005 to 12:00 at 5/17/2005 are as training sample, and the power load data from 13:00 to 20:00 at 5/17/2005 as testing sample. Average relative error is used to evaluate the accuracy of forecasting, and make comparison of all kinds of training time and MAE, of which the result is shown in Table1.

**Table 1.** Comparison of prediction results of SVR , BP and QP

index	MAE%	RMSE%	Training time /s
SVR	2.25	1.6	24
BP	3.10	2.6	86
QP	2.83	1.94	157

Basically the forecasting error is under 3%, and even 80% is under 2%. The results of the study showed that the presented forecast method has higher forecast accuracy than BP artificial neural method and more training speed than SVR employing QP method.

## 5 Conclusions

In the paper, a new short-term load forecasting method is presented to enhance accuracy and cut down training time. In this method, the global QP problem can be broken

down into a sequence of optimization sub-problems of size two. And the results show that the SVR which based on SMO arithmetic has more accuracy than BP artificial neural method and more training speed than SVR employing QP method. It not only resolves the problem that BP method is easy fall to local minimal point and also provides one feasible solution to large datasets.

## References

1. Li Yuancheng,Fang Tingjian,Yu Erkeng: Study of support vector machines for short term load forecasting . Proceedings of the CSEE, Vol.23(2003)55-59.
2. Drucker H., Burges C.J.C., Kaufman L., Smola A., Vapnik V: Support Vector Regression Machines. In Mozer M., Jordan M.,Jordan M.,and Petsche T., editors, Advances in Neural Information Processing Systems 9.Cambridge, MA,1997.MIT Press, Page(s):155-161.
3. Platt.J: Fast training of support vector machines using sequential minimal optimization. In B.Schölkopf, C.Burges, and A.Smola,editors, Advances in Kernel Methods-Support vector Learning,MIT Press,1998.
4. Osuna E, Freund R, Girosi F: An improved training algorithm for support vector machine. Proc. The 1997 IEEE workshop on neural networks for signal processing. Amelea Island, FL,1997,276-285.
5. Gary William Flake, Steve Lawrence. Efficient SVM Regression Training with SMO. Machine Learning, vol.46(2002)271-290.

# Filtering Objectionable Image Based on Image Content

Zhiwei Jiang<sup>1</sup>, Min Yao<sup>1,2</sup>, and Wensheng Yi<sup>1</sup>

<sup>1</sup> College of Computer Science, Zhejiang University, Hangzhou 310027, P.R. China

<sup>2</sup> State Key Lab. For Software Engineering, Wuhan University, Wuhan 430072, P.R. China  
j\_z\_w@zju.edu.cn

**Abstract.** This paper proposes an effective system to detect adult image. We take this task as a two-class pattern classification problem. The system first applies histogram color model to detect the skin regions, then extracts color, texture, shape features from skin regions, after that, the features is fed to a SVM to determine whether the input image is benign or not. Experimental results show that the proposed method can achieve a satisfactory classification performance with high speed, which is suitable for real-world applications.

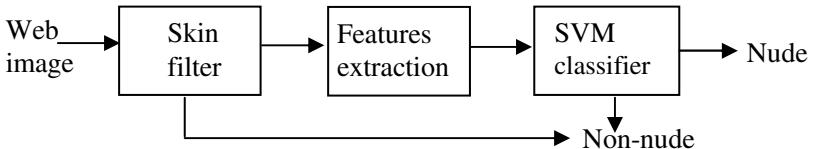
## 1 Introduction

Pornographic content on web is harmful for under-age net surfer. It calls for effective way to keep children from accessing the objectionable content. Maintaining a list of URL is a feasible way, the inconvenience is that the lists need to be updated frequently by human. Another method is to utilize the text on web page, this method may fail to work when there is no pornographic text or text is displayed in picture mode. Since nude images are main characteristic of adult web site, filtering nude images by virtue of image analysis technique is a promising way.

Forsyth's system [1] used a set of grouping rules to identify naked people. Since there is innumerable human posture, a comprehensive grouping rules may be hard to construct; Wang's WIPE system [2] adopted matching techniques of CBIR domain to screen objectionable image. In his system, features repository is a crucial factor that will affect accuracy and speed; Jones's system [3] yield good performance purely based on color attributes in nude detection. But only color cue is insufficient to capture the whole features of images.

For a practical system, both accuracy and speed are required, to achieve this goal, we refine objectionable image detection techniques based on the following three points: (1)Through Jones's work we believe that color can be a powerful cue to detect skin regions. (2)Many wildlife and landscape pictures will resemble human skin in color, under which shape and texture cues is necessary for further discrimination. (3)An appropriate learning scheme and classifier can improve both accuracy and speed.

The basic structure of proposed system is shown as Fig. 1. The remainder of this paper is organized as follows, Section 2 introduces color models and skin filter, Features extraction are described in section 3, Classifier are given in section 4, Experiment results are discussed in section 5, The work is concluded in section 6.



**Fig. 1.** Basic structure of nude image detection system

## 2 Skin Detection

For skin filters, we adopt the techniques proposed by Jones and Rehg [3]. The skin pixel likelihood ratio is  $L(rgb) = P(rgb|skin)/P(rgb|\neg skin)$ , here  $P(rgb|skin)$  and  $P(rgb|\neg skin)$  is the probability of a given color with  $rgb$  value belongs to the skin and non-skin classes. A particular pixel with  $rgb$  value is classified as skin if  $L(rgb) \geq \theta$ ,  $\theta$  is a threshold which can be adjusted to trade-off between correct detections and false positives. Since we don't solely depend on color cue to detect nude images, we wish to accept as much of the probable skin pixels as possible, So we choose to use a lower threshold value at skin filter stage through it may also allow some non-skin pixels to pass the skin filter.

## 3 Feature Description and Extraction

Based on the fact that there is a strong correlation between large patches of skin region and nude images, some researchers use only color cue to separate nude images and non-nude images [3]. We believe that relying solely on color information to identify nude image is insufficient, the main reason lies in: (1)Many wildlife and



**Fig. 2.** Typical images marked mistakenly as nude by skin filter relying solely on color cue.  
 (a)tiger with skin resemble human's in color. (b)image after skin filter with threshold set to 0.6  
 (c)desert with color similar to human skin (d) image after skin filter with threshold set to 0.6.

landscape pictures will resemble human skin in color. (2)Highly saturated or shadowed skin may be missed. (3)Images close to each other in color may actually be unrelated in semantics. To capture the main characteristics of image, we computed a joint feature vector from the output of the skin detector, the feature vector consists of color, texture and shape information.

### 3.1 Color Information

The color features are computed directly from the output of the skin detector, they are: percentage of pixels detected as skin, number of connected components of skin, average probability of the skin pixels, the mean and deviation of R,G,B color value of skin pixels and the entropy of R,G,B color value of skin pixels. Then the color feature vector with 12 features is constructed.

### 3.2 Texture Information

Texture is another important primitive visual cue, different textures in an image are usually very apparent to a human observer, but up to day there still lack good mathematical definition for diverse texture. According to the work in [4], The Gabor filter masks can be considered as orientation and scale tunable edge and line detectors. The statistics of these microfeatures in a given region can be used to characterize the underlying texture information. Gabor wavelet based texture is robust to orientation and illumination change. It is a powerful tool to extract texture features. After filtering image with Gabor wavelet at three scales and four orientations, we use mean and the standard deviation to character texture region, then twelve texture features is extracted. For RGB image, each color channel contains twelve features, so the texture feature vector with 36 features is now constructed.

### 3.3 Shape Information

To human observer, Shape is one of the most important feature of object, how to capture this spatial information is a tricky task. Traditionally, moments have been widely used in pattern recognition applications to describe the geometrical shapes [5,6]. Good representation of shape should be independent of position, size, and orientation. We get seven translation, rotation, and scale invariant moments as introduced in [5], these moments are computed using all the information of shape boundary and interior region. An improved moment given in [7] are computed only on the shape boundary. We use Daubechies-4 wavelet to decomposed image into four frequency bands: LL, HL, LH and HH. Then applied a zero-crossing detector in HL, LH and HH band to detect the vertical, horizontal and diagonal edges:  $E_v$ ,  $E_h$  and  $E_d$ , the integrated edge is obtained as  $E = \sqrt{E_v^2 + E_h^2 + E_d^2}$ . We then adopt the method in [7] to get  $4 \times 7 = 28$  invariant moments from  $E$ ,  $E_v$ ,  $E_h$ ,  $E_d$  edge image, so the shape feature vector with  $35 = 28 + 7$  features is now constructed.

## 4 Images Classification Based on SVM

The image feature extraction stage, described in the previous section, produce a feature vector with 83 features for each image. In this stage, our task is to find the decision rule that will optimally separates objectionable image from benign one. Traditional learning techniques can do well in minimizing the training error, but it does not imply a small expected test error, In other words, their carry little

information about the generalization ability of a learning machine. SVM is a new kind of machine learning method based on Statistical learning theory proposed by Vapnik [8], SVM pursues structure risk minimum (SRM) instead of empirical risk minimum (ERM), this principle can guarantee generalization ability of system. SVM is designed by finding a optimal hyperplane to solve two-class problems, Our classification task is a binary classification problems to distinguish nude image from non-nude one, So SVM is an ideal classifier for our classification task.

## 5 Experimental Results

We used manually classified 300 nude images and 1200 non-nude images to train a SVM classifier. all images are obtained from Internet. Nude images include Caucasians, Asians and Blacks. Non-nude images are diverse, including human, wildlife, buildings, scenery and so on. SVM training process take the SMO algorithm introduced in [9]. The classifier was then tested on a set of 2989 test images with 338 adult image and 2651 benign images. Recall and accuracy are used as the performance measures, Experiment results show that the proposed method achieve a satisfactory performance with high speed, it obtain 97.3% recall and 98.09% accuracy. Table 1 summarizes the results.



**Fig. 3.** Typical mistakenly classified images. (a) close-up face image (b) little undressed pornographic image (c) poor exposed image.

Mistaken classification occurs mostly in three cases: close-up face image, little undressed pornographic image and poor exposed image as shown in figure 3. To deal with these cases, face detection and posture detection techniques are required. The average processing speed is about 14 images/s on PC (Pentium IV 2.66GHZ CPU, 512M RAM). The high speed makes our algorithm practical for real-world applications.

**Table 1.** Performance of SVM classification

	Number of correctly classified images	Recall	Accuracy
Nude images(338)	329	97.3%	98.09%
Non-nude images(2651)	2603	N/A	

## 6 Conclusion

Our work is to develop techniques to filter adult images on Internet based on image content. We first applies histogram color model to detect the skin regions, Then a joint features vector contains color, texture and shape information are extracted. We utilize Gabor wavelets to extract texture features, Daubechies-4 wavelet to get edge image, invariant moments to extract shape features. After joint feature vector is composed, It is fed to a SVM to classify whether input image is nude or not. Experiment results show that the proposed method achieves a satisfactory performance with high speed, which make it suitable for real-world applications.

At present, our SVM classifier is fixed once its train process is finished, this is not enough to classify infinite new images. One improvement is to use much large collections of training images, the other improvement is to incorporate a long-term learning scheme into our system, like relevance feedback and SVM active learning techniques. The latter one is what we are going to investigate. We will also incorporate face and posture detection technique into our system in new version.

## Acknowledgement

This work is supported by the Specialized Research Fund for the Doctoral Program of Higher Education (SRFDP) under contract 20040335129, and the Natural Science Foundation of Zhejiang Province under contract Z104267.

## References

1. Margaret Fleck, David A. Forsyth, Chris Bregler: Finding Naked People. In European Conf. on Computer Vision. (1996)Vol. 2: 593-602.
2. J. Z. Wang, G Wiederhold and O. Firschein: System for Screening Objectionable Images. Computer Communications. (1998)Vol. 21:1355–1600.
3. M. J. Jones and J. M. Rehg: Statistical Color Model With Application to Skin Detection. Proceedings of CVPR. (1999)274-280, June.
4. Ma, W.Y., Manjunath, B.S.: Texture features and learning similarity. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. (1996)425-430.
5. M. K. Hu: Visual Pattern Recognition by Moment Invariants. IEEE Transactions on information theory. (1962)Vol. 8, Issue 2, 179-187.
6. R. J. Prokop and A. P. Reeves: A survey of moment based techniques for unoccluded object representation and recognition. Graphical Models and Image Processing. (1992)Vol. 54, No. 5, pp. 438-460, September.
7. C.C. Chen: Improved moment invariants for shape discrimination. Patten Recognition. (1993)Vol. 26: 683-686.
8. V. Vapnik: The Nature of Statistical learning Theory. New York, Springer-Verlag, 1995.
9. John C. Platt: Fast Training of Support Vector Machines using Sequential Minimal Optimization. <http://www.reaserach.microsoft.com/~jplatt>

# MRA Kernel Matching Pursuit Machine

Qing Li, Licheng Jiao, and Shuyuan Yang

Institute of Intelligent Information Processing and National Key Laboratory  
for Radar Signal Processing, Xidian University, Xi'an 710071, China  
kingdomyangfan@hotmail.com

**Abstract.** Kernel Matching Pursuit Machine (KMPM) is a relatively new learning algorithm utilizing Mercer kernels to produce non-linear version of conventional supervised and unsupervised learning algorithm. But the commonly used Mercer kernels can't expand a set of complete bases in the feature space (subspace of the square and integrable space). Hence the decision-function found by the machine can't approximate arbitrary objective function in feature space as precise as possible. Multiresolution analysis (MRA) shows promise for both nonstationary signal approximation and pattern recognition, so we combine KMPM with multiresolution analysis technique to improve the performance of the machine, and put forward a MRA shift-invariant kernel, which is a Mercer admissible kernel by theoretical analysis. An MRA kernel matching pursuit machine (MKMPM) is constructed in this paper by Shannon MRA shift-invariant kernel. It is shown that MKMPM is much more effective in the problems of regression and pattern recognition by a large number of comparable experiments.

## 1 Introduction

The KMPM uses kernel methods to map the data in input space to a high-dimensional feature space in which the problem becomes linearly separable [1,2], and many kinds of kernels can be used in KMPM as long as they are Mercer admissible kernels [1], such as the RBF and polynomial kernels. However, the commonly used Mercer kernels can't expand a set of complete bases in the feature space (subspace of the square and integrable space). Hence the decision-function found by the machine can't approximate arbitrary objective function precisely in feature space, which means the decision function learned by KMPM is only the approximation to the objective function but not its reconstruction.

The concept of multiresolution approximations of  $L^2(R)$  and the way to link it to orthonormal wavelet bases of  $L^2(R)$  was introduced by Mallat and Zhang [3,4]. According to this technique, the MRA functions can be proved to be a set of complete bases in the subspace of the square and integrable space, which shows the great flexibility and validity in signal coding and information processing. Nowadays, the application field of multiresolution analysis includes many domains such as pattern recognition, selective modifications of signals, detection and classification. Since the success of the multiresolution analysis in machine learning, it is valuable for us to study the problem of whether a better performance could be obtained if we apply multiresolution analysis to KMPM. In this paper, an MRA shift-invariant kernel,

Shannon MRA kernel, is proposed, which is an admissible Mercer kernel by theoretical analysis. Numerical experiments show the validity for improving the performance of KMPC.

## 2 MRA Mercer Kernel

It is well known that the Shannon MRA is essentially the only regular MRA having the property of shift-invariance [5]. The form of the Shannon MRA can be expressed as

$$\phi_s(t) = 2^{-m/2} \frac{\sin \pi(2^{-m}t - n)}{\pi(2^{-m}t - n)}, \quad (1)$$

where  $m, n \in \mathbb{Z}$ . When using Shannon MRA in pattern recognition, we give a minor revision to the Shannon MRA and rewrite it as

$$\phi_s(t) = \frac{\sin \pi(\frac{t}{\alpha} - b)}{\sqrt{\alpha} \pi(\frac{t}{\alpha} - b)}, \quad (2)$$

which will be get when substituting  $2^m$ ,  $n$  with  $\alpha$  (dilation factor),  $b$  (shift factor). Here,  $\alpha, b \in \mathbb{R}$ .

In order to make use of the advantage of multiresolution analysis, we propose a MRA Mercer kernel. First, one lemma must be cited.

**Lemma 1 (Mercer dot-product MRA kernel [6]).** Let  $\phi(x)$  be a MRA function, and let  $a$  and  $b$  denote the dilation and shift factors, respectively. If  $\mathbf{x}, \mathbf{x}' \in \mathbb{R}^n$ , then MRA dot-product kernel is

$$K(\mathbf{x}, \mathbf{x}') = \prod_{i=1}^n \phi\left(\frac{x_i - b_i}{a}\right) \phi\left(\frac{x'_i - b'_i}{a}\right), \quad (3)$$

From lemma 1, the Mercer MRA dot-product kernel of the MRA function  $\phi(x)$  can be constructed directly, but the dilation and shift factors must be predefined. However, when kernel method is used for the fields of machine learning, the lack of shift-invariance is especially undesirable, because the choice of the parameters in the kernel functions is quite difficult.

**Lemma 2 (Mercer shift-invariant kernel [6]).** The shift-invariant kernel  $K(\mathbf{x}, \mathbf{x}') = K(\mathbf{x} - \mathbf{x}')$  is an admissible Mercer kernel if and only if the Fourier transform of  $K(\mathbf{x})$  satisfy

$$F[K](\omega) = (2\pi)^{-n/2} \int_{\mathbb{R}^n} \exp(-i(\omega \cdot \mathbf{x})) K(\mathbf{x}) d\mathbf{x} \geq 0, \quad (4)$$

where  $\mathbf{x} \in \mathbb{R}^n$ .

From lemma 2, we only need to predefine dilation factor if we could design a MRA Mercer shift-invariant kernel. So the difficulty of the MRA kernel to the practical use will be greatly decreased.

In the following, a practical MRA shift-invariant kernel is designed by the MRA function chosen as Shannon MRA function.

**Theorem 3.** Given the MRA function (2) and the dilation factor  $\alpha \geq 0$ . If  $\mathbf{x}, \mathbf{x}' \in R^n$ , the MRA shift-invariant kernel is

$$K(\mathbf{x}, \mathbf{x}') = \prod_{i=1}^n \phi_s(x_i - x'_i) = \prod_{i=1}^n \left( \frac{\sin \pi \left( \frac{x_i - x'_i}{\alpha} \right)}{\sqrt{\alpha} \pi \left( \frac{x_i - x'_i}{\alpha} \right)} \right), \quad (5)$$

which is an admissible Mercer kernel.

Now, the regression function learned by MRA kernel is

$$f_R(\mathbf{x}) = \sum_{k=1}^N \beta_k K(\cdot, \tilde{\mathbf{x}}_k) = \prod_{k=1}^N \beta_k \left( \frac{\sin \pi \left( \frac{x_i - \tilde{x}_i}{\alpha} \right)}{\sqrt{\alpha} \pi \left( \frac{x_i - \tilde{x}_i}{\alpha} \right)} \right), \quad (6)$$

and the decision function for pattern recognition is

$$f_C(\mathbf{x}) = \text{sgn} \left( \sum_{k=1}^N \beta_k K(\cdot, \tilde{\mathbf{x}}_k) \right) = \text{sgn} \left( \prod_{k=1}^N \beta_k \left( \frac{\sin \pi \left( \frac{x_i - \tilde{x}_i}{\alpha} \right)}{\sqrt{\alpha} \pi \left( \frac{x_i - \tilde{x}_i}{\alpha} \right)} \right) \right), \quad (7)$$

where  $\tilde{\mathbf{x}}_k = (\tilde{x}_1, \dots, \tilde{x}_n)$ ,  $k = 1 \sim N$  are support points.

### 3 Validation of MKMPM's Effectiveness

In the paper, we compare the MRA kernel function with the most commonly used kernel function, RBF kernel. In each experiment of this paper, the parameters  $p$  for the RBF kernel and  $\alpha$  for the Shannon MRA kernel are selected by using cross validation that is in wide use [7]. We adopt the parameters' notation of KMPC as follows: N—maximum of the basis functions; stops—KMPC stopping criterion (pre-defined accuracy); fitN—KMPC doing a back-fitting in every fitN-step. In the test of regression, we adopt the approximation error as  $e_{ss} = \sqrt{\left( \sum_{i=1}^l (y_i - f_i)^2 \right) / l}$ , and both in the regression and classification tests, the loss function of the KMPC adopts squared loss. For avoiding the weak problem, each experiment has been performed 50

independent runs, and all experiments were carried on a Pentium IV 2.6Ghz with 512 MB RAM using Matlab6.01 compiler.

### 3.1 Regression Experiments

A well-known dataset, Boston housing dataset, choosing from the UCI machine learning database<sup>1</sup>, has been tested in this experiment.

The input space of Boston housing dataset is 13 dimensions. It contains 506 samples, and some of them are taken as the training examples while others as testing examples. Test Parameters: RBF kernel  $K(\mathbf{x}, \mathbf{y}) = \exp(-\|\mathbf{x} - \mathbf{y}\|^2 / 2p^2)$  with  $p = 3.32e+2$ , MRA kernel  $\alpha = 3.53e+2$  and parameters of KMPM  $N = 180$ ,  $stops = 0.01$ ,  $fitN = 10$ . Table 1 lists the approximation errors using the two kernels.

**Table 1.** Approximation Results of Boston Housing Dataset

# Training	# Test	Kernel	# s.p.	Error
350	156	MRA	11	8.0121
		RBF	7	8.0966
400	106	MRA	13	7.9801
		RBF	7	8.0736
450	56	MRA	12	7.2311
		RBF	7	7.8405

### 3.2 Experiments of Pattern Recognition

Waveform is one dataset of the UCI repository. It has 21 characteristic attributes with all including noise and one class attribute. It contains three classes, waveform 0, waveform 1 and waveform 3, whose number of data is 5000. When training KMPM, we choose one class as positive samples and others two classes as negative samples. The selection of the training samples is shown in table 2 and the left samples as test data. In the table 2, “Team  $i$ ” means the test on waveform  $i$  ( $i = 1 \sim 3$ ) and “+, -” represent positive and negative samples, respectively.

**Table 2.** Selection of the Training Data

Group	# Training examples					
	Team 1		Team 2		Team 3	
	+	-	+	-	+	-
1	200	400	200	400	200	400
2	250	500	250	500	250	500

<sup>1</sup> URL:<http://www.ics.uci.edu/mlearn>

We select RBF kernel function  $p = 10$  and MRA kernel  $\sigma = 10$ . KMPM Parameters:  $N = 120$ ,  $stops = 0.05$ ,  $fitN = 4$ . Table 3 lists the classification result by MRA kernel and RBF kernel, respectively.

**Table 3.** Recognition Results of Waveform Dataset Selection of the Training Recognition Results of Artificial Dataset

Group	Kernel	Recognition rates		
		Team 1	Team 2	Team 3
1	MRA	91.35%	92.01%	91.86%
	RBF	89.47%	89.83%	89.95%
2	MRAI	92.65%	92.45%	93.01%
	RBF	90.12%	89.87%	90.56%

## 4 Concluding Remarks

Kernel Matching Pursuit Machine (KMPM) is a class of learning algorithm utilizing Mercer kernels to map the input space of the problems to a high-dimensional feature space in which the problem becomes linearly separable, while allowing a better control of the sparsity of the solution. But the commonly used Mercer kernels can't expand a set of complete bases in feature space. Hence the decision-function found by the machine can't approximate arbitrary objective function in such space as precise as possible. In this paper, the multiresolution analysis technique has been combined with the kernel matching pursuit machine to improve the performance of the machine, and the Shannon MRA shift-invariant kernel has been proposed, which is also a Mercer admissible kernel by theoretical analysis. At last, the MRA kernel matching pursuit machine (MKMPM) has been constructed and the results of our simulations show the feasibility and validity of the MKMPM in regression and pattern recognition.

## References

- Engel, Y., Mannor, S., Meir, R. The kernel recursive least-squares algorithm. *IEEE Trans. Signal Processing*, vol. 52, Issue: 8, pp. 2275-2285, August 2004.
- Pascal Vincent, Yoshua Bengio. Kernel matching pursuit. *Machine Learning*, 48:165--187, 2002.
- Davis G., Mallat S., Z. Zhang. Adaptive time-frequency decompositions. *Optical Engineering* 33(7), 2183-2191.
- Mallat S. A theory for nuliresolution signal decomposition: The wavelet representation. *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 11, pp. 674-693, July 1989.
- Bastys, A. Periodic shift-invariant multiresolution analysis. *IEEE, Digital Signal Processing Workshop Proceedings*, pp.398 – 400, 1996.
- L. Zhang, W. D. Zhou, L. C. Jiao. Wavelet support vector machine. *IEEE Trans. On Systems, Man, and Cybernetics. Part B: Cybernetics*. Vol. 34, no. 1, February 2004.
- Kearns M., Ron D. Algorithmic stability and sanity-check bounds for leave-one-out cross validation. *Proc. Tenth Conf. Comput. Learning Theory*. New York: ACM, 1997, pp. 152-162.

# Multiclass Microarray Data Classification Using GA/ANN Method

Tsun-Chen Lin<sup>1</sup>, Ru-Sheng Liu<sup>1</sup>, Ya-Ting Chao<sup>2</sup>, and Shu-Yuan Chen<sup>1</sup>

<sup>1</sup> Department of Computer Science and Engineering, Yuan Ze University, Nei-Li, Chung-Li,  
Taoyuan, 32026, Taiwan, ROC  
lintsunc@ms01.dahan.edu.tw, {csrobinl, cschen}  
@saturn.yzu.edu.tw

<sup>2</sup> Graduate School of Biotechnology and Bioinformatics, Yuan Ze University, Nei-Li,  
Chung-Li, Taoyuan, 32026, Taiwan, ROC  
ytchao@saturn.yzu.edu.tw

**Abstract.** This work aims to explore the use of gene expression data in discriminating heterogeneous cancers. We introduce hybrid learning methodology that integrates genetic algorithms (GA) and artificial neural networks (ANN) to find optimal subsets of genes for tissue/cancer classification. This method was tested on two published microarray datasets: (1) NCI60 cancer cell lines and (2) the GCM dataset. Experimental results on classifying both datasets show that our GA/ANN method not only outperformed many reported prediction approaches, but also reduced the number of predictive genes needed in classification analysis.

## 1 Introduction

The analysis of gene expression profiles, which serve as molecular signatures for cancer classification, has become a challenging research topic in bioinformatics. The special characteristic of microarray classification problem is that the datasets usually contain very small samples in an extremely high dimensional gene space. In the past few years, many classification algorithms combining rank-based gene selection methods have been applied to 2-class to 4-class microarray datasets and most of them can achieve prediction accuracies close to 95-100%. However, if this problem was expanded to datasets that contain more than five classes such as the 9-class NCI60 dataset [1] and the 14-class GCM dataset [2], the performance of these methods would deteriorate significantly [3]. Instead, GA was proposed to extract genes in chromosomes to consider the correlations between features on the actual classification task itself [4-7]. Consequently, these hybrid learning methodologies have shown a better degree of accuracy than rank-based methods in multiclass microarray classification. This leads our works to ensemble above two approaches for classifying more difficult datasets in two steps. First, in order to filter irrelevant genes, the between-groups to within groups sum of squares (BSS/WSS) ratio of Dudoit et al. [8] was first performed for small data subsets. Next, we applied GA/ANN in a way that considers correlations between genes and allows non-linearly separating data in classification. Finally, GA/ANN achieved greater classification accuracy of 90.8% and 100% for

NCI60 data and GCM data respectively and needed less number of genes than reported techniques.

## 2 Methods

To reduce the dimensionality space of genes from 6128-14476 to 1000 for NCI60 data and GCM data respectively, a preliminary selection of genes with the highest BSS/WSS ratios are selected for small data subsets. Besides, we also randomly split them into 30 perturbed versions of data, and each of them is divided into the ratio of 2:1 (2/3 samples for training and 1/3 samples for testing).

For feature selection, the genetic algorithms were adopted from Ooi and Tan [4], with toolboxes of two selection methods including stochastic universal sampling (SUS) and roulette wheel selection (RWS). In addition, two tuning parameters,  $Pc$ : crossover rate and  $Pm$ : mutation rate, were used to tune one-point and uniform crossover operations to evolve the population of individuals in the mating pool. The format of chromosome is represented by the string  $S_i$ ,  $S_i = [g_1 \ g_2 \ \dots \ g_{20}]$ , where  $g_1, g_2, \dots, g_{20}$ , are the indices of 20 genes corresponding to a dataset and to be used as discriminatory genes evaluated by the fitness function of  $f(S_i) = (1 - E_t) \times 100$  for sample classification, where  $E_t$  means the prediction error rate of the training set. The parameters of GA to work with ANN in classifying data are given by the following procedure.

### Procedure GAANN

```

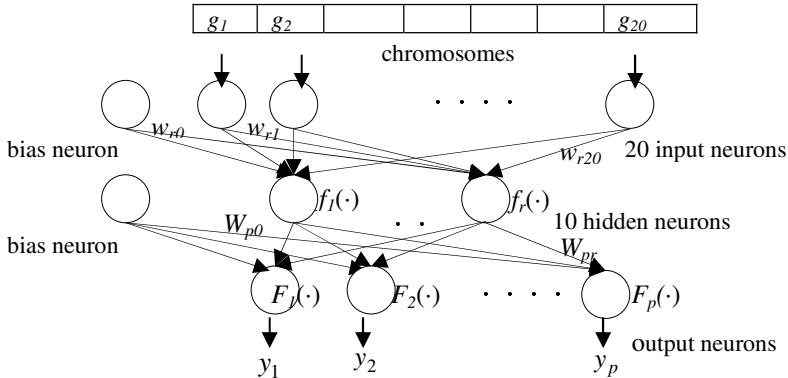
For seeds = 1 to 10 {
    generate initial population of 50 chromosomes with the size of genes equal to 20
    while (the maximum generations = 100 is not reached) {
        select a population by stochastic universal sampling operation for mating pool
        generate offspring using uniform crossover from parents using  $Pc=1$ 
        apply mutation to offspring using  $Pm=0.002$ 
        replace worst individuals with offspring
        determine the useful genes by the score of fitness = 100 }}}

```

For classification tasks, the feedforward perceptron algorithm of ANN was developed by modifying those of Nørgaard [9] to set up the neural networks. Figure 1 shows the 3-layer winner-takes-all neural networks with 20 neurons and 10 neurons for the input and the hidden layer respectively. The chromosomes consisting of 20 features will represent patterns of training samples with known class labels  $y = 1, \dots, q$  to be fed into input layer to build the classifier. The target is made depending if a sample belonging to class  $p$ , the output target for the  $p$ -th neuron is set to “1” and the other neurons elsewhere are set to “0”. Therefore, the presentation of training samples will then induce the weight matrix of the class indicator neuron to have the maximum activation. So, if we want to predict the class of an unseen query sample of class label  $p$ , the system output should be the “winning unit”, i.e., the neuron of  $p$  with,  $y_p > y_k$  for  $p \neq k$ ,  $k = 1 \dots q$ . Otherwise the sample is misclassified. The symbol  $y_p$  is defined as

$$y_p = F_p \left( \sum_{r=1}^{10} W_{pr} f_r \left( \sum_{i=1}^{20} w_{ri} g_i + w_{r0} \right) + w_{p0} \right) \quad (1)$$

where  $g_i$  are the neuron inputs and  $w_{pr}$  are the input weights for the  $p$ -th neuron.  $F(u)=\tanh(u)$  is a hyperbolic tangent function for neurons to perform a nonlinear mapping of the neuron input.

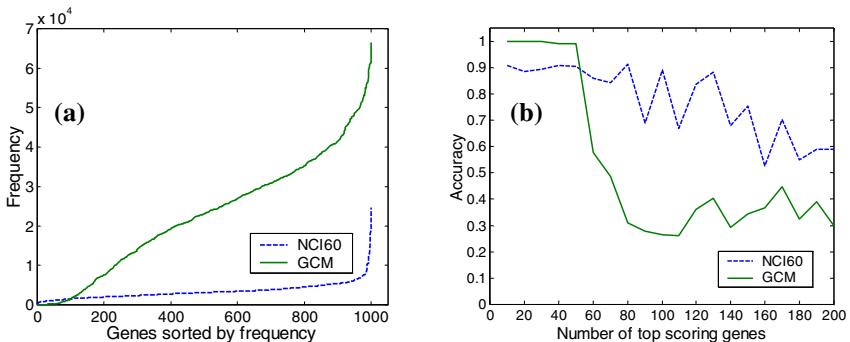


**Fig. 1.** The neural architecture runs the GA/ANN method

### 3 Results

During the 300 runs of GA/ANN procedure on 30 datasets, we obtained almost 150,000 and 1,200,000 subsets of genes that can fully separate training samples with 100% accuracy for NCI and GCM data respectively. We then examined the frequency of membership of the genes and calculated the mean  $\mu$  of frequencies, and the standard deviation  $\sigma$  of each feature in these near optimal sets. The 1000 genes were subsequently rank ordered according to the number of times each was selected as shown in Figure 2(a). We first identified 10 top ranked genes by their access numbers through GenBank, namely W03157, AA045756, AA053504, AA033882, N98804, AA047106, W25510, H18563, W73203, and W47652 (ranging from 3.56 to 10.54) for NCI data and U43944, R33301, W68502, AA024428, T50576, S77393, X52332, D11922, AA412505 and CCNE ( $\sigma$  ranging from 2.38 to 2.97) for GCM data to classify 30 versions of test samples. As the results, the novel samples of NCI data were correctly classified with average accuracy of 90.8% (range 75-100%) and 100% (range 100-100%) classification accuracy for GCM data. Furthermore, we also tested GA/ANN on the 4-class SRBCT dataset [10]. Expectably, we obtained the same results of 100% prediction accuracy as those in Deutsch [5] and Lee et al. [11] while Cho et al. [12] used the same method; they only achieved 96% classification accuracy for SRBCT data. With a choice of only a few top genes, the classification may not be reliable, when too many genes will add noise to the classification. For both datasets, we tried to vary the size of input pattern in steps of 10, first using the top 10 genes, then adding the other top 10 genes and so forth up to the top 200 genes as the best subset to classify the test samples among 30 datasets to find the average classification performance. Figure 2(b) illustrated that the GA/ANN with 10 top scoring genes were good enough for classification, and the classification accuracy dropped significantly

when more than 60 genes ( $\sigma$  ranging from 1.68 to 2.97) were used for GCM data. In addition, we also compared the classification accuracies achieved by GA/ANN with previously published methods in Table 1.



**Fig. 2.** The effect on classification accuracy of selected genes

**Table 1.** Results compared to some other reported methods

Classification Method	Accu- racy (%)	No. of genes	Classification Method	Accu- racy (%)	No. of genes
<u>NCI60 data</u>					
GA/ANN	90.8	10	GCM data	100	10
GA/MLHD [4]	80	13	GA/ANN	86	32
GA/SVM [7]	88.52	40	GA/MLHD [4]	80.99	40
GA/KNN [6]	76.23	30	GA/SVM [7]	78.26	16063
BSS/WSS/DLDA [8]	88.33	30	OVA/SVM [13]	54.34	100

## 4 Conclusion

The main issue in pattern recognition for microarray data classification is to identify useful genes for accurate classification. Our belief is that such a hybrid learning system will identify more discriminatory feature subsets than those produced by existing methods for two reasons. First, GA/ANN that embedded gene selection process in classification task itself can explore more informative genes than rank-based gene selection methods. Second, neural networks describing the nonlinear relationships between genes for discriminating samples are more powerful than linear discriminant analyses. The results not only show good classification accuracy but also need the least number of discriminatory genes, and we hope it can be useful for microarray data analysis in cancer diagnosis.

**Acknowledgments.** This work was partially supported by the National Science Council of Taiwan, R.O.C., under Grants NSC-95-2745-E-155-008-URD.

## References

1. Ross, D.T., Scherf, U., Eisen, M.B., Perou, C.M., Rees, C., Spellman, P., Iyer, V., Jef-frey, S.S., Van de Rijn,M., Waltham, M., Pergamenschikov, A., Lee, J.C., Lash-kari, D., Shalon, D., Myers, T.G., Weinstein, J.N., Botstein, D., Brown, P.O.: Systematic variation in gene expression patterns in human cancer cell lines. *Nature Genetics* 24 (2000) 227–235
2. Ramaswamy, S., Tamayo, P., Rifkin, R., Mukherjee, S., Yeang, C.H., Angelo, M., Ladd, C., Reich, M., Latulippe, E., Mesirov, J., Poggio, T. et al.: Multiclass cancer diagnosis using tumor gene expression signatures. *Proceedings of the National Academy of Sciences USA* 98 (2001) 15149–15154
3. Li, T., Zhang, C., Oghara, M.: A comparative study of feature selection and multiclass classification methods for tissue classification based on gene expression. *Bioinformatics* 20 (2004) 2429-2437
4. Ooi, C.H., Tan, P.: Genetic algorithms applied to multi-class prediction for the analysis of gene expression data. *Bioinformatics* 19 (2003) 37–44
5. Deutsch, J.M.: Evolutionary algorithms for finding optimal gene sets in microarray prediction. *Bioinformatics* 19 (2003) 45–52
6. Thanyaluk, J.U., Stuart, A.: Feature selection and classification for microarray data analysis: Evolutionary methods for identifying predictive genes. *BMC Bioinformatics* 6:148 (2005) 1-11
7. Liu, J.J., Cutler, G., Li, W., Pan, Z., Peng, S., Hoey, T., Chen, L., Ling, X.B.: Multiclass cancer classification and biomarker discovery using GA-based algorithms. *Bioinformatics* 21 (2005) 2691–2697
8. Dudoit, S., Fridlyand, J., Speed, T.P.: Comparison of discrimination methods for the classification of tumors using gene expression data. *Journal American Statistical Association* 97 (2002) 77-87
9. Nørgaard, M.: Neural Network Based System Identification Toolbox. Tech. Report. 00-E-891, Department of Automation, Technical University of Denmark. (2000)
10. Khan, J., Wei, J.S., Rigner, M., Saal, L.H., Ladani, M., Westermann, F., Berthold, F., Schwab, M., Antonescus, C.R., Peterson, C., Meltzer, P.S.: Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. *Nature Medicine* 7 (2001) 673–679
11. Lee, Y., Lee, C.K.: Classification of multiple cancer types by multicategory support vector machines using gene expression data. *Bioinformatics* 19 (2003) 1132–1139
12. Cho, H.S., Kim, T.S., Wee, J.W., Jeon, S.M., Lee, C.H.: cDNA Microarray Data Based Classification of Cancers using Neural Networks and Genetic Algorithms. *Nanotech* 1 (2003) 28-31
13. Yeang, C.H., Ramaswamy, S., Tamayo, P., Mukherjee, S., Rifkin, R.M., Angelo, M., Reich, M., Lander, E., Mesirov, J., Golub, T.: Molecular classification of multiple tumor types. *Bioinformatics* 1 (2001) 1–7

# Texture Classification Using Finite Ridgelet Transform and Support Vector Machines\*

Yunxia Liu, Yuhua Peng, and Xinhong Zhou

School of Information Science and Engineering, Shandong University

Jinan, Shandong, China 250100

liuyunxia1983@mail.sdu.edu.cn, pyuhua@sdu.edu.cn,  
zhouxinhong@mail.sdu.edu.cn

**Abstract.** Based on energy distribution analysis of FRIT coefficients, a novel feature extraction method of low computation complexity in FRIT domain was proposed for texture classification in this paper. A ‘one-against-one’ multi-class SVM with RBF kernel was adopted as classifier. Experiments carried out on abundant texture databases with varying sizes demonstrated its validity.

## 1 Introduction

In recent decades, texture classification has received considerable attention in various application areas ranging from industrial automation to medical diagnosis etc. Texture analysis methods can be classified into four primary categories, namely statistical, geometrical, model-based and signal processing-based approaches<sup>[1]</sup>.

Lots of wavelet-based methods were proposed and have achieved good results due to their multi-resolution characteristics. However, research on Human Visual System (HVS) reveals that it relies on edges, contours more than pixels in information acquisition. Thus 2-D separable wavelet loses its superiority. New transform schemes with better geometrical presentation ability are expected to arise.

Based on the statistical study of energy distribution in FRIT domain, a transform domain subband division scheme is obtained. Subband statistics are then extracted as features for classification. Compared with former feature extraction work done by Shutao Li<sup>[2]</sup> and Arivazhagan<sup>[3]</sup> in Ridgelet domain, computation complexity of our method is greatly reduced. Besides, FRIT has a much smaller coefficient matrix, thus leads to the possibility of more precise subband division in transform domain.

Support vector machine (SVM) is adopted as classifier in this paper. It outperforms the traditional risk-based approaches and has demonstrated excellent performance in a variety of pattern recognition problems in various practical problems<sup>[4]</sup>. The Radial Basis Function (RBF) is used as kernel function, and the ‘one against one’ strategy is adopted for multi-class classification problem.

---

\* This project is sponsored by SRF for ROCS, SEM (2004.176.4) and NSF SD Province (Z2004G01) of China.

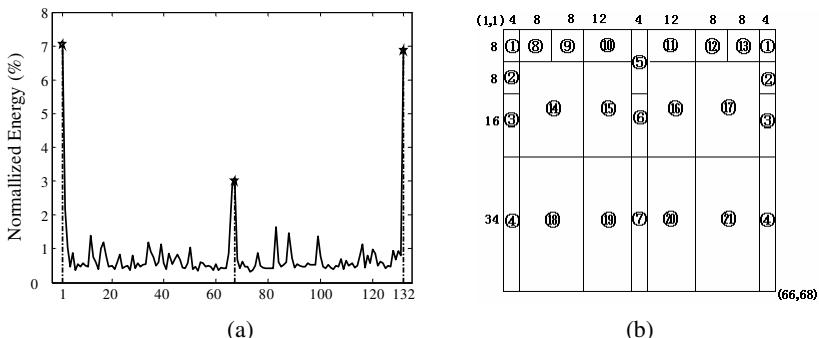
## 2 Feature Extraction Using Finite Ridgelet Transform

FRIT is a multiscale orthogonal transform pioneered by Do and Vetterli for better representation of linear singularities [5]. After the Finite Radon Transform (FRAT) mapping linear singularities in image domain to point singularities in FRAT domain, a collection of orthogonal transforms are applied to FRAT coefficients columnwisely to accomplish the FRIT. The resulting transform is invertible, non-redundant and leads to a family of directional orthogonal bases for digital images.

On computing FRIT of an image with dyadic size  $n \times n$ , one first expands the image to a prime size of  $p \times p$ , where  $p$  is the minimum prime number larger than  $n$ . A fast implementation of the FRAT described in [6] involves  $p^2$  multiplications only, and coefficient matrix is  $p \times (p+1)$ . The DRT adopted in [3] has a coefficients matrix twice size of the input image. Computation complexity of the iterative algorithm adopted in [7] is  $O(n^2 \log(n^2))$ , and the coefficient matrix is four times over-complete. Obviously, as the inequality  $O(p^2) \approx O(n^2) < O(n^2 \log n^2)$  holds true for all natural number, we conclude that FRIT has advantages over existing methods in terms of both computation complexity and coefficient over-completeness, which will benefit its application in feature extraction.

We investigate statistical properties of FRIT for subband division. Firstly, energy distribution among FRIT columns is examined. Fig.1 (a) displays typical normalized energy distribution waveform obtained by D20 in Brodaze. In fact, it holds true for almost each texture image that energy tends to distribute more in the middle ( $45^\circ$ ) and the both ends ( $0^\circ$  and  $90^\circ$ ) columns of FRIT, while energy of other columns is relative small and distributed randomly. Definition of FRAT lines may help to explain this that  $0^\circ$ ,  $45^\circ$  and  $90^\circ$  FRAT lines being identical to natural lines are more likely to present peaks. Secondly, a three-level decomposition of DWT (which is assumed as the default setup in the sequel) is applied to all FRAT projections for a further lookup of energy distribution within FRIT columns. Our experiment reveals that approximation coefficients of lower frequency concentrate most of the energy within each FRIT columns, which is consistent with wavelet theory.

Instructed by the idea that more energy distributed areas possess more information and should be covered with finer subbands, we propose an FRIT-based subband



**Fig. 1.** (a) Normalized energy distribution ( $n = 128$ ). (b) subband division ( $n = 64$ ).

division method, see Fig.1 (b). Occasions for  $32 \times 32$  and  $128 \times 128$  images are similar, and are not given for limitation of space.

### 3 Support Vector Machine for Classification

Assume the training set  $\{(x_i, y_i), i=1, 2, \dots, l\}$  is denoted by  $D$ , where each input training vector  $x_i \in R^n$  coming from two classes and the output label  $y_i \in \{+1, -1\}$ . Firstly, training vectors  $x$  are mapped to a Hilbert space  $F$  via a nonlinear map  $\phi: R^n \rightarrow F$ . Then SVM construct a classifier with maximized separation gaps between the positive and negative examples by solving the following primal problem:

$$\min_{w, b, \varepsilon} \frac{1}{2} w^T w + C \sum_{i=1}^l \varepsilon_i \quad (1)$$

with the constriction conditions:  $y_i(w^T \phi(x_i) + b) \geq 1 - \varepsilon_i, \varepsilon_i \geq 0, i=1, \dots, l$ .

For extention of binary SVM to multi-class classifiers, Hsu [8] compared the performance of three commonly used strategies and concluded that ‘one-against-one’ and DAGSVM methods outperforms others for practical use. The ‘one-against-one’ voting strategy is adopted in our method, in which  $C_i^2 = l(l-1)/2$  classifiers are constructed and each takes charge of classifying data from two different classes.

### 4 Method and Procedures

We adopt the database in [2] as the first database. Sixteen natural textures (i.e. Brick.0000, Fabric.0007, 0008, 0013, 0017, 0018, Flowers.0005, 0006, 0007, Grass.0001, Metal.0000, 0002, 0004, Misc.0001, Sand.0000, 0002) of size  $512 \times 512$  from the VisTex are selected as the second database. It’s a challenging work as some of them are very similar to each other. Complete separate training and testing sets are required in our experiment. Fig.2 depicts the classification steps.

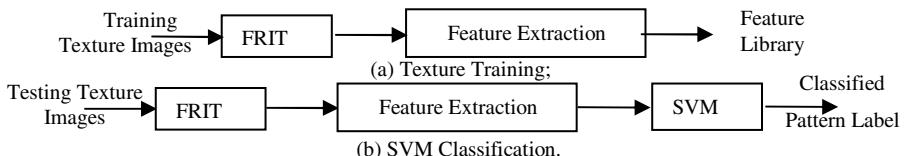


Fig. 2. Texture classification datagram

#### 4.1 Feature Extraction and Performance Evaluation

Each image is divided into non-overlapping subimages ( $64 \times 64$ ), thus 1000 and 1024 subimages are resulted for the two databases. A parameter  $ratio \in (0, 1)$  is introduced to control the size of training and testing datasets. Local energies

$$\frac{1}{M \times N} \sum_{x=1}^M \sum_{y=1}^N I_i^2(x, y) \quad (2)$$

from each FRIT subband (indexed by  $i, i=1, 2, \dots, 21$ ) are then used as texture features, where  $M$  and  $N$  denote the subband size. Average of each subimage subtracted in the first step of FRAT is also combined to the feature vector as well. Thus a 22-dimensional feature vector is extracted for images with size of  $64 \times 64$ .

As suggested by previous work in [2], DRT combining with DWT helps for classification performance improvement, DWT features are also involved in our method here. Mean and standard deviation of detail subbands of DWT ( $LH_k, HL_k, HH_k$ , for  $k=1, 2, 3$ ) are calculated according to the following formula [9]:

$$Mean(m) = \frac{1}{N^2} \sum_{i,j=1}^N |d(i, j)|; Std = \sqrt{\frac{1}{N^2} \sum_{i,j=1}^N (d(i, j) - m)^2} \quad (3)$$

Performance is evaluated by means of computing classification gain:

$$G(\%) = \frac{C_{corr}}{M} \times 100\%, \quad (4)$$

where  $C_{corr}$  is the number of images correctly classified and  $M$  is the total number of images belonged to that particular class.

## 4.2 Experiment Results

A RBF kernel function is used here as preliminary results suggest that it performs other kernel for our texture images. The garma value is set to be 0.04 and C=5, which can all be adjusted to achieve better performance.

Classification results carried on the first database are given in Tabel.1, together with results in [2]. Obvious improvement of classification gain can be observed, while the FRIT-based methods are of lower computation complexity.

**Table 1.** Comparison of different feature extraction methods in terms of classification gain

Ratio	DWT	DRT	COM	FRIT	FRIT&DWT
12.5%	79.8%	75.9%	84%	86.3%	93.6%
25%	84.9%	84.9%	91.4%	91.2%	96.2%
50%	90.3%	90.1%	94.6%	91%	98.2%
75%	92.4%	91.3%	95.7%	92%	98.4%

Experiment on the second database employs a subtler adjusting of  $ratio$ . As shown in Fig.3 (a), FRIT-based classification method outperforms the DWT-based one by an average of 3.16 percents, while the combined method of FRIT and DWT possesses the best performance. As training data is always small in practical applications, a further lookup of (a) is given in (b). The combined method achieves an average 1.42 percents classification gain advantage over [2]. A comprise between this two methods can be chosen according to practical application requirements.

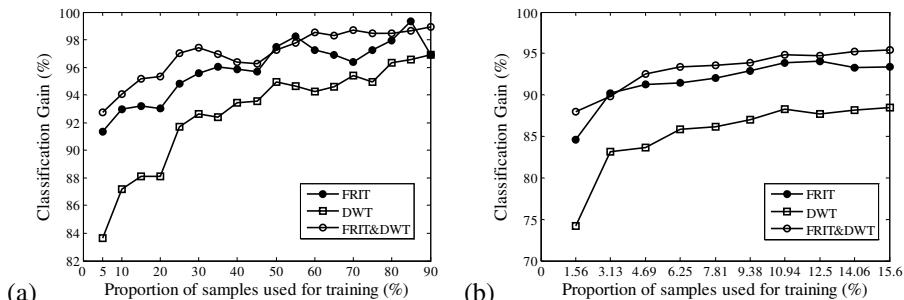


Fig. 3. Classification Gain comparison on database two

## 5 Conclusions

An FRIT-based feature extraction method with low computation complexity for texture classification is proposed in this paper. A multi-class SVM constructed following the ‘one-against-one’ strategy is adopted as classifier, using RBF as its kernel. Experiments carried out on abundant databases demonstrated its validity. Both classification gain improvement and satisfying results are obtained.

## References

1. Tuceyran M. and Jain A. K., Texture analysis, in Handbook of Pattern Recognition and Computer Vision, World Scientific Publishing Co., Chapter 2.1, (1998) 207-248
2. Shutao Li, Yi Li, Yaonan Wang, Combining Wavelet and Ridgelet Transforms for Texture Classifications using Support Vector Machines, Proceedings of International Symposium on Intelligent Multimedia, Video and Speech Processing, October (2004)442-445
3. Arivazhagan, S.; Ganesan, L.; Subash Kumar, T.G.; Texture classification using Ridgelet transform. 6<sup>th</sup> ICCIMA, 16-18 Aug. (2005) 321 – 326
4. C.J.C. Burges, A tutorial on support vector machines for pattern recognition, Data Mining Knowledge Discovery 2 (2) (1998) 121–167
5. Do, M.N, Vetterli, M, The finite ridgelet transform for image representation, IEEE Transactions on Image Processing, Volume 12, Issue 1, Jan. (2003) 16 - 28
6. Matus, F.; Flusser, J. Image representation via a finite Radon transform. IEEE Transactions on PAMI, Volume 15, Issue 10, Oct. (1993) 996 – 1006
7. David L. Donoho, Ana Georgina Flesia, Digital Ridgelet Transform Based on True Ridge Functions, Technical Report 31 (2001)
8. C.-W. Hsu and C.-J. Lin. A comparison of methods for multi-class support vector machines. IEEE Transactions on Neural Networks, 13(2) (2002) 415–425
9. DeBrunner, V.; Kadiyala, M. Texture classification using wavelet transform Circuits and Systems, 1999. 42nd Midwest Symposium , 8-11 Aug. (1999 ), vol. 2, 1053 - 1056

# Reduction of the Multivariate Input Dimension Using Principal Component Analysis

Jianhui Xi<sup>1</sup> and Min Han<sup>2</sup>

<sup>1</sup> Department of Automation, Shenyang Institute of Aeronautical Engineering,  
Liaoning, China, 110034  
xjhui\_01@163.com

<sup>2</sup> School of Electronic and Information Engineering, Dalian University of Technology,  
Liaoning, China, 116023

**Abstract.** There are limitations for the existing methods to model multivariate time series because that defining the input components is highly difficult. The main purpose of this paper is to expand the principal components analysis (PCA) method to extract the joint information of multiple variables. First, both the linear correlations and the nonlinear correlations are detected to initialize an embedding delay window, which contains enough information for prediction. Then, the PCA method is expanded to extract the joint information of multiple variables in a complex system. Finally, neural network makes predictions on the basis of approximating both the functional relationship between different variables and the map between current state and future state.

## 1 Introduction

Most of previously published prediction methods are concentrated on the modeling of univariate time series. The main idea involves two steps: reconstruct a phase space from data [1], and find a functional relationship between current state and future state [2]. In the first step, the embedding dimension  $m$  and the time delay  $\tau$ , are computed so that in the  $m$  dimensional space, such time-lagged vectors topologically describe over time an object that is equivalent to the conjectured attractor of the physical system. However, for a practical complex system, the internal dynamics is often contained in multiple model variables. When a vectorial time series is available, it is possible to exploit the joint information to obtain a better reconstruction, in turn, also produces a predictive improvement. In fact, once the embedding vectors have been built in reason, the prediction procedure is exactly the same as in the univariate case. But the joint information is highly difficult to be extracted from different time series.

Cao et al [3] select  $m$  and  $\tau$  from a great deal of input vector sets according to the prediction precision, which needs big computation. Some previous research [4] suggests that it may be more appropriate to fix the reconstruction window,  $T = \tau(m - 1)$ , rather  $\tau$  alone. Considering that for the case of time series prediction, recent delays are more important than older delays,  $\tau$  is usually small. Consequently,  $m$  is usually large, especially for multivariate prediction, which may cause the model to have an excess number of inputs that highly correlated. This problem can be solved by reducing

dimension of the delay vector with Principal component analysis (PCA) method. Liang et al [5] proves the optimal performance of the proper orthogonal bases of the singular value decomposition (SVD). Therefore, PCA realized by SVD extracts the principal components from stochastic process.

In this paper, to include enough prediction information in the input vector, a linear function and a nonlinear function are respectively used to detect the linear and the nonlinear correlations in the data [1]. An initial embedding delay window is defined. Then, use PCA to reconstruct the input vectors and use neural networks to model the relationship between inputs and outputs. In the following sections, section 2 describes the methodology and basic model structure. In section 3, simulation using the proposed method is described. Finally, section 4 gives a summary and discussion.

## 2 Basic Modal Structure

More than one time series can be observed in the modeling of complex system. Figure 1 shows a schematic representation of the basic prediction model with  $n$  variables, where  $x_1(t), \dots, x_n(t)$ ,  $t = 1, \dots, N$ , are the observed time series.  $y_1(t), \dots, y_n(t)$  are the output of the prediction model. The error are denoted by  $e_1(t), \dots, e_n(t)$ .  $m_1, \dots, m_n$  are the embedding dimension corresponding to each variable. Taking a system including two variables as an example, the initial input vector is built like

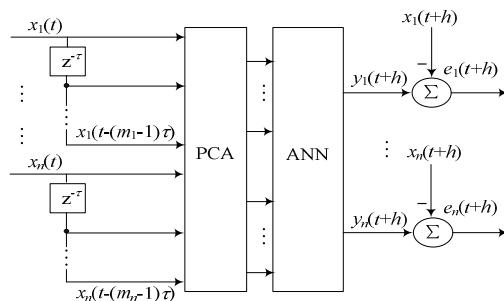
$$\begin{aligned}\vec{x}(t) &= \{x_1(t), \dots, x_1(t-(m_1-1)\tau), x_2(t), \dots, x_2(t-(m_2-1)\tau)\}^T \\ &= \{\vec{x}_1(t)^T, \vec{x}_2(t)^T\}^T\end{aligned}\quad (1)$$

First, in order to retain enough prediction information for each variable, a delay window  $T_{im} = \tau_i (m_i - 1)$  ( $i = 1, 2$ ) are often first fixed. Here a linear function and a nonlinear function are respectively used as

$$\Phi_{xx}(T_{im}) = E\{[x_i(t) - \bar{x}_i(t)][x_i(t-T_{im}) - \bar{x}_i(t)]\} \quad (2)$$

$$\Phi_{x^2x^2}(T_{im}) = E\{[x_i^2(t) - \bar{x}_i^2(t)][x_i^2(t-T_{im}) - \bar{x}_i^2(t)]\} \quad (3)$$

where  $\Phi_{xx}(T_{im})$  could only detect the linear correlations,  $\Phi_{x^2x^2}(T_{im})$  is designed to



**Fig. 1.** Structure of basic prediction model

detect nonlinear correlations in the data.  $E\{\cdot\}$  is the mathematical expectation and the overbars indicate averaging with respect to time. Assuming that  $T_{im}^x$  is the time of the first maximum of  $\Phi_{xx}(T_{im})$  and  $T_{im}^{x^2}$  is defined analogously for  $\Phi_{x^2x^2}(T_{im})$ , choose

$$T_{im} = \max(T_{im}^x, T_{im}^{x^2}) - 1 \quad (4)$$

The data in  $T_{im}$  could contain the linear and nonlinear correlations of the series.

Second, PCA is separately implemented for  $\bar{x}_1(t)$  and  $\bar{x}_2(t)$ . Because a suitable  $\tau_i$  for all model variables is difficultly obtained, we often choose  $\tau_i=1$ , which require the lowest functional complexity. If  $\bar{z}_1(t)$  is the principal components of  $\bar{x}_1(t)$ , and  $\bar{z}_2(t)$  corresponds to the  $\bar{x}_2(t)$ , the principal components  $\bar{z}(t)$  of input vector is

$$\bar{z}(t) = \left[ \bar{z}_1(t)^T, \bar{z}_2(t)^T \right]^T \quad (5)$$

$$\bar{z}_i(t) \approx \bar{x}_i(t) \tilde{\mathbf{V}} \quad (6)$$

where  $\tilde{\mathbf{V}}$  ( $m_i \times r$ ) is the first  $r$  columns of  $\mathbf{V}$  which is obtained from

$$\mathbf{U} \Sigma \mathbf{V}^T = \mathbf{X}_i \quad (7)$$

$$\Sigma = \text{diag}[s_1 \ s_2 \ \cdots \ s_p \ 0 \ \cdots \ 0] \quad (8)$$

$\mathbf{X}_i$  ( $l \times m_i$ ) contains the original  $\bar{x}_i(t)$ ,  $l = N - \max_i [(m_i - 1)\tau_i]$ .

Estimate the energy contribution  $\eta_j$  of a principal component as

$$\eta_j = \frac{s_j^2}{\sum_{i=1}^p s_i^2}, \quad j = 1, 2, \dots, p \quad (9)$$

Retain the corresponding  $r$  principal components if  $\sum_{j=1}^r \eta_j > \eta_0$ ,  $0.5 < \eta_0 < 1$ .

Finally, neural network are used to search the functional relationship between the current state  $\bar{x}(t)$  and the future state after time  $h$ ,  $\bar{x}(t+h)$ . In this paper, a four-layer feedforward neural network is adopted [3]. The activation functions of hidden layers are sigmoid functions and linear functions are for the nodes of output layer. The popular gradient descent algorithm is used to train the weights of the network.

### 3 Simulation

The monthly rainfall and temperature time series of Dalian, China, from 1905 to 1940, totaling 432 records, are modeled. According to Equation (2) and (3), choose an embedding priori by fixing the delay time  $\tau=1$  and  $m_1 = m_2 = 12$ , namely,

$$\tilde{\zeta}(t) = \{x(t), \dots, x(t-11), y(t), \dots, y(t-11)\}^T$$

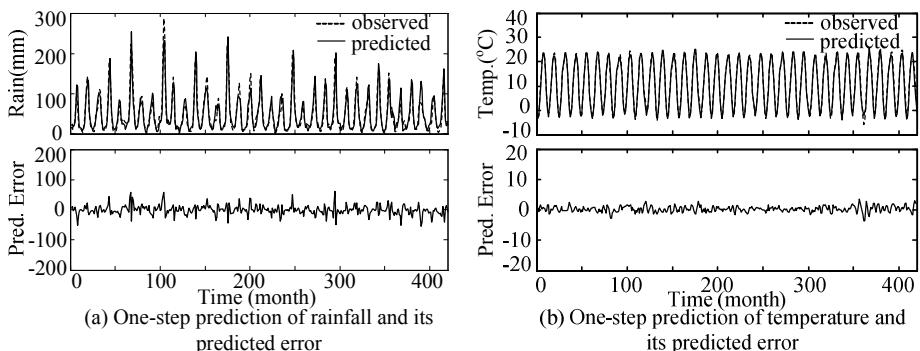
where  $x(t)$  represents the rainfall and  $y(t)$  represents the temperature. Consequently, 420 input-output samples are generated. Use the first 300 samples to train the neural network and another 120 samples to test the neural network.

Firstly, use PCA method to extract the important uncorrelated information.  $\eta_0 = 0.96$ , the eigenvalues and the  $\eta_j$  computed as Equation (9) are shown in Table 1. It is found that the sum of  $\eta_j$  of the first 9 components of rainfall series is greater than  $\eta_0$ , as well as 5 components of temperature series do. Therefore, the dimension of input vector can be reduced to  $m = 14$ . Then, the current state is presented to the neural network. Figure 2 shows the predicted one-step rainfall series and temperature series. It can be seen that the neural network gives a very good prediction results.

In order to quantitatively measure the performance of proposed prediction model, the simulation results are compared with those obtained from univariate model and document [3]. Prediction accuracy ( $E_{PA}$ ) and root-mean-square error ( $E_{RMSE}$ ) are used as indicators [6]. The univariate model only uses the univariate information. Document [3] uses the mutual information method to define  $m_i$  and  $\tau_i$  of each variable. The neural network predictors of all the three models are the same. Table 2 is the comparison of one-step prediction and five-step prediction between three models. When the

**Table 1.** The eigenvalues and their energy contribution of rainfall and temperature series

$j$		1	2	3	4	5	6
$x(t)$	$s_j$	8.2452	8.0538	4.1887	3.7475	3.3335	2.0734
	$\eta_j$	0.2452	0.2395	0.1246	0.1114	0.0991	0.0617
$y(t)$	$s_j$	30.8307	17.6706	17.6629	1.3168	1.2191	0.5734
	$\eta_j$	0.4386	0.2514	0.2512	0.0187	0.0173	0.0082
$j$		7	8	9	10	11	12
$x(t)$	$s_j$	1.4497	1.0119	0.6153	0.4352	0.2834	0.1893
	$\eta_j$	0.0431	0.0301	0.0183	0.0129	0.0084	0.0056
$y(t)$	$s_j$	0.3836	0.2544	0.1650	0.1133	0.0693	0.0413
	$\eta_j$	0.0055	0.0036	0.0023	0.0016	0.0010	0.0006



**Fig. 2.** One-step prediction comparison of the predicted and observed curves of rainfall and temperature series and their error

prediction lead times increase, the  $E_{RMSE}$  and  $E_{PA}$  obtained by multivariate prediction are generally closer to the ideal value than those obtained by univariate prediction. The joint information of  $x(t)$  and  $y(t)$  series produces a predictive improvement. The proposed method has better model identification ability and the computation is simple.

**Table 2.** Prediction model performance comparison of rainfall and temperature of Dalian

		$x(t)$		$y(t)$	
		$E_{RMSE}$	$E_{PA}$	$E_{RMSE}$	$E_{PA}$
One-step prediction	PCA+NN	15.5745	0.9457	1.0069	0.9946
	Document [3]	19.0334	0.9288	1.4797	0.9901
	Univariate	17.4778	0.9297	0.8948	0.9954
Five-step prediction	PCA+NN	26.8938	0.8258	1.1283	0.9929
	Document [3]	28.4662	0.8227	1.6162	0.9849
	Univariate	31.2086	0.7533	0.9543	0.9948

## 4 Conclusion

In this paper, PCA method is used to extract the uncorrelated information and search the principal components from the rudely reconstructing vectors. Then the good approximation ability of neural network is used to model the multivariate complex system. When built the initial embedding delay window, synchronously considering the linear correlations and the nonlinear correlations could assure the enough prediction information included in the input space.

## Acknowledgement

This research is supported by the project (60374064) of the National Natural Science Foundation of China. It is also supported by the Doctor Foundation (05YB22) of the Shenyang Institute of Aeronautical Engineering. All the support is appreciated.

## References

1. Aguirre L. A.: A nonlinear correlation function for selecting the delay time in dynamical reconstructions. Physics Letters A, 203 (1995) 88-94.
2. Bakker R., Schouten J. C., Giles C. L., et al.: Learning chaotic attractors by neural networks. Neural Computation, 12 (2000) 2355-2383.
3. Cao L., Mees A., Judd K.: Dynamics from multivariate time series. Physica D, 121(1998) 75-88
4. Rosenstein M. T., Collins J. J., Luca C. J. D.: Reconstruction expansion as a geometry-based framework for choosing proper delay time. Physica D, 73 (1994) 82-98.
5. Liang Y. C., Lee H. P., Lim S. P., Lin W. Z., Lee K. H., Wu C. G.: Proper orthogonal decomposition and its application---part I: theory. Journal of Sound and Vibration, 252 (2002) 527-544
6. Chen J. L., Islam S., Biswas P.: Nonlinear dynamics of hourly ozone concentrations: nonparametric short term prediction. Atmospheric environment, 32 (1998) 1839-1848

# Designing Prolog Semantics for a Class of Observables\*

Lingzhong Zhao<sup>1,2</sup>, Tianlong Gu<sup>2</sup>, Junyan Qian<sup>2</sup>, and Guoyong Cai<sup>2</sup>

<sup>1</sup> Electronic Engineering School, Xidian University, Xi'an, China 710071

<sup>2</sup> Department of Computer Science, Guilin University of Electronic Technology,

Guilin, China 541004

gl1zzhao@guiet.edu.cn

**Abstract.** Prolog is a well-known logic programming language. The information on the correct partial answers (cpa) and correct call patterns (ccp) of goals is useful for static analysis of Prolog programs. Decorated tree (DT) semantics is a goal-independent denotational semantics for Prolog that has been shown to be promising in Prolog program analysis. We extend the work in [7] and propose a two-step method for achieving cpa or ccp semantics from DT semantics. This paper is mainly concerned with the design of semantic domains.

## 1 Introduction

Static program analysis is the process of determining conservative approximations of the runtime properties of programs before their executions. Static analysis of Prolog programs is essential in optimizing Prolog compilers and Prolog programs [3]. (Correct) partial answers and (correct) call patterns are useful observables for analysis of Prolog programs [2,5].

Partial answers (pa) of a goal are the answers computed at any intermediate step in the SLD-derivation of the goal. Correct partial answers are those partial answers that belong to successful derivations. Given a program and a goal  $G$  whose input substitution is  $\rho$ ,  $\rho$  can be regarded as a partial answer of  $G$ ; if  $\rho$  is not correct  $G$  will not lead to any solution. Given the following program  $\{nat(X):-\; nat(X-1).\; nat(0).\}$ , the correct partial answers analysis of the program will show that a substitution is a correct partial answer of goal  $nat(X)$  only when  $X$  is not a free variable. Using this information we conclude that goal  $nat(5)$  may have a solution, but it's impossible for goal  $nat(Y)$  with variable  $Y$  being free to have any solution because empty substitution is not a correct partial answer of this goal.

Call patterns (cp) are the atoms that are selected during the SLD-derivation of a goal. Correct call patterns are those call patterns that belong to successful derivations. Information on the call patterns and correct call patterns of a goal is useful for the optimization of Prolog programs [2,4].

The development of appropriate cpa (ccp) semantics is the base for semantics-based correct partial answers (correct call patterns) analysis of Prolog programs.

---

\* This work is supported by National Natural Science Foundation of China (No.60563005) and Guangxi Natural Science Foundation of China (No.0542036).

Several pa and cp semantics for logic programs have been proposed in literature [2,4,5,6,8], but works on cpa and ccp semantics is relatively few. In the case of Prolog language, it's often difficult to define a cpa or ccp semantics for program analysis when considering control rules of Prolog and cut operator.

Decorated tree (DT) semantics is a goal-independent denotational semantics for Prolog [7] that has been shown to be promising for the goal-independent computed answers and call patterns analysis of Prolog programs [8]. We extend the work of [7] by showing that cpa or ccp semantics can be achieved by first abstracting DT semantics to an intermediate modified decorated tree (MDT) semantics and then achieving cpa or ccp semantics as an abstraction of MDT semantics.

This paper is mainly concerned with the design of semantic domains and the details of the resultant semantics will be described in other papers.

## 2 Preliminaries

We assume the familiarity with the basic algebraic structures and Prolog language. The basic notations are used in the usual way. A sequence is an ordered collection of elements possibly with repetitions. The set of all possible sequences of elements of E is denoted by  $\text{Seq}(E)$ .  $:::$  represents sequence concatenation. A sequence is denoted by a variable with a tilde sign on it. The empty sequence is explicitly written as  $\varepsilon$ .

An abstract syntax was used for Prolog programs in [7] to simplify the design of semantic operators. The basic idea is to look at Prolog as an instance of the general constraint logic programming (CLP) scheme. Since an atom  $p(x, y)$  can be represented as  $\exists z(z = (x, y) \wedge p(z))$ , without loss of generality all the predicates in Prolog programs are assumed to be unary. The clause has the form  $p(x):- G_1 \text{ or...or } G_n$  with  $n \geq 1$ , where  $G_1, \dots, G_n$  are goals defined by the grammar:

$$G ::= c \mid p(x) \mid G \text{ and } G \mid \text{exists } x. G \mid \text{cut}(G) \mid \text{cut}_d(G),$$

where  $c$  is a constraint,  $x$  is a program variable,  $d$  in construct  $\text{cut}_d(G)$  is a non-negative integer which represents the effect scope of a cut operator. The cut operator in  $\text{cut}(G)$  is assumed to have an infinite effect scope and is called open cut, and that in  $\text{cut}_d(G)$  is called closed cut. The set of all possible goals is denoted by  $\mathbb{G}$ . The expression  $p(x)$ , where  $p$  is a predicate symbol, is called *procedure call*. A goal is called *divergent* if and only if it contains a procedure call. The notation  $\text{div}(G)$  means  $G$  is divergent. A goal is *convergent* if it's not divergent. The constraints domain from which  $c$  is taken is defined as a lattice  $\langle \mathcal{B}, \leqslant, \vee, \wedge, \text{true}, \text{false} \rangle$ .

Next we recall the main concepts of the DT semantics defined in [7]. In DT semantics, the program denotation associates to any predicate in a Prolog program a decorated tree that describes all the possible SLD-derivations from the predicate. An observability constraint in a node of a decorated tree tells how the node is affected by divergent computation or by cut operators, and therefore it can be used to determine whether a Prolog interpreter can actually visit the node according to the Prolog control rules. This is the main idea of the technique of “control compilation” [1,7].

The concept of tree is basic in DT semantics. A tree is an element of the set  $\mathbb{T} = \{(G, \varepsilon) \mid G \in \mathbb{G}\} \cup \{(G, \tilde{t}) \mid G \in \mathbb{G}, \text{div}(G) \text{ and } \tilde{t} \in \text{Seq}(\mathbb{T})\}$ .

A decorated tree is achieved by adding an observability constraint to every node of a tree. The set  $\mathcal{O}$  of observability constraints is formalized in definition 2.1.

**Definition 2.1.** The set  $\mathcal{O}$  of observability constraints is defined as the minimal set containing basic constraints set  $\mathcal{B}$  and such that if  $S \subseteq \mathcal{O}$  then  $\sqcap S$  and  $\sqcup S$  belong to  $\mathcal{O}$  and such that if  $o \in \mathcal{O}$  then  $\neg o \in \mathcal{O}$ . We will often write  $o_1 \sqcap \dots \sqcap o_n$  for  $\sqcap\{o_1, \dots, o_n\}$  and  $o_1 \sqcup \dots \sqcup o_n$  for  $\sqcup\{o_1, \dots, o_n\}$ . Moreover  $true_{\mathcal{O}}$  and  $false_{\mathcal{O}}$  are shorthand for  $\sqcap\{\}$  and  $\sqcup\{\}$ , respectively. An observability constraint  $o$  is true if and only if  $o \in \mathcal{B}$  and  $o \neq false$ , or  $o = \sqcap S$  and every  $o \in S$  is true, or  $o = \sqcup S$  and there exists  $o \in S$  which is true, or  $o = \neg o'$  and  $o'$  is not true. A constraint  $o$  is false when it is not true. Next we use a unary operator  $\diamond obs$  to convert a basic constraint in  $\mathcal{B}$  to an observability constraint, and also write  $b$  for  $b \diamond obs$  when it does not make confusions.

The set  $\mathbb{DT}$  of decorated trees is defined as

$$\mathbb{DT} = \{(o + G, \varepsilon) \mid o \in \mathcal{O}, G \in \mathbb{G}\} \cup \{(o + G, \tilde{t}) \mid o \in \mathcal{O}, G \in \mathbb{G}, \text{div}(G) \text{ and } \tilde{t} \in \text{Seq}(\mathbb{DT})\}.$$

Given a decorated tree, it is assumed that the root has height 0 and its children have height 1, and so on. A Prolog interpreter can visit a node in a sequence  $\tilde{t}$  of decorated trees only if the observability constraint contained in the node is true. A node represents a solution if it is observable, convergent and consistent.

**Example.** The tree  $T$  in Fig.1 (a) can be translated into the decorated tree in Fig.1 (b).

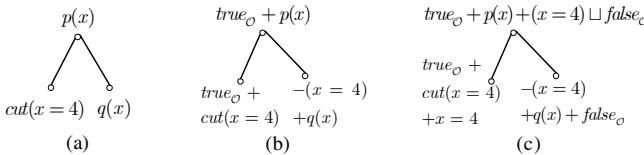


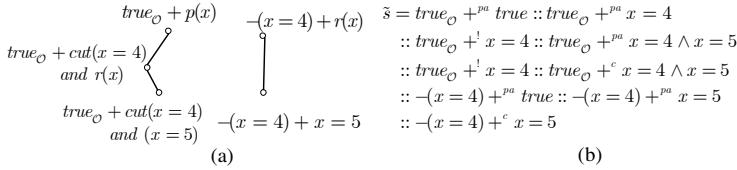
Fig. 1. (a) A tree  $T$  and its corresponding (b) decorated tree and (c) MDT

### 3 Problems in Achieving cpa and ccp Semantics

It's not difficult for us to design a pa semantics as an abstraction of DT semantics using approach similar to that for achieving call pattern semantics [7]. What we need is a new semantic domain that is capable of describing partial answers. We define  $\mathbb{PA} = C^c \cup C^d \cup C^l \cup C^{pa}$ , where  $C^c = \{o +^c b \mid o \in \mathcal{O} \text{ and } b \in \mathcal{B}\}$  (convergent constraints),  $C^d = \{o +^d b \mid o \in \mathcal{O} \text{ and } b \in \mathcal{B}\}$  (divergent constraints),  $C^l = \{o +^l b \mid o \in \mathcal{O} \text{ and } b \in \mathcal{B}\}$  (cut constraints), and  $C^{pa} = \{o +^{pa} b \mid o \in \mathcal{O} \text{ and } b \in \mathcal{B}\}$  (partial answer constraints). Then the set  $\text{Seq}(\mathbb{PA})$  will be the semantic domain for our partial answer semantics.

Every decorated tree will be abstracted into an element of  $\text{Seq}(\mathbb{PA})$ . The abstraction of a node  $(o + G)$  will give rise to convergent constraints (representing a possible solution) if  $G$  does not contain procedure call, to divergent constraints if  $G$  contains procedure call, and to cut constraints (representing conditions for the execution of a cut) if  $G$  contains open cuts. Moreover the abstraction of a node  $(o + G)$  will always yield a constraint  $(o +^{pa} b)$ , where  $b$  is a basic constraint representing the partial answer obtained from  $G$ . In any case the observability  $o$  is just maintained.

**Example.** Given the sequence  $\tilde{t}$  of decorated trees in Fig.2 (a), it can be abstracted into the sequence  $\tilde{s}$  of constraints in Fig.2 (b), which is obtained by collecting in a depth-first manner the sequences of constraints produced by every node in  $\tilde{t}$ .



**Fig. 2.** (a) A sequence  $\tilde{t}$  of decorated trees and (b) its abstraction

Suppose we want to get a cpa semantics. The problem is how to determine whether a partial answer (or its corresponding pa constraint) is “correct”, i.e. whether it will eventually lead to a successful derivation. In a decorated tree we know that a partial answer  $b$  can correspond to a node  $N$  in the tree, and the partial answer is correct if and only if there is a consistent and observable convergent leaf in the tree rooted at node  $N$ . In Fig.2 we know from  $\tilde{t}$  that  $p(x)$  and  $p(4)$  are not correct partial answers of goal  $p(x)$ . But in the pa semantics decorated trees are “flattened” into a sequence of constraints and it’s difficult to determine the essential structure of the original decorated trees only from their abstractions, so we generally cannot tell whether a partial answer constraint is correct or not. This means that we need a more informative semantic domain for cpa semantics. The similar situation occurs when we are trying to get a ccp semantics.

The problems we learn from this example are in two aspects. Firstly, according to the idea of “control compilation” we need a parameter to explicitly specify the conditions under which a partial answer (or call pattern) will eventually lead to a successful derivation; Secondly, we must have some means to determine from a sequence of constraints the essential structures of the original decorated trees.

## 4 Solutions – A Two-Step Method

A natural solution to the first aspect of problem is to add an additional parameter *success condition* to every node of a decorated tree. The resultant decorated tree is called a *Modified Decorated Tree* (MDT). By *success condition* we denote the conditions under which a decorated tree node is on at least one successful SLD-derivation. Since every convergent and consistent leaf in a decorated tree  $T$  represents a possible solution, the success condition of the root node of  $T$  can be computed by collecting all such leaves. Formally the set  $\text{MDT}$  of MDTs is defined as:

$$\begin{aligned} \text{MDT} = & \{(o+G+sc, \varepsilon) \mid o, sc \in \mathcal{O}, G \in \mathbb{G}\} \\ & \cup \{(o+G+sc, \tilde{t}) \mid o, sc \in \mathcal{O}, G \in \mathbb{G}, \text{div}(G) \text{ and } \tilde{t} \in \text{Seq}(\text{MDT})\}. \end{aligned}$$

For example, the decorated tree in Fig.1 (b) can be translated into the MDT in Fig.1(c). On the domain  $\text{Seq}(\text{MDT})$  a MDT semantics can be achieved, which constitutes the first step for achieving cpa or ccp semantics.

The second step deals with the second aspect of problem and our solution is to make use of the depth-first nature of Prolog search rules. Since every node in a decorated tree corresponds to a partial answer constraint, the structure of the original decorated trees is somewhat clear in their abstractions. So for the purpose of identifying the essential structures of the original trees we only have to record the *height* of a node in its corresponding partial answer constraint. In the following the height of a node is also called the height of its corresponding partial answer constraint. For example, in Fig.2 the heights of partial answer constraints in  $\tilde{s}$  appear in a regular pattern 0-1-2-0-1. We know immediately that the first three partial answer constraints are from the abstraction of the same decorated tree and the last two are also from the abstraction of the same decorated tree.

Now we can propose a semantic domain for cpa semantics, which consists of four kinds of constraints. We define  $\mathbb{CPA} = \mathbb{C}^c \cup \mathbb{C}^d \cup \mathbb{C}^l \cup \mathbb{C}^{cpa}$ , where  $\mathbb{C}^c$ ,  $\mathbb{C}^d$  and  $\mathbb{C}^l$  are defined in section 3, and the set  $\mathbb{C}^{cpa}$  of correct partial answer constraints is defined as  $\mathbb{C}^{cpa} = \{o + {}^x b + sc \mid o, sc \in \mathcal{O}, b \in \mathcal{B}, x \in \mathbb{N}\}$ , where  $\mathbb{N}$  is the set of non-negative integers . Every MDT will be abstracted into a sequence of constraints in  $\mathbb{CPA}$  in much the similar way to the case of pa semantics. The main difference is that the abstraction of every node  $(o + G + sc)$  will yield a cpa constraint  $(o + {}^x b + sc)$ , where  $x$  is the height of the node. When producing other kinds of constraints from a node the abstraction process simply ignores the success condition part of the node. On the domain  $\text{Seq}(\mathbb{CPA})$ , we can define the cpa semantics as an abstraction of MDT semantics.

Similar idea can be applied to get semantics for correct call patterns. We propose the set  $\mathbb{C}^{ccp}$  of correct call pattern constraints as  $\mathbb{C}^{ccp} = \{o + {}^x b + sc, p \mid o \in \mathcal{O}, sc \in \mathcal{O}, b \in \mathcal{B} \text{ and } x \in \mathbb{N} \} \cup \{{}^x sc \mid sc \in \mathcal{O} \text{ and } x \in \mathbb{N}\}$ , where  $p$  is a procedure call and  ${}^x sc$  is the pseudo-call pattern constraint. The constraints domain  $\mathbb{CCP}$  for ccp semantics can be defined as  $\mathbb{CCP} = \mathbb{C}^c \cup \mathbb{C}^d \cup \mathbb{C}^l \cup \mathbb{C}^{ccp}$ . Every MDT will be abstracted into an element of  $\text{Seq}(\mathbb{CCP})$ . We only note that the abstraction of a node  $(o + G + sc)$  will give rise to a constraint  $(o + {}^x b + sc, p)$  if  $G$  is divergent and  $p$  is the leftmost procedure call present in  $G$ , and to a constraint  $({}^x sc)$  if  $G$  is not divergent. In both cases  $x$  is the height and the success condition of the node. On the domain  $\text{Seq}(\mathbb{CCP})$  we can define ccp semantics as an abstraction of MDT semantics.

## References

- [1] R. Barbuti, M. Codish, R. Giacobazzi, G. Levi, Modeling Prolog control, Journal of Logic and Computation, 3: 579-603, 1993.
- [2] M. Codish, D. Dams, E. Yardeni. Bottom-up Abstract Interpretation of Logic Programs, Journal of Theoretical Computer Science, 124: 93-125, 1994.
- [3] P. Cousot, R. Cousot. Abstract Interpretation and Applications to Logic Programs, Journal of Logic Programming 13 (23): 103-179, 1992.
- [4] M. Gabbielli and R. Giacobazzi. Goal Independency and Call Patterns in the Analysis of Logic Programs, In Proceedings of SAC'94, 1994, pp 394-399.
- [5] M. Gabbielli, G. Levi, M.C. Meo. Observable Behaviors and Equivalences of Logic Programs, Information and Computation, 122(1): 1-29, 1995.

- [6] M. Gabbielli, and M.C. Meo. Fixpoint Semantics for Partial Computed Answer Substitutions and Call Patterns. Lecture Notes In Computer Science, 632: 84-99, 1992.
- [7] F. Spoto, Operational and Goal-independent Denotational Semantics for Prolog with cut. The Journal of Logic Programming, 42: 1-46, 2000.
- [8] F. Spoto, Giorgio Levi, Abstract interpretation of Prolog programs, Lecture Notes in Computer Science, 1548: 455-470, 1999.

# A Fingerprint Capture System and the Corresponding Image Quality Evaluation Algorithm Based on FPS200\*

Hong Huang, Jianwei Li , and Wei He

Key Lab. on Opto-electronic Technique and systems, Ministry of Education,  
Chongqing University, 400030 Chongqing, China  
19992096@cqu.edu.cn

**Abstract.** A design of the fingerprint capture system based on chip FPS200 is introduced. The capture system uses the MCU interface with DSP. In this paper, a new method based on the intrinsic directional features of fingerprint is proposed to evaluate the quality of images acquired by FPS200 sensor. Using directional map, the available area of image and clearness is checked to calculate the overall image quality score that can be used to quantitatively determine the quality of the fingerprint image. If the quality of image is unsatisfied, prompt is presented: whether the finger is wet or dry, then DSP adjusts the parameters of FPS200 sensor automatically until image of good quality has been acquired. Experimental results show that the performance of AFIS has indeed improved.

**Keywords:** FPS200 sensor, directional map, prominent direction, image quality evaluation.

## 1 Introduction

Automatic Fingerprint Identification System (AFIS) is becoming more popular in security access and E-commerce applications. However, the identification performance of such system is very sensitive to the quality of the captured fingerprint image. Since AFIS is expected to work independently, exposed to potentially a large number of users, the fingerprint sensor that is attached to the system is possibly subjected to inappropriate use. This includes the applying of one's finger that is too dry or wet. Poor quality image causes AFIS to have higher operation problems such as false acceptance and false rejection; resulting in the user facing difficulties when using AFIS for identification.

Previous related literatures can be found in various publications. However, what contribute to the bad quality of fingerprint images and how to acquire images of good quality? They haven't been touched upon. Hence, in this paper, we propose a new method to justify the quality of the fingerprint image for the authentication system.

The paper is organized as follows. The FPS200 sensor and the fingerprint capture system is briefly introduced in Section 2. In Section 3, the proposed quality evaluation algorithm is presented. The results based on the proposed algorithm are discussed in Section 4 and conclusions describe in Section 5.

---

\* This work is partially supported by Science & Technology department of Chongqing Grant # CSTC.2004AA2001- 8277-02 to Jianwei Li and Hong Huang.

## 2 FPS200 and Fingerprint Capture System

The fingerprint capture system consists of FPS200 sensor and DSP. The system can acquire fingerprint image without the help of PC. Veridicom's robust FPS200 solid-state, 500 dpi sensor is an ideal direct-contact fingerprint acquisition device. The chip's 256×300 array and thin package provide a space-saving, cost effective image area. The sensor is easy to integrate into DSP without requiring external interface devices. The sensitivity of the chip is controlled by three Registers. The Registers are the Discharge Time Register (DTR), Discharge Current Register (DCR) and the Programmable Gain Control Register (PGC). If an image which was captured from FPS200 with the default values of the registers is of poor quality, DSP can adjust the values of the registers automatically through software until image of good quality has been captured.

## 3 Assessing Fingerprint Quality

In order to lessen the burden of DSP, the quality analysis uses a sub-sampled image. The analysis samples the image at rate in  $x$  and  $y$  directions. The sub-sampled image is further divided into square blocks of different sizes  $B$ . The experimental result shows that the sub-sampling image does not affect the performance of algorithm.

### 3.1 Direction and Foreground Estimation

This step determines if a given block depicts a portion of a fingerprint and extracts a nominal direction from a foreground block. Any number of existing strategies can be adopted [1, 2,3]. For efficiency reasons, we use the method proposed by Mehtre [4]. At each pixel in a given block, a number of pixels are selected along a line segment of an orientation ( $d$ ) and pre-specified length ( $l$ ) centered around that pixel; variation in the intensities of the selected pixels is then determined by computing the sum of intensity differences  $D_d(i,j)$  between the given pixel and the selected pixels,

$$D_d(i, j) = \sum_{(i', j')} |f(i, j) - f_d(i', j')| \quad (1)$$

with  $d = 0, \pi/n, \dots, \pi$  and where  $f(i,j)$  is the intensity of a *pixel*( $i,j$ ) and  $f_d(i', j')$  are the intensities of the neighbors of along direction  $d$ . This indicates the summation of differences between the given pixel of interest, *pixel*( $i,j$ ), and a number  $l$  neighboring pixels along each of the directions. The variation in intensities is computed for  $n$  discrete orientations; the orientation at a pixel  $d'$  is the orientation of the line segment for which the intensity variation thus computed is minimal. Regions of background and portions of impressions having faint residual leftover of earlier captured prints on a dirty input device usually exhibit small intensity variation around their neighborhoods. To determine if an image pixel belongs to the background, the intensity variation  $D(i,j)$  at the *pixel*( $i,j$ ) of interest is subsequently obtained by summing up the differences in the  $n$  directions with  $D(i,j) = \sum_d D_d(i,j)$  and when  $D$  is less than a background threshold  $T$  for each  $d$ , the pixel is classified as a background pixel. When more than a fraction of pixels in a block are background pixels, the block

is regarded as background block. Using connected component analysis, foreground components that are smaller than a certain threshold fraction of the total image area  $T_3$  are considered spurious. A fingerprint with no legitimate foreground area is of poorest quality.

### 3.2 Dominant Direction

After the foreground blocks are marked, it is determined if the resulting direction for each block is prominent. The idea is that a block with a prominent direction should exhibit a clear ridge/valley direction that is consistent with most of the pixel directions in the block. Existence of a dominant direction can be assessed by computing a histogram of directions  $D_d(1)$  at each pixel in a given block. If the maximum value of the histogram is greater than a prominent threshold  $T_1$  the block is said to have a dominant direction, and is labeled as prominent. Bifurcations of ridges may often result in two dominant directions in a block. Therefore, if two or more directions of the direction histogram are greater than a bifurcation threshold,  $T_2 < T_1$ , the corresponding block is labeled as such. A post-processing step removes blocks that are inconsistent with their neighbors. If a “directional” block is surrounded by “non-directional” blocks, it is relabeled as a non-directional block. Similarly, a non-directional block surrounded by neighboring directional blocks is changed to a directional block. Using connected component analysis, finally, regions of dominant blocks with area smaller than a threshold number of blocks  $\beta$  are discarded. The result is that the fingerprint foreground image is partitioned into (i) regions of contiguous blocks with direction and (ii) regions of blocks without direction or non-contiguous blocks with direction.

### 3.3 Quality Computation

Since regions (or accordingly minutiae) near the centroid are likely to provide more information for biometrics authentication, the overall quality of the fingerprint image is computed from the directional blocks by assigning relative weight  $w_i$  for foreground block  $i$  at location  $x_i$  given by

$$w_i = \exp\{-\|x_i - x_c\|^2/(2q^2)\} \quad (2)$$

Where  $x_c$  is the centroid of foreground, and  $q$  is a normalization constant. The overall quality  $Q$  of a fingerprint image is obtained by computing the ratio of total weights of directional blocks to the total weights for each of the blocks in the foreground,  $Q = \sum_D W_i / \sum_F W_i$ . Here  $D$  is the set of directional blocks and  $F$  the set of foreground blocks. The quality  $Q$  is used as a measure of how much reliable directional information is available in a fingerprint image. If the computed  $Q$  is less than the quality threshold,  $T$ , the image is considered to be of poor quality.

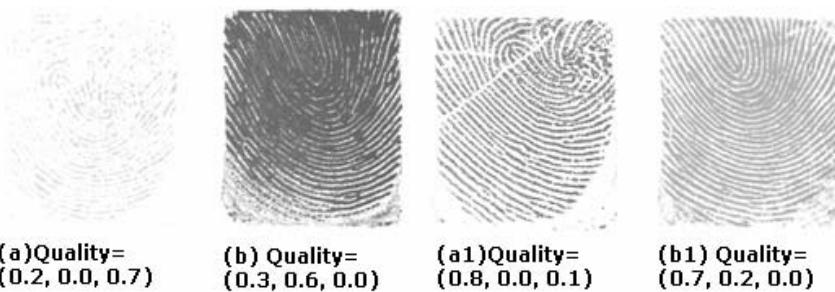
### 3.4 Dryness and Wetness

Displayed equations or formulas are centered and set on a separate line (with an extra line or half-line space above and below). Displayed expressions should be numbered for reference. The numbers should be consecutive within each section or within the contribution, with numbers enclosed in parentheses and set on the right margin. Once

it is determined that the fingerprint is of a certain poor quality, it is desirable to be able to identify a more specific cause of the low quality. We describe a method of distinguishing wet poor quality prints from dry poor quality prints based on simple statistical pixel intensity based features. The idea is that for a wet impression, there are a relatively large number of blocks whose contrast is very small. Similarly, for a dry impression, there are a relatively large number of blocks where the contrasts of their neighbors vary significantly.

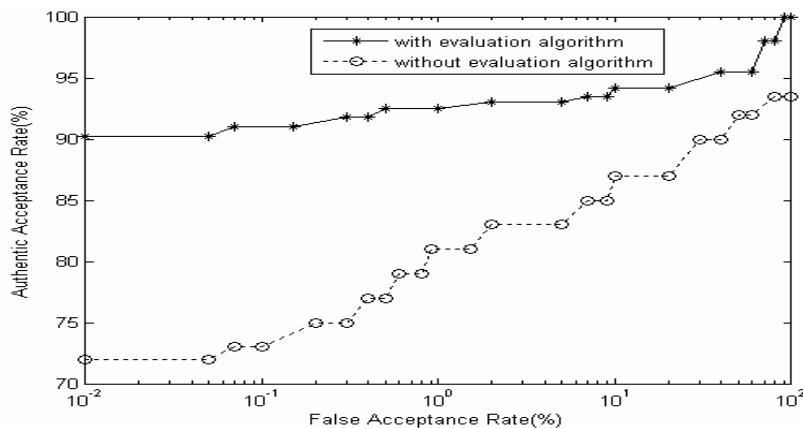
## 4 Experimental Result

The fingerprint capture system and the evaluation algorithm have been integrated into practical embedded AFIS (E-AFIS V1.0 of Chongqing University) to assess their performance. Typical operational parameters for the FPS200 and the proposed quality assessment algorithm for and 8-bit, 500 dpi 256×300 gray scale images are:  $s=2$  and  $B=7$ ,  $DTR=0x30$ ,  $DCR=0x02$ .  $PGC=0xA$ . The database was acquired from a set of 27 volunteers in the 20-73 age range (55% male). In the first test, the fingerprint evaluation algorithm was not applied. Each fingerprint image was directly acquired and matched. In the second test, the fingerprint evaluation algorithm was applied to adjust the parameters of FPS200 sensor to capture fingerprint image of high quality. Then, the verification was conducted on the improved fingerprint images. Figure 1 illustrates the results of the quality assessment of the original images with default parameters of FPS200 (a and b) and the results of the quality assessment of the images acquired through the corresponding parameters adjusting (a1 and b1).



**Fig. 1.** A fingerprint quality assessment measure. Quality(x,y,z) indicates print of overall quality x, moistness y, and dryness z.

The receiver operating curves (ROC) resulting from these two tests are shown in Figure 2. From these experimental results, we can observe that the performance of the fingerprint verification system is significantly improved when our auto-adaptive fingerprint capture system is applied. In particular, the capture system and the algorithm substantially reduced the false reject rate while maintaining the same false accept rate.



**Fig. 2.** Receiver Operating Curves (ROC); the ROC shows the improvement in verification performance using the evaluation algorithm

## 5 Conclusions

Automatically and consistently determining suitability of a given input measurement (biometrics sample) for automatic identification is a challenging problem. This paper proposes a method of the evaluation of fingerprint quality and a system of an auto-adapt fingerprint image acquisition based on the corresponding results of categorizing a poor quality fingerprint into either dry or wet prints. Currently, we notice that the value of threshold is crucial to the accuracy of algorithms, so we are examining the possibility of image evaluation based on the auto-select threshold. We are also exploring generic description of image quality in other domains such as faces, irises, etc.

## References

1. L.Hong,Y.Wan, and A.K.Jain, Fingerprint Image Enhancement: Algorithm and Performance Evaluation, IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 20, no.8., August 2001.
2. E.Lim, X.D.liana. W.Y.Yau, Fingerprint Quality and Validity Analysis, IEEE Interdental Conference On Image Processing, ICIP 2002, September 2002. pp. 335-342.
3. Ratha.NK, Bolle.R.M, Fingerprint image quality estimation, IBM Computer Science, vol. LNCS2688, Guildford, UK, Springer 2003.
4. B.M. Mehtre, Fingerprint Image Analysis for Automatic Identification, Machine Vision and Applications, Springer-Verlag, vol. 6, pp.124-139, 1993.

# Multi-agent Motion Tracking Using the Particle Filter in ISpace with DINDs

TaeSeok Jin<sup>1</sup>, ChangHoon Park<sup>2</sup>, and SooHong Park<sup>3</sup>

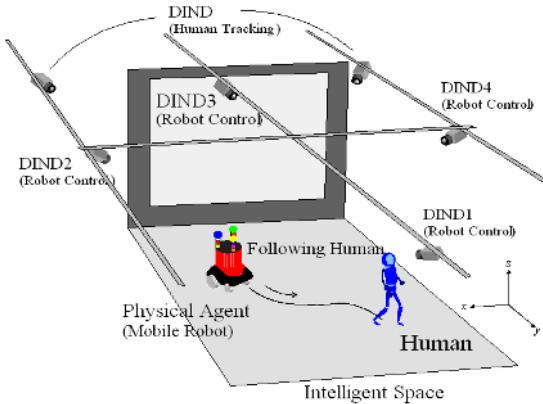
<sup>1,3</sup> Dept. of Mechatronics Engineering, DongSeo University,  
San 69-1 Churye-dong, Sasang-ku, Busan 617-716, Korea  
`{jints, shpark}@dongseo.ac.kr`

<sup>2</sup> Dept. of Game Engineering, Hoseo University,  
29-1 Sechul-ri Baebang-myun, Asan, Chungnam 336-795, Korea  
`chpark@office.hoseo.ac.kr`

**Abstract.** We present a method for representing, tracking and human following by fusing distributed multiple vision systems in ISpace, with application to pedestrian tracking in a crowd. And the article presents the integration of color distributions into particle filtering. Particle filters provide a robust tracking framework under ambiguity conditions. We propose to track the moving objects by generating hypotheses not in the image plan but on the top-view reconstruction of the scene. Comparative results on real video sequences show the advantage of our method for multi-object tracking. Simulations are carried out to evaluate the proposed performance. Also, the method is applied to the intelligent environment and its performance is verified by the experiments.

## 1 Introduction

Video object tracking in dense visual clutter, although being notably challenging, has many practical applications in scene analysis for automated surveillance, such as the detection of suspicious moving objects (pedestrians or vehicles), or the monitoring of an industrial production [1]-[4]. The quality of an object tracking system is very much dependent on its ability to handle ambiguous conditions, such as occlusion of an object by another one. To cope with such ambiguities, multi-hypotheses techniques have been developed [5]. In the standard techniques using multi-hypotheses for the state estimation and tracking, the Kalman filter is used under the premise that the noise distributions are Gaussian and the system dynamics are linear [6]. However, when tracking human movements, non-linear and non-stationary assumptions make it suboptimal to use. The particle filter algorithms track objects by generating multiple hypotheses and by ranking them according to their likelihood. They suppose that the correct hypothesis is retained [7],[8]. Many tracking filters have been proposed using this approach, defining the states as being each static posture or position of the objects and modeling a motion sequence by the composition of these states with some transitional probabilities [9]. Intelligent Space [3] is constructed as shown in Fig.1.



**Fig. 1.** Intelligent environment by distributed cameras

## 2 Image Processing Flow

### 2.1 Extraction of Objects

Fig. 3(a) is the raw image captured by the CCD camera. Extracted objects, which are human and robot, are shown in Fig. 2(b), 2(c). It is clear that this can extract the multiple objects simultaneously. When the image of the size of 320X240 pixels is captured and Pentium IV 866 MHz PC is used, this process is performed at the speed of 28 to 30 frames per second.



**Fig. 2.** Captured image and extracted objects

### 2.2 Target Regions Encoded in a State Vector Using Particle Filter

Particle filtering provides a robust tracking framework, as it models uncertainty. Particle filters are very flexible in that they not require any assumptions about the probability distributions of data. In order to track moving objects (e.g. pedestrians) in video sequences, a classical particle filter continuously looks throughout the 2D-image space to determine which image regions belong to which moving objects

(target regions). A target region over the 2D-image space can be represented for instance as follows:

$$\mathbf{r} = \{\mathbf{l}, \mathbf{s}, \mathbf{m}, \gamma\} \quad (1)$$

where  $\mathbf{l}$  is the location of the region,  $\mathbf{s}$  is the region size,  $\mathbf{m}$  is its motion and  $\gamma$  is its direction. In the standard formulation of the particle filter algorithm, the location  $\mathbf{l}$ , of the hypothesis, is fixed in the prediction stage using only the previous approximation of the state density.

$$\bar{\mathbf{r}} = (1 - \lambda) \bar{\mathbf{r}}_{t-1} + \lambda E[\mathbf{r}_t] \quad (2)$$

where  $\lambda$  weights the contribution of the mean state to the target region. So, we update the target model model during slowly changing image observations.

### 3 Object Model Update for Tracking Moving Objects

In multi-object tracking, the hypotheses are verified at each time step by incorporating the new observations (images). A well known measure of association (strength) of the relationship between two images is the normalized correlation.

$$dc_{i,j} = corr_{nor}(targ et_i; hypothesis_{i,j}) \quad (3)$$

where  $i$  : target region, and  $j$  : an hypothesis of the target region  $i$ . The observation of each hypothesis is weighted by a Gaussian function with variance  $\sigma$ .

$$h^{(i;j)} = \frac{1}{\sqrt{2\pi}\sigma_{dc}} e^{\frac{-(1-dc_{i,j})^2}{2\sigma_{dc}^2}} \quad (4)$$

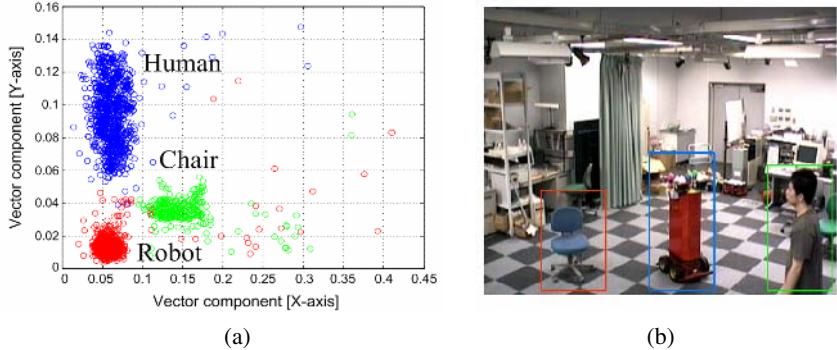
where  $h^{(i;j)}$  is the observation probability of the hypothesis  $j$  tracking the target  $i$ . The obvious drawback of this technique is the choice of the region size (defined in previous section) that will have a great impact on the results. Larger region sizes are less plagued by noise effects.

### 4 Experiments

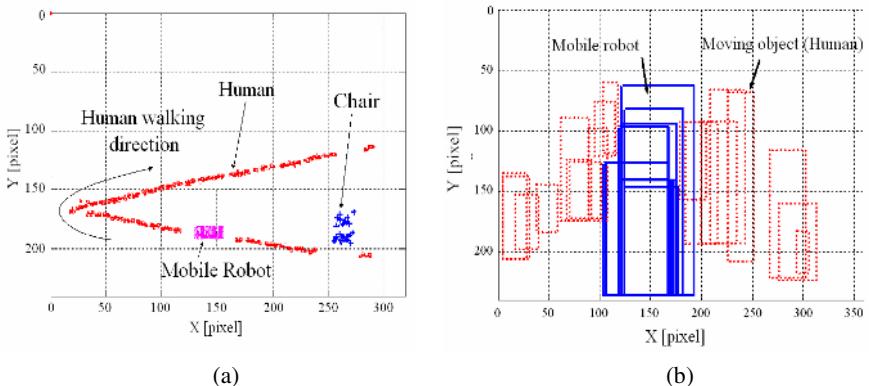
Some experiments are performed to verify this tracking method. Fig. 3 shows the experimental environment and objects that should be tracked by this method. Three objects, which are human, a mobile robot and a chair, exist in this environment.

In this experiment, the system does not have object models for these objects in advance. A mobile robot and a chair are static at the beginning and human is walking between them afterward. Fig. 3(a) shows the clustering result of the feature vectors obtained in a given time, when three objects exist in the space as shown in Fig. 3(b). The results are displayed in Fig. 4(a). In this sequence, the moving object is completely occluded by the static objects in frame, but despite of other good object candidates in

the neighborhood, the particle filter performs perfectly. Small gaps during tracking can occur when the occlusion continues for a longer period. Fig. 4(b) shows the robustness of the color-based particle filter against occlusion and strong scale changes. Furthermore, in frame 126 the effect of the initialization is illustrated.



**Fig. 3.** Experiments: moving area and models



**Fig. 4.** Experiment results: color-based particle filer against occlusion and strong scale

## 5 Conclusion

In this paper, the proposed tracking method adds an adaptive appearance model based on color distributions to particle filtering. The color-based tracker can efficiently and successfully handle non-rigid and fast moving objects under different appearance changes. Moreover, as multiple hypotheses are processed, objects can be tracked well in cases of occlusion or clutter.

The vision system of Intelligent Space needs real time processing, tracking of multiple objects, extension to cooperative multiple cameras network and overcoming partial occlusion. To realize them, it is required that model based method and feature

based method are combined efficiently. Then, new tracking strategy was proposed based on extracting the objects by background subtraction and creating color appearance model dynamically with particle filter. This strategy achieved real-time and robust tracking of multiple objects. Especially, correct matching had been kept after the occlusion among objects happened in the experimental results.

## References

1. Senior, A.: Tracking with Probabilistic Appearance Models. In Proc. ECCV workshop on Performance Evaluation of Tracking and Surveillance Systems (2002) 48-55.
2. Bierlaire, M., Antonini, G., Weber, M.: Behavioural Dynamics for Pedestrians," in K. Axhausen (Ed.), Moving through nets: the physical and social dimensions of travel, Elsevier (2003) 1-18.
3. Lee, J.H., Hashimoto, H.: Intelligent Space -concept and contents. Advanced Robotics, Vol.16, No.3 (2002) 265-280.
4. Morioka, K., Lee, J.H., Hashimoto, H.: Human Centered Robotics in Intelligent Space. IEEE International Conference on Robotics and Automation(ICRA'02) (2002) 2010-2015.
5. Choo, K., Fleet, D.J.: People tracking using hybrid Monte Carlo filtering. In Proc. Int. Conf. Computer Vision, vol. II (2001) 321-328.
6. Anderson, B., Moore, J.: Optimal Filtering. Prentice-Hall, Englewood Cliffs (1979).
7. Doucet, A., de Freitas, N., Gordon, N.: Sequential Monte-Carlo Methods in Practice. Springer Verlag, (2001).
8. Kitagawa, G.: Monte Carlo Filter and Smoother for Non-Gaussian Nonlinear State Space Models. Journal of Computational and Graphical Statistics, Vol. 5 (1996) 1-25.
9. Nummiaro, K., Koller-Meier, E., Van Gool, L.J.: Object Tracking with an Adaptive Color-Based Particle Filter. DAGM-Symposium Pattern Recognition (2002) 353-360.

# Combining Multiple Sets of Rules for Improving Classification Via Measuring Their Closenesses

Yixin Bi<sup>1</sup>, Shengli Wu<sup>1</sup>, Xuming Huang<sup>2,3</sup>, and Gongde Guo<sup>2,3</sup>

<sup>1</sup> School of Computing and Mathematics, University of Ulster, Newtownabbey,  
Co. Antrim, BT37 0QB, UK  
[{y.bi, s.wu}@ulster.ac.uk](mailto:{y.bi, s.wu}@ulster.ac.uk)

<sup>2</sup> Dept. of Computing, Univ. of Bradford, Bradford, BD7 1DP, UK

<sup>3</sup> Dept. of Computer Science, Fujian Normal Univ., Fuzhou, 350007, China  
[{G.Guo, X.Huang}@brad.ac.uk](mailto:{G.Guo, X.Huang}@brad.ac.uk)

**Abstract.** In this paper, we propose a new method for measuring the closeness of multiple sets of rules that are combined using Dempster's rule of combination to improve classification performance. The closeness provides an insight into combining multiple sets of rules in classification – in what circumambience the performance of combinations of some sets of rules using Dempster's rule is better than that of others. Experiments have been carried out over the 20-newsgroups benchmark data collection, and the empirical results show that when the closeness between two sets of rules is higher than that of others, the performance of its combination using Dempster's rule is better than the others.

## 1 Introduction

There are some recent research activities of studying the use of multiple classifiers for improving classification performance and investigating what is the best way to combine multiple classifiers. A typical approach for generating multiple classifiers is boosting techniques [1]. Several derivatives of rules-based boosting techniques have been developed. The boosting techniques are used to generate a set of weighted rules in a system called SLIPPER [2]. Weiss and Indurkhy proposed a lightweight rule induction method which is used to generate compact Disjunctive Normal Form (DNF) decision rules [3].

There are different approaches to combining multiple classifiers. It sometime depends on the nature of classifier outputs [4]. If classifier outputs are class labels, a majority or weighted majority voting is used [5]. If continuous outputs like similarity scores or posteriori probabilities are available, a minimum, maximum, average or some other linear combination have been suggested [6]. If the classifier outputs are interpreted as fuzzy membership values, belief values or evidence, fuzzy rules [7], and Dempster-Shafer techniques [4] are used.

In this study, we develop a novel approach to combining multiple sets of rules in terms of combining multiple classifiers by using Demspter's rule of combination [8]. The distinguishing aspects of our approach include: 1) the multiple sets of rules are constructed from multiple sets of attributes generated by our rough

sets-based algorithm, instead of a single set of rules as used in conventional approaches; 2) a novel method for modeling the classification decisions resulting from multiple sets of rules as multiple pieces of evidence which indicate how likely it is that a new instance belongs to a decision class; 3) a novel closeness measure provides an insight into combining multiple sets of rules in classification – in what circumstance the performance of combinations of some sets of rules by using Dempster's rule of combination is better than that of others. In this paper, we particularly focus on the third aspect – the closeness measure.

## 2 Rough Sets for Rule Generation

In rough sets [9], objects or instances are organized into an information system. An *information system*  $\mathcal{I}$  is a system  $\langle U, A \rangle$ , where

1.  $U = \{u_1, u_2, \dots, u_i, \dots, u_{|U|}\}$  is a finite non-empty set, called the *universe* or *object space*; elements of  $U$  are called *objects*;
2.  $A = \{a_1, a_2, \dots, a_j, \dots, a_{|A|}\}$  is also a finite non-empty set; elements of  $A$  are called *attributes*;
3. for every  $a \in A$  there is a mapping  $a$  from  $U$  into some space  $a : U \rightarrow a(U)$ , and  $a(U) = \{a(u) \mid u \in U\}$  is called the *domain* of attribute  $a$ .

An information system can be expressed intuitively in terms of an *information table* as follows:

$$\frac{U \setminus A \mid a_j}{u_i \quad | \quad a_j(u_i)} \quad (1)$$

where attribute  $a_j$  has domain  $a_j(U) = \{a_j(u_i) \mid i = 1, 2, \dots, |U|\}$ . When there is no repetition of objects, the information table is a relation. We want to find dependencies in relations and to discover rules which are hidden in information tables. We can consider some attributes are condition attributes and some others are decision ones. Then we can discover the relation between condition and decision, predict decision from condition. Therefore, an information system  $\langle U, A \rangle$  is called a decision table, if we have  $A = C \cup D$  and  $C \cap D = \emptyset$ , where attributes in  $C$  are called condition attributes and attributes in  $D$  are called decision attributes.

The rough set-based approach to inductive learning consists of two steps. The first step is to find multiple single covering solutions for all training instances held in a decision table. Specifically, given a set of condition attributes  $C$  and a subset  $B \subseteq C$ , a covering attribute set is found directly by computing its dependency degree. The direct solution involves adding an attribute at a time, removing the attribute covered by the attribute set, and then the process is repeated until the dependency of  $B$  is equal (or approximately equal) to  $C$ . At the end of the induction of conjunctive attributes, more than one set of attributes will be found.

The second step is to transform multiple sets of attributes to multiple sets of rules and weight each rule based on counting the identical attribute values  $a_j(u_i)$ . As a result, a number of rule sets will be produced, denoted by  $\mathfrak{R} = \{R_1, R_2, \dots, R_{|\mathfrak{R}|}\}$ , where  $R_i = \{r_{i1}, r_{i2}, \dots, r_{|R_i|}\}$ ,  $1 \leq i \leq |\mathfrak{R}|$ . Each set of

rules is called a intrinsic rule set, referred to as a classifier. It plays an independent role in classifying unseen instances. The relation between two sets of intrinsic rules is in DNF as the rules are in  $R_i$ . To examine the effectiveness of using multiple classifiers to classify unseen cases, our approach does not involve any rule optimization between multiple sets of rules. More details about these algorithms can be found in [10].

### 3 Evaluation

For our experiments, we have chosen a public benchmark dataset 20-newsgroup. It consists of 20 categories ( $C_1, C_2, \dots, C_{20}$ ), and each category has 1,000 documents, so the dataset contains 20,000 documents in total. By using the rough sets-based algorithm, ten sets of attributes (reducts) have been selected and ten corresponding sets of intrinsic rules have been constructed, which are denoted by  $R_0, R_1, \dots, R_9$ . To examine the performance of the different rule sets in combinations, we first rank these rulesets in descending order based on their performance, and then use 70% classification accuracy as a cut-off point to select 6 sets of rules for combinations. Table 1 presents the estimates of classification accuracy of these rulesets.

**Table 1.** The estimated performance of the six rulesets (%)

	$C_1$	$C_2$	$C_3$	$C_4$	$C_5$	$C_6$	$C_7$	$C_8$	$C_9$	$C_{10}$	$AV$
$R_0$	86.79	56.81	68.84	63.88	58.02	79.21	93.38	84.03	81.39	92.45	76.48
$R_1$	87.73	55.02	71.64	59.38	45.00	77.94	93.41	82.20	80.35	92.73	74.54
$R_4$	86.55	57.86	68.95	60.88	53.36	78.83	93.81	88.89	92.01	93.26	77.44
$R_5$	88.43	58.94	66.51	63.80	54.21	78.36	93.63	86.86	81.48	93.53	76.58
$R_6$	89.57	58.33	74.41	58.55	42.89	80.90	92.98	85.17	79.21	93.31	75.53
$R_8$	85.84	53.79	61.76	58.47	40.42	79.38	93.81	85.97	80.00	93.80	73.32

The rulesets have been combined using Dempster's rule of combination in various ways. We first take  $R_4$  with the highest classification accuracy, and then combine it with  $R_0, R_1, R_5, R_6, R_8$ . The results are denoted by  $DS_{40}, DS_{41}, DS_{45}, DS_{46}, DS_{48}$ . We then take the second best ruleset  $R_5$  to combine  $R_0, R_1, R_6, R_8$ . The combined rulesets are ranked and the best combined ruleset  $DS_{45}$  is chosen for the next round of combination. It is combined with  $R_0$ , and then combined with  $R_6$ , and so forth, resulting in a ranked list of the combinations  $DS_{450}, DS_{4506}, DS_{45061}$ , and  $DS_{450618}$ . From the resulting combinations, we find that the classification accuracies have dropped with the addition of more rulesets and there is no indication that the combinations of more than two rulesets can outperform the combined ruleset  $DS_{45}$ . Therefore, it is concluded that the combination of the best ruleset with the second best ruleset provides the best combination in achieving the biggest predictive performance.

## 4 Measuring Closeness Between Pairs of Rulesets

Now we attempt to give an answer to the question why such results can be achieved by an empirical measure called *closeness*. Let  $R_1, R_2, \dots, R_L$  be a group of rulesets, and let  $C = \{c_1, c_2, \dots, c_{|C|}\}$  be a set of classes. It is assumed that the rulesets have been ranked in descending order where  $R_1$  has the highest performance and  $R_L$  has the lowest performance. We also denote by  $\pi_R$  the final decisions of ruleset  $R$ , called a decision performance profile as follows:

$$\pi_R = \langle \pi_R(c_1), \pi_R(c_2), \dots, \pi_R(c_{|C|}), AV_R \rangle \quad (2)$$

where  $\pi_R(c_i)$  is a classification accuracy on category  $c_i$  and  $AV_R$  is the mean value of the classification accuracies of  $|C|$  classes. The closeness between a pair of rulesets can be calculated by formula (4), where  $\varepsilon$  is a coefficient obtained by tuning. In this setting, its value is 0.4.

$$\lambda_{R_i} = \{c \in C \mid \pi_{R_i}(c) > \pi_{R_j}(c), i \neq j\} \quad (3)$$

$$closeness(R_i) = \varepsilon \times AV_{R_i} + (1 - \varepsilon) \times \lambda_{R_i} \quad (4)$$

Having introduced the closeness measure, we now examine the closeness between any pairs of rulesets. From Table 1, we know ruleset  $R_4$  has the best performance, therefore we first take  $R_4$  as a reference ruleset, and then compare the closeness of pairs of rulesets with it.

By using formulae (3) and (4), we calculate the closeness between the classifiers, Table 2 presents the decision profile of  $R_0$  and  $R_5$ , and their  $closeness(R_0)$  and  $closeness(R_5)$  to the best ruleset  $R_4$  are 32.99 and 34.23 respectively. We also calculate another pair of rulesets  $R_5$  and  $R_6$  as shown in Tables 3, the

**Table 2.** Accuracy difference between rulesets  $R_0$  and  $R_5$

	$C_1$	$C_2$	$C_3$	$C_4$	$C_5$	$C_6$	$C_7$	$C_8$	$C_9$	$C_{10}$
$R_0$	-0.24	1.05	0.11	-3.00	-4.66	-0.37	0.42	4.86	10.62	0.81
$R_5$	-1.88	-1.08	2.44	-2.92	-0.85	0.47	0.18	2.03	10.53	-0.26

**Table 3.** Accuracy difference between rulesets  $R_5$  and  $R_6$

	$C_1$	$C_2$	$C_3$	$C_4$	$C_5$	$C_6$	$C_7$	$C_8$	$C_9$	$C_{10}$
$R_5$	-1.88	-1.08	2.44	-2.92	-0.85	0.47	0.18	2.03	10.53	-0.26
$R_6$	-3.02	-0.48	-5.46	2.34	10.47	-2.07	0.83	3.72	12.80	-0.05

$closeness(R_5)$  and  $closeness(R_6)$  to  $R_4$  are 34.83 and 32.01 respectively. From these results, it can be seen that  $R_5$  has the highest closeness to ruleset  $R_4$  among the two pairs of rulesets. This provides a possible explanation to why the combination of rulesets  $R_4$  and  $R_5$  using DS can achieve the best classification accuracy. Similarly, we examine the closeness between any other pairs of rulesets using this measure. The details can be found in [10].

## 5 Conclusion

In this paper, we have presented a method built on Rough Set theory for generating multiple sets of rules, which can be combined using Dempster's rule of combination. From our experimental results, it is observed that the combination which can achieve the highest predictive performance consists of the two sets of rules in which one is the best and the other is the second best. In an attempt to explore why the combination of the best and the second best rulesets can achieve better performance than the other combinations, we propose an closeness measure. Based on our empirical study, we found that when a ruleset has the highest closeness to the best ruleset, then the combination of it with the best ruleset can achieve the best performance.

## References

1. Y. Freund and R. Schapire, "Experiments with a new boosting algorithm". Proc of the Thirteenth International Conference on Machine Learning, 1996, pp148-156.
2. W. W. Cohen, and Y. Singer, "Simple, Fast, and Effective Rule Learner". Proc of Annual Conference of American Association for Artificial Intelligence, 1999b, pp335-342. 4.
3. S.M. Weiss and N. Indurkhy, "Lightweight Rule Induction". In Proc of the International Conference on Machine Learning, 2000. 5.
4. L. Xu, A. Krzyzak and C.Y. Suen, "Several Methods for Combining Multiple Classifiers and Their Applications in Handwritten Character Recognition". IEEE Trans. on System, Man and Cybernetics, 1992, Vol. 22 (3), pp418-435.
5. C.J. Whiteaker and L. Kuncheva, "Examining the Relationship between Majority Vote Accuracy and Diversity in Bagging and Boosting". Technical report. University of Wales, Bangor, 2003.
6. J. Kittler, M. Hatef, R.P.W. Duin and J. Matas, "On Combining Classifiers. IEEE Transactions on pattern Analysis and Machine Intelligence". 1998, Vol. 20(3), pp226-239.
7. S.B. Cho and J.H. Kim, Combining Multiple Neural Networks by Fuzzy Integral for Robust Classification, IEEE Trans. Systems, Man, and Cybernetics, vol. 25, no. 2, pp. 380-384, 1995.
8. Shafer, G. A Mathematical Theory of Evidence, Princeton University Press, Princeton, New Jersey, 1976.
9. J.W. Guang and D.A. Bell, "Rough computational methods for information systems". Artificial Intelligence 1998, 105:77-103.
10. Bi, Y. Combining Multiple Classifiers For Text Categorization Using The Dempster-Shafer Theory Of Evidence, PhD thesis, University of Ulster, 2004.

# Multiple SVMs Enabled Sales Forecasting Support System

Yukun Bao, Zhitao Liu, Rui Zhang, and Wei Huang

Department of Management Science & Information System, School of Management,  
Huazhong University of Science and Technology, Wuhan 430074, China  
[yukunbao@mail.hust.edu.cn](mailto:yukunbao@mail.hust.edu.cn)

**Abstract.** This paper proposes a multiple SVMs enabled sales forecasting support system (SFSS). The SFSS has a two-stage system architecture. In the first stage, agglomerative hierarchical clustering(AHC) is used to partition the goods into several patterns based on similarity measure. In the second stage, multiple SVMs that best fit partitioned patterns are constructed by finding the appropriate kernel function and the optimal free parameters of SVMs. The experiment shows that this integrated system achieves significant improvement in forecasting performance compared with single SVMs models.

## 1 Introduction

Sales forecasting for a firm's effective operation has long been recognized by both practitioners and academics [1]. Due to the weakness of traditional time series forecasting methods in dealing with noisy and nonstationary data, new nonlinear forecasting techniques have been payed more attention [2].

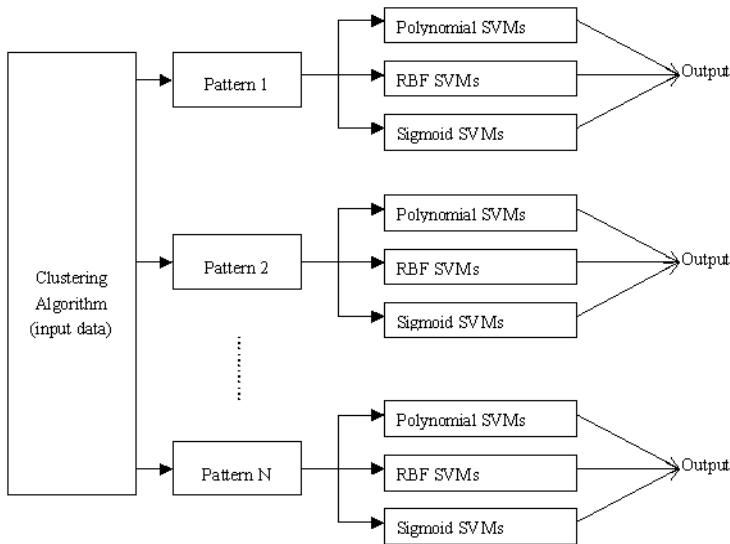
Recently, support vector machines (SVMs) have been proposed as a novel technique in time series forecasting [3, 4, 5, 6]. In the modeling of time series, two of the key problems are noise and non-stationarity. In general, it is hard for a single model including SVMs to capture such a dynamic input-output relationship inherent in the data. A potential solution is to use a mixture of forecasting support system architecture [7].

This paper presents our efforts on establishing a sales forecasting support system (SFSS) enabled by multiple SVMs.

## 2 Architecture and Algorithms Related of SFSS

### 2.1 Architecture

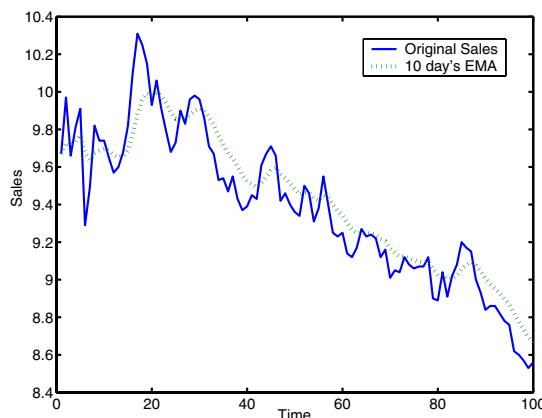
We design a two-stage architecture shown in Fig.1. In the first stage, clustering algorithm is used to partition the goods' into several demand patterns. In the second stage, a multiple-SVMs are competed to solve partitioned sales patterns. For each particular pattern, only the SVMs that best fits it is used for the final prediction. This is very different from a single SVMs model that learns the whole input space globally and thus cannot guarantee that each local input data is the best learned. Moreover, the prior clustering can reduce the computational time dramatically compared with non clustering one.



**Fig. 1.** Architecture of SFSS

## 2.2 Data Processing

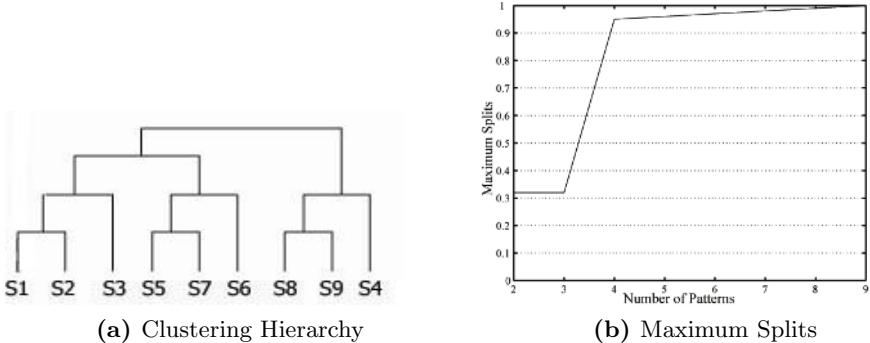
Finding good representation for the data is a crucial and labor intensive task. Depending on different problems, it is necessary to perform some data processes in order to satisfy the requirement of the special models. In our experimental we choose 10 day's EMA as data processing approach Fig.2 shows.



**Fig. 2.** Original Sales compared with 10 day's EMA in our experiment

### 2.3 Clustering Algorithm

We use classical agglomerative hierarchical clustering algorithm to group the goods when it is based on a similarity measure. Details of this algorithm can be found in ref. [8, 9]. Fig.3 shows agglomerative hierarchical clustering algorithm results.



**Fig. 3.** Experimental results of AHC in our experiment

### 2.4 Kernel Function Parameters Selection

Kernel function parameters selection is a key point of our proposed system. In following part, we raise an example to show how we find the best parameters for RBF kernel function.

There are two parameters while using RBF kernels:  $C$  and  $\gamma$ . It is not known beforehand which  $C$  and  $\gamma$  are the best for one problem; consequently some kind of model selection (parameter search) must be done [5]. We use a grid-search on  $C$  and  $\gamma$  using cross-validation. Basically pairs of  $(C, \gamma)$  are tried and the one with the best cross-validation accuracy is picked. We found that trying exponentially growing sequences of  $C$  and  $\gamma$  is a practical method to identify good parameters (for example,  $C = 2^{-5}, 2^{-3}, \dots, 2^{15}; \gamma = 2^{-15}, 2^{-13}, \dots, 2^3$  ).

## 3 Experimental Results

### 3.1 Data Sets

As an experiment setting, We selected 140 daily sales data from a manufacturing firm's management information system. By AHC clustering, all the goods are partitioned into 9 sales patterns.

### 3.2 Performance Evaluation

The prediction performance is evaluated using the normalized mean squared error (NMSE). NMSE is the measures of the deviation between the actual and predicted values. The smaller the values of NMSE, the closer are the predicted

time series values to the actual values. The NMSE of the test set is calculated as follows:

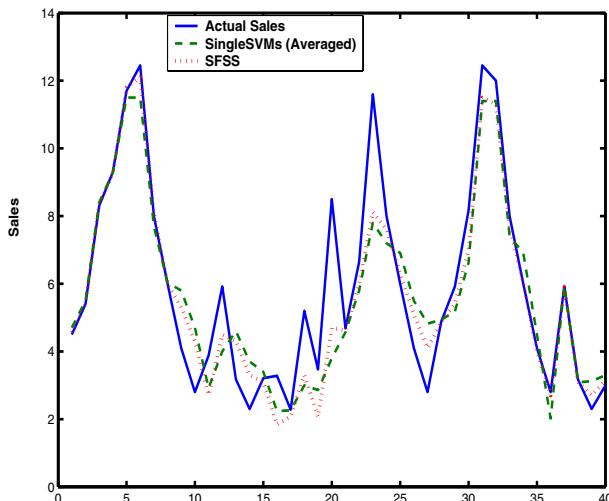
$$NMSE = \frac{1}{\delta^2 n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (1)$$

$$\delta^2 = \frac{1}{n-1} \sum_{i=1}^n (y_i - \bar{y})^2 \quad (2)$$

where  $n$  represents the total number of data points in the test set.  $\hat{y}$  represents the predicted value.  $\bar{y}$  denotes the mean of the actual output values.

### 3.3 Results

Table 1 shows the NMSE values of different single SVMs for each sales pattern. Bold figure indicates the best SVMs kernels selected by SFSS and tells out the best prediction method under our numerical case. Figure 4 illustrates the predicted and actual values. The (blue) solid line is the actual value. The (red) dot line is the predicted value of SFSS , and the (green) dash line is the predicted value of the averaged single SVMs model. From the figure, it can be observed that the SFSS forecast more closely to the actual values than the averaged single SVMs in most of the time period.



**Fig. 4.** Forecasting results comparison of Cluster 1

**Table 1.** NMSE Values of comparative methods

Kernel	SP1	SP2	SP3	SP4	SP5	SP6	SP7	SP8	SP9
Poly.	0.574	0.602	0.552	<b>0.425</b>	0.782	0.481	0.685	0.548	0.450
RBF	<b>0.407</b>	0.458	<b>0.385</b>	0.572	0.515	<b>0.472</b>	<b>0.394</b>	0.502	<b>0.327</b>
Sigmoid	0.832	<b>0.440</b>	0.623	0.827	<b>0.429</b>	0.562	0.739	<b>0.492</b>	0.551

## 4 Conclusions and Acknowledgements

The experimental results show that the SFSS is more effective and efficient in forecasting demand than the single SVM model. This forecasting support system has significance in helping marketing managers foresee the market tendency and enhancing the plans for both logistics and production departments.

This research is granted by National Science Foundation of China(70401015) and Hubei Provincial Key Social Science Research Center of Information Management.

## References

1. Remus W, Simkin MG. Integrating forecasting and decision making. In: Makridakis S, Wheelwright SC, editors. *The handbook of forecasting: a manager's guide*. 2nd ed. New York, USA: Wiley, 1987.
2. Fildes R, Hastings R. The organization and improvement of market forecasting. *Journal of Operation Research Society*, 45(1994) 1-16.
3. Müller K.R., Smola A.J., Rätsch G., Schölkopf B., Kohlmorgen J., and Vapnik V. Prediction time series with support vector machines. *Lecture Notes in Computation Science* Vol.1327,1997,999-1004.
4. Francis, E.H.T. and Cao L. Application of support vector machines in financial time series forecasting. *Omega: International Journal of Management Science*, 29(2001) 309-317.
5. Smola, A.J. and B. Schölkopf, A tutorial on support vector regression. *Statistics and Computing*, 14(2004) 199-222.
6. Bao YK, Lu YS and Zhang JL. Forecasting stock price by SVMs regression. *Lecture Notes in Artificial Intelligence*, Vol.3192,2004:295-303.
7. Cao L. Support vector machines experts for time series forecasting. *Neurocomputing*, 51(2003) 321-339.
8. Saporta, G., 1990. *Probabilités, Analyse de Données et Statistique*. Editions Technip, Paris.
9. Gordon, A.D., 1999. *Classification*, 2nd Edition. Chapman & Hall, London.

# The Application of B-Spline Neurofuzzy Networks for Condition Monitoring of Metal Cutting Tool

Pan Fu<sup>1</sup> and A.D. Hope<sup>2</sup>

<sup>1</sup> Mechanical Engineering Faculty, Southwest JiaoTong University  
610031 Chengdu, China  
pfu@home.swjtu.edu.cn

<sup>2</sup> Systems Engineering Faculty, Southampton Institute  
Southampton SO14 OYN, U.K.  
tony.hope@solent.ac.uk

**Abstract.** Metal cutting operations constitute a large percentage of the manufacturing activity. Cutting tool condition monitoring is certainly the important monitoring requirement of unintended machining operations. A multi-purpose intelligent tool condition monitoring technique for metal cutting process will be introduced in this paper. The knowledge based intelligent pattern recognition algorithm is mainly composed of a fuzzy feature filter and algebraic neurofuzzy networks. It can carry out the fusion of multi-sensor information to enable the proposed intelligent architecture to recognize the tool condition successfully.

## 1 Introduction

Traditional metal cutting tool change strategies result in frequent tool changes and high production costs. Automatic machining operations rely greatly on the ability to monitor the cutting tool wear states, so the machining quality can be assured. In an unmanned production environment, this function can be performed by an integrated system composed of sensors and intelligent signal processing algorithms.

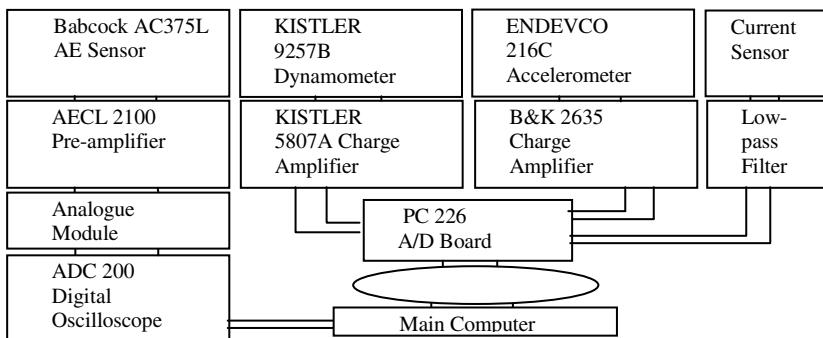
The acoustic emission (AE) signal was applied to monitor the tool life of the grinding wheel [1]. The input features were derived from measurements of acoustic emission during face milling and topography of the machined surfaces [2]. Several works had been done in order to build grinding monitoring systems that could determine the end of grinding wheel life [3]. X. Li, etc. showed that the r.m.s. of the different frequency bands of vibration measured indicates the drill wear condition [4]. Tool breakage and wear conditions were monitored in real time according to the measured spindle and feed motor currents for drilling operations, respectively [5]. In one study, the in-process monitoring method using the power sensor was proposed to detect some malfunctions due to the tool wear in the internal grinding process [6].

A new on-line fuzzy neural network (FNN) model was developed with the functions of classifying flank and crater wear [7]. An advanced approach for online and indirect tool wear estimation in turning using neural networks was developed [8]. Two methods using Hidden Markov models, as well as several other methods that directly use force and power data were used to establish the health of a drilling tool

[9]. In this study, a unique neurofuzzy network based pattern recognition algorithm has been developed to accomplish multi-sensor information integration and tool wear state classification.

## 2 Tool Condition Monitoring System and Signal Feature Extraction

The tool wear monitoring system is composed of four types of sensors, signal conditioning devices and the main computer, as shown in Fig. 1. Part of the condition monitoring experiments were carried out at the Advanced Manufacturing Lab. of Southampton Institute, U.K., on a Cincinnati Milacron Sabre 500 (ERT) Vertical Machining Centre. Here, the wear value of the milling cutter was monitored. A typical group of features extracted from the original time domain and frequency domain for the further pattern recognition are as follows. Power consumption signal: mean value; AE-RMS signal: mean value, skew and kurtosis; Cutting force, AE and vibration: mean value, standard deviation and the mean power in 10 frequency ranges.



**Fig. 1.** The tool condition monitoring system

## 3 Fuzzy Clustering Feature Filter

For extracted features as mentioned above, their effectiveness for reflecting tool wear state may change under different machining conditions. In order to improve the efficiency and reliability of the neurofuzzy pattern recognition algorithm, redundant features should be removed. Here the fuzzy clustering technique is applied to develop an effective fuzzy feature filter. Bezdek suggested using an objective function approach for clustering the data into hyperspherical clusters[10].

A family of fuzzy sets  $\{A_i, i = 1, 2, \dots, c\}$  is defined as a fuzzy c-partition on  $X$ . The membership value of  $k$ -th data point in  $i$ -th fuzzy set is  $\mu_{A_i}(X_k) = \mu_{ik} \in [0,1]$  and  $\sum_{i=1}^c \mu_{ik} = 1$  (for all  $k=1,2,\dots,n$ ). The objective function  $J_m$  can then be defined.

$$J_m(U, V) = \sum_{k=1}^n \sum_{i=1}^c (\mu_{ik})^{m'} \|X_k - V_i\|^2 \quad (1)$$

where  $U$  is the fuzzy c-partition matrix.  $V$  is a vector of cluster centers,  $m'$  is a weighting parameter. The objective function  $J_m$  measures the weighted sum of distances between cluster centers and elements in the corresponding fuzzy clusters. Hence the optimum fuzzy  $C$ -partition can be obtained by minimizing the objective function:

$$J^*(U^*, V^*) = \min J(U, V) \quad (2)$$

In this study a fuzzy clustering feature filter has been developed to remove the redundant signal features. Those features whose variations of cluster centers for different models are smaller than a threshold  $\varepsilon_1$ , or whose variation of practical values corresponding to their own cluster center are bigger than a threshold  $\varepsilon_2$  should be removed. This is because they don't have recognizable variation along with the development of tool wear values or they don't have stable values. After the feature filtering process, the size of the input space of the neurofuzzy networks for pattern recognition is greatly reduced but its high accuracy and reliability can be assured.

## 4 B-Spline Neurofuzzy Networks and the Application

A neurofuzzy system processes information using fuzzy reasoning techniques, but it can be trained using normal type learning algorithms because it also has a multi-layer ANN structure. The combination of the rule based representation and adaptive numeric processing can lead to a robust modeling system. Some neurofuzzy systems use B-spline or Gaussian basis functions to represent fuzzy membership functions [11][12].

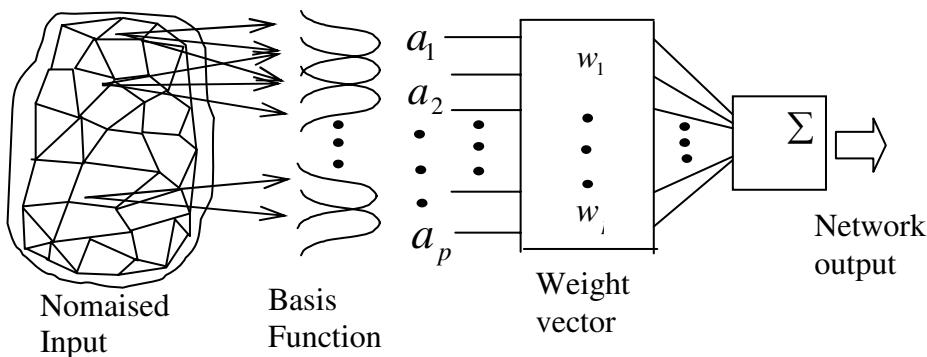
A robust neurofuzzy system can model the relationship between the signal features and the cutting tool wear values. The relation between signal features and tool wear values can be expressed by the descriptions like this: if *the power consumption is large* and *cutting force is medium* and .... and *the AE-RMS is small* then *the tool wear value is large*. In tool condition monitoring processes, signal features can be treated as fuzzy sets and B-spline basis functions, piecewise polynomials of order  $k$ , can be used to represent their membership functions. B-spline basis functions can implement binary fuzzy sets ( $k=1$ ), triangular fuzzy sets ( $k=2$ ), quadratic fuzzy sets ( $k=3$ ) and cubic fuzzy sets ( $k=4$ ). From a fuzzy viewpoint, the univariate B-spline basis functions represent fuzzy linguistic statements, such as *the cutting force is positive small*. Multivariate fuzzy sets are formed to represent fuzzy conjunction. This link enables the B-spline networks to be interpreted as a set of fuzzy rules and allows modeling and convergence to be derived for the fuzzy networks.

A number of fuzzy rules are needed to satisfactorily describe the relationship between the input and output. These rules are combined by a fuzzy union operator to form a fuzzy algorithm. To implement the fuzzy algorithm, fuzzy sets are used to

represent linguistic statements and functions are chosen to implement the fuzzy logic operators *AND*, *OR*, and *IF(.)*, *THEN(.)*. Fuzzy intersection and implication are realized by using the algebraic functions (“product” and “addition” operators respectively). Then the input-output relationship is no longer undetermined. It can be presented by a non-linear multivariable function that can be initialized and illustrated by fuzzy rules. By adopting fuzzy sets defined by B-spline basis functions and fuzzy operators defined by algebraic functions, then for the fuzzy singleton input, the centre of gravity defuzzified output of the fuzzy system can be given as:

$$y = \sum_{j=1}^n \omega_j^j B_j^k(x) \quad (3)$$

The structure of the resulting algebraic neurofuzzy system is shown in Fig. 2. Neurofuzzy systems can be regarded as a particular implementation of the ordinary fuzzy system. Here the multivariate fuzzy input sets  $B_j^k(x)$  are termed as basis functions  $a_j(X)$ . The multivariate basis functions are defined on a lattice, which is created from the projection of all the individual knot vectors parallel to the remaining input axes. The weight associated with each basis function shows an estimate of the value of the network's output given that the input lies within that set. The neurofuzzy networks calculate the outputs as a linear weighted combination of the fuzzy input set memberships. The linear coefficient is the adjustable weights ( $w_j$ ).



**Fig. 2.** The algebraic neurofuzzy network

The nemofuzzy system can be a powerful tool for tool condition monitoring. The real-valued input (filtered signal features) are transformed to fuzzy sets with membership functions represented by B-splines. A fuzzy rule base is then established to describe the mapping between the systems input (filtered signal features) and output states (tool wear value, or VB). So the network can learn and memorize the feature patterns of the models (tool with known wear values) and be able to classify the incoming objects (tools being monitored).

## 5 The Comprehensive Verification of the Neurofuzzy Pattern Recognition Architecture

The tool condition monitoring system together with the fuzzy pattern recognition algorithm developed from this study had been verified extensively using orthogonal experimental design method. This method can reveal the influence of each factor and their mutual interaction on experimental results with a minimum number of experiments. In the face milling environment, 4 important cutting parameters, cutting speed; feed rate; cutting depth and workpiece material were selected as factors and the orthogonal table  $L_{27}(3^{13})$  was selected. Under 27 cutting conditions, cutting tool wear states (new, medium worn, severely worn) were correctly recognized with average recognition confidence of 78%. Because the experiments designed by an orthogonal table are representative, the recognition results illustrated that the developed tool condition monitoring system has consistent high accuracy and reliability under a wide range of cutting conditions.

## References

1. Hsieh, J., Inasaki, I.: Monitoring of dressing and grinding process with acoustic emission signals. *Trans. Japan Soc. Mech. Eng.* (1985) vol. 51, 2174-2179
2. Wilkinson, P., Reuben, R. L., Jones, J. D. C.: Tool wear prediction from acoustic emission and surface characteristics via an artificial neural network. *Mechanical Systems and Signal Processing*. 6 (1999) 955-966
3. Diniz, A. E., Hassui, J., Oliveira, J. F., Gomes, J.: Experimental evaluation on grinding wheel wear through vibration and acoustic emission. *Wear (Switzerland)* 6 (1998) 7-14
4. Li, X., Dong, S.: Hybrid learning for tool wear monitoring. *Int. J. of Advanced Manufacturing Technology*. 5 (2000) 303-307
5. LI, X. L., Tso, S. K., Wang, J.: Real-time tool condition monitoring using wavelet transforms and fuzzy techniques. *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*. (2000) vol. 30, 352-357
6. Wakuda, M., Inasaki, I.: Diagnosis of internal grinding process by power monitoring. *Trans. Japan Soc. Mech. Eng.* (1991) vol. 57, 3107-3112
7. Chungchoo, C., Saini, D.: On-line tool wear estimation in CNC turning operations using fuzzy neural network model. *Int. J. of Machine Tools and Manufacture*. 2 (2002) 70-76
8. Sick, B.: Tool wear monitoring in turning: A neural network application. *Measurement and Control*, 4 (2001) 207-212
9. Ertunc, H. M., Loparo, K. A.: A decision fusion algorithm for tool wear condition monitoring in drilling. *Int. J. of Machine Tools and Manufacture*. 1 (2001) 1347-1362
10. Bezdek, J. C.: *Pattern recognition with fuzzy objective function algorithm*. Plenum Press, USA. (1981)
11. Brown, M., Harris, C. J.: A perspective and critique of adaptive neurofuzzy systems used for modeling and control applications. *Int. J. Neural Systems* 2 (1995) 197-220
12. Wang, L. X.: *Adaptive fuzzy systems and control: design and stability analysis*. Englewood Cliffs, Prentice Hall, USA. (1994)

# Simplified Fuzzy-PID Controller of Data Link Antenna System for Moving Vehicles

Jong-kwon Kim<sup>1</sup>, Soo-hong Park<sup>2</sup>, and TaeSeok Jin<sup>3</sup>

<sup>1</sup> Shin Dong Digitech Co., Ltd., 3F, Jinyoung Bldg, 1213-1, Choryang-1 Dong,  
Pusan 601-839, Korea  
[jkkim@shindong.com](mailto:jkkim@shindong.com)

<sup>2,3</sup> Department of Mechatronics Engineering, Dongseo University,  
Sasang Ku, Pusan 617-716, Korea  
[{shspark, jints}@dongseo.ac.kr](mailto:{shspark, jints}@dongseo.ac.kr)

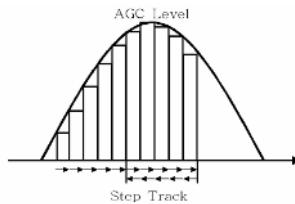
**Abstract.** A simplified Fuzzy-PID controller is designed for 2-axes antenna stabilization and tracking system. Next, the performance of the controller is further verified by computer simulations with Matlab, a 2-Axes antenna, and a small unmanned test helicopter. With the advantages of embedded controller and simplified fuzzy control theory, high performance techniques for antenna tracking control are designed. A comparison between the performance of the conventional PID controller and the Fuzzy-PID controller, which is designed by the same PID control gains, is made as a way to verify the performance of the designed antenna servo control system. The proposed Fuzzy-PID controller has better superior performance than the conventional PID controller with respect to all cases of simulations and experiments.

## 1 Introduction

The performance of the stabilized tracking system is mainly depended on the servo control system, which drives the antenna pedestal. However, the outdoor antenna servo system is subjected to significant torque disturbances from wind pressures and gusts on the antenna structures, as well as bearing and aerodynamic frictions. This control system should provide a sharp directivity in spite of the environmental disturbances and internal uncertainties [1, 2]. The more flexible and rapid prototyping design is needed for these embedded antenna control system. Therefore, the implementation of a robust yet simple structural real-time controller is needed for the precise generation of the referential signal and robust tracking performance in real environment [3, 4, 5]. As mentioned earlier, in the following section, a simplified fuzzy-PID controller is designed for a 2-axes antenna stabilization-tracking system and the performance of the controller is verified by simulations and real field experiments. A comparison between the performance of the conventional PID controller and the Fuzzy-PID controller, which is designed by the same PID control gains, is made as a way to verify the performance of the designed antenna servo control system. Furthermore, to verify the performance of the designed fuzzy controller for tracking antenna system, simulation block is designed and tested under several environmental disturbance conditions.

## 2 Antenna System

Stabilization is the process of de-coupling the vehicle's motion from the antenna. If we consider the dynamics of moving vehicles (such as ships and aircraft), the vehicles are affected by the desired motion as well as other disturbance motions. In this paper, we represented the 3-directional motion of a vehicle as longitudinal, lateral and vertical direction as X, Y and Z axis, respectively with respect to center of mass. The disturbance motion of vehicle is further obtained as the combination of rotation motion of each axis. The rotational motion with respect to X, Y, Z-axis is defined as Roll, Pitch, and Yaw (RPY). A conventional antenna tracking method, step-tracking algorithm, as illustrated in Figure 1, is used in this paper [6]. Generally, the step tracking operation begins when the antenna is commanded to make an initial one-step turn in any direction; right after the level of the receiving signal is compared with the previous level before the turn. If the signal level has increased, the antenna continues another one-step turn in the same direction. If the signal level has decreased, the turn of the antenna is reversed. Within these step-by-step turns, the receiver antenna can track the maximum point of the signal level.



**Fig. 1.** Step-Tracking Method

## 3 Controller Design

In this paper, a simplified fuzzy-PID controller is designed for the real-time implementation on an embedded tracking antenna control system. To simplify the fuzzy rule set, the control system consists of control block 1 and control block 2. The function diagram of the fuzzy system is shown in Figure 2. The inputs of control block 1 are error and error rate and, the inputs of Control block 2 are error rate and error acceleration. Figure 3 illustrated the input membership function for error, error rate, and error acceleration. EN, EP, RN, RP, AN and AP are error negative, error positive, error rate negative, error rate positive, error acceleration negative and error acceleration positive, respectively. By using these rule sets, the mathematical equation for this fuzzy rule can be generated and the simplified equation can be embedded in a real-time embedded controller.

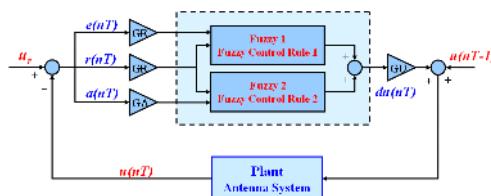
The Fuzzy control Rules for control block 1 are shown as:

- R1: IF Error is EP and Error Rate is RP, THEN Output is OP
- R2: IF Error is EP and Error Rate is RN, THEN Output is OZ
- R3: IF Error is EN and Error Rate is RP, THEN Output is OZ
- R4: IF Error is EN and Error Rate is RN, THEN Output is ON

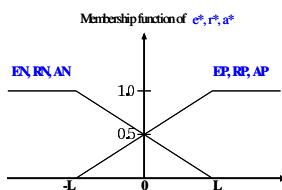
The Fuzzy control Rules for control block 2 are shown as:

- R1: IF Error Rate is RP and Error Acceleration is RP, THEN Output is OPM
- R2: IF Error Rate is RP and Error Acceleration is AN, THEN Output is ONM
- R3: IF Error Rate is RN and Error Acceleration is AP, THEN Output is OPM
- R4: IF Error Rate is RN and Error Acceleration is AN, THEN Output is ONM

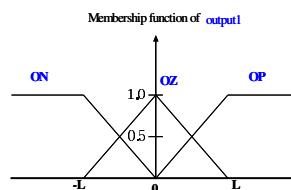
The output membership function of control block 1 is shown in Figure 4. ON, OZ and OP are output negative, output zero, and output positive, respectively. The output membership function of control block 2 is illustrated in Figure 5. OMN and OPM are output middle negative and output positive meddle, respectively.



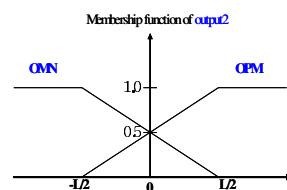
**Fig. 2.** The function Diagram of Fuzzy System



**Fig. 3.** Input MF



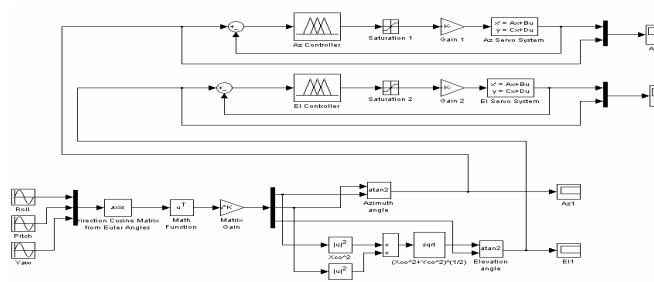
**Fig. 4.** Output MF for Block 1



**Fig. 5.** Output MF for Block 2

## 4 Simulations

The Simulation block diagram is shown as in Figure 6. The comparison with the conventional PID controller was made as a way to test the availability of the designed



**Fig. 6.** Simulation Block Diagram of Matlab and Simulink

fuzzy-PID controller. As a result, the conventional PID controller consists of fixed control gains and fuzzy-PID controller consists of varied PID control gains. The step response of fuzzy-PID control system and PID control system is shown in Figure 7. The response of tracking antenna system with sine environmental disturbances is further illustrated in Figure 8. As shown in Figure 8, the fuzzy-PID controller has better performance than conventional PID controller. Thus results are due to the variable gain of fuzzy-PID controller and early adaptation of error.

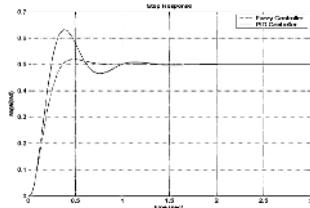


Fig. 7. The Step Responses

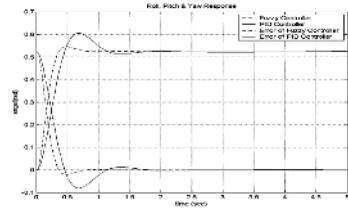
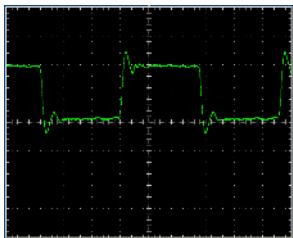


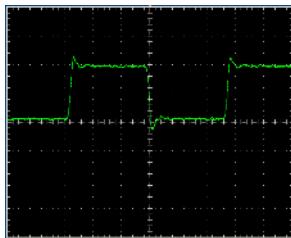
Fig. 8. The response with the Roll, Pitch and Yaw Motion

## 5 Experiment

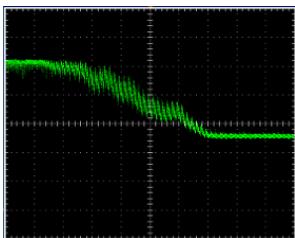
For tracking antenna system, the frame of SeaTel's 1898 satellite antenna was modified with flat antenna and BLDC motor which is suitable to data-link between moving vehicles was used. For main controller, a DSP controller (TMS320F2812, TI) was used. In order to develop an efficient antenna tracking control system, an unmanned helicopter, which is easily controlled by radio control facilities, was used as a target moving vehicle. The tracking antenna system can track the unmanned helicopter by sensing the data link signal level. The tuner selects a channel of the desired frequency from the received antenna signals with the AGC output proportional to the received signal level. The AGC output voltage usually lies between 1.2[V] and 2.8[V], which is filtered by a low-pass filter. The real-time controllers are developed using a DSP processor and a simplified Fuzzy-PID algorithm is implemented in this DSP controller. The step responses of two types of controller are further illustrated in Figure 9 and Figure 10. These resultant plots further emphasize the fact that Fuzzy-PID controller is superior to the conventional PID controller in point of view of overshoot and setting time. Figures 11 and Figure 12 show the variations of the AGC voltage according to variations of the azimuth angle. During the executions of the tracking, the maximum AGC gain was about 3.0 [V] and the minimum was about 1.3 [V]. The performance of Fuzzy-PID controller was better than that of conventional PID controller. In cases where PID controller was used, greater vibration in the mount was observed. The Fuzzy-PID controllers show better tracking performances than the PID controllers. Although this value depends on environmental conditions, the results of the physical experiments show that, with the AGC gain below 1.5 [V], data link beam can be satisfactorily received.



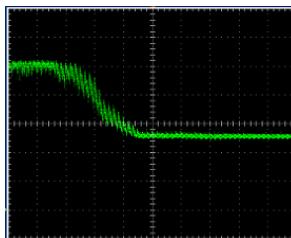
**Fig. 9.** Step Response of PID Controller



**Fig. 10.** Step Response of Fuzzy-PID Controller



**Fig. 11.** AGC Level of PID Controller



**Fig. 12.** AGC Level of Fuzzy-PID Controller

## 6 Conclusion

In this research, a stable tracking algorithm was tested by computer simulations for a 2-axes data link antenna system. By using coordinate transformation, the compensated angle calculation algorithm was derived and to a fuzzy-PID control algorithm was used to increase the performance of the antenna system. The performances of conventional PID controller and the fuzzy-PID controller were compared in both computer simulations and real field experiments. The conventional PID controller might have a better performance in the condition of exact gain tuning in one or two cases, while the fuzzy-PID controller, which proposed in this paper, is very simple and more robust than the ordinary PID controller in varying environmental conditions, which include mass change due to fuel consumption and wind speed change throughout experiments. In this paper, we can conclude that, the designed fuzzy-PID controller has superior performance than the conventional-PID controller, and it satisfied the performances of general purpose data link antenna system with respect to all cases of simulations and experiments.

## References

1. S.W. Park, J.H. Shin, B.J. Choi, Y.H. Kim, " Optimal Azimuth Shift for Mobile Direct Broadcasting Satellite Antenna ", Proceedings of the 13th KACC, October, 1998.
2. J. K. Kim, K. R. Cho, D. W. Lee and C. S. Jang, "A Study on the Tracking Antenna System for Satellite Communication Using Embedded Controller", ICCAS 2004, Bangkok Thailand, August, 2004.

3. N. Hirakoso, S. Matunaga, Y, Ohkami, "Experiment of Antenna Pointing Control for Satellite Communications Tracking Systems on Vehicles", The Japan Society of Mechanical Engineers (JSME), C-68, 667, March, 2002.
4. S. V. Subba B. Subramanyam, R. Nandakumar, "Program Mode of Antenna Position Control using Microcomputer", IE Journal, Vol 65, 1984.
5. Mohammad Jamshidi, Nader Vadiee, Timithy J, Ross, "Fuzzy Logic and Control: Software and Hardware Applications", Vol 2, Prentice Hall, 1993.
6. C.H, Ch, S.H, Lee, T. Y, Kwon, C. Lee, "Antenna control system using step tracking algorithm with  $H_\infty$  controller", Inter national journal of control, automation, and systems, Vol 1, No.1, March, 2003.

# Neuron Based Nonlinear PID Control

Ning Wang and Jinmei Yu

National Laboratory of Industrial Control Technology, Institute of Advanced Process Control,  
Zhejiang University, Hangzhou, 310027, P.R. China  
nwang@iipc.zju.edu.cn

**Abstract.** The neuron based nonlinear PID control method for plant with uncertainties is proposed in this paper. In this control system, the neuron based nonlinear PID controller is constructed by selecting the neuron inputs, and the control action of the nonlinear PID controller are determined by modifying the neuron weights on-line. The neuron based nonlinear PID controller produces the control signal in model-free way. With an example of the basis weight control of a paper machine, the experiments are made. The simulation results show that the neuron model-free controller has good control performance, fast transient response and strong robustness.

## 1 Introduction

PID controllers have been widely used in industry to solve various control problems due to their simple structure and relatively easy tuning controller parameters. However, it is difficult for us to use linear PID controllers to deal with plants with big uncertainties or highly nonlinear characteristics. Many design methods of nonlinear PID controllers are discussed. The simple way of designing nonlinear PID controllers is to use a nonlinear function. F. G. Shinskey described an error-square PID controller, in which the PID controller gain increases with the error by using a nonlinear function [1]. F. J. Jiang *et al.* defined a nonlinear function to design the nonlinear PID controller [2]. But those PID control methods can not control plants with big uncertainties efficiently. In recent years, the nonlinear PID controllers based on fuzzy sets or neural networks are investigated [3],[4]. In [5], N. Wang *et al.* developed the neuron model and its learning strategy for control, and proposed a neuron control method. Used in some industrial processes, this model-free control method has reached its success [6],[7]. In this paper, the neuron based nonlinear PID control method is proposed and the simulation results of applying the proposed control method to control the basis weight control of a paper machine are illustrated.

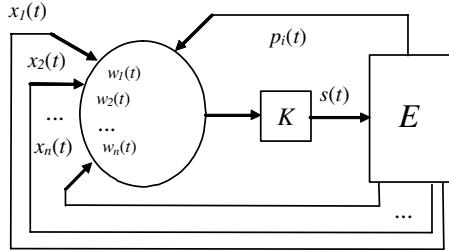
## 2 The Neuron Model and Learning Strategy for Control

In [5], the adaptive neuron model (in Fig.1) for model-free control is proposed by N.Wang *et al.* (1991). In Fig.1,  $E$  is the surroundings of the neuron.

The neuron output  $s(t)$  can be written as

$$s(t) = K \sum_{i=1}^n w_i(t) x_i(t), \quad (1)$$

where,  $K > 0$  is the neuron proportional coefficient;  $x_i(t)$  ( $i=1, 2, \dots, n$ ) denote the neuron inputs,  $w_i(t)$  are the connection weights of  $x_i(t)$  and they are determined by some learning rules. It is widely believed that a neuron self-organizes by modifying its synaptic weights.



**Fig. 1.** The neuron model for control

According to the well known hypothesis proposed by D. O. Hebb, the learning rule of a neuron is hence formulated as

$$w_i(t+1) = w_i(t) + d p_i(t), \quad (2)$$

where,  $d > 0$  is the learning rate;  $p_i(t)$  denote learning strategy. The associative learning strategy is suggested for control purposes as follows by N. Wang *et al.* [5].

$$p_i(t) = z(t)s(t)x_i(t). \quad (3)$$

It expresses that an adaptive neuron, which uses the learning way integrating Hebbian learning and Supervised learning, makes actions and reflections to the unknown outsides with the associative search. It means that the neuron self-organizes the surrounding information under supervising of the teacher's signal  $z(t)$  and gives the control signal. It also implies a critic on the neuron actions.

### 3 The Neuron Based Nonlinear PID Controller

#### 3.1 The Nonlinear PID Controller [2]

In [2], F.J. Jiang *et al.* presented a nonlinear PID controller as follows

$$u(t) = K_{NP} f(e, \alpha_P, \delta_P) + T_{NI} f(\dot{e}, \alpha_I, \delta_I) + T_{ND} f(\ddot{e}, \alpha_D, \delta_D), \quad (4)$$

where,  $K_{NP}$ ,  $T_{NI}$ ,  $T_{ND}$  are the three parameters of the nonlinear PID controller. And  $f(\cdot)$  is the nonlinear function defined as

$$f(x, \alpha, \delta) = \begin{cases} \text{sign}(x) \cdot |x|^\alpha & \text{when } |x| > \delta, \\ \delta^{\alpha-1} x & \text{when } |x| \leq \delta, \end{cases} \quad (5)$$

where,  $\alpha$ ,  $\beta$  are the parameters of the nonlinear function. Usually,  $\alpha$  is between 0 and 1 ( $0 < \alpha \leq 1$ ). When  $\alpha = 1$ , it is the linear function of  $f(x) = x$ .  $\delta$  is a small positive number applied to create a small linear area in this nonlinear function when  $x$  is around zero.  $e(t)$  is the error. The nonlinear function  $f(x, \alpha, \delta)$  gives high gain for small  $x$  and a small gain for large  $x$ .

### 3.2 The Neuron Controller [5]

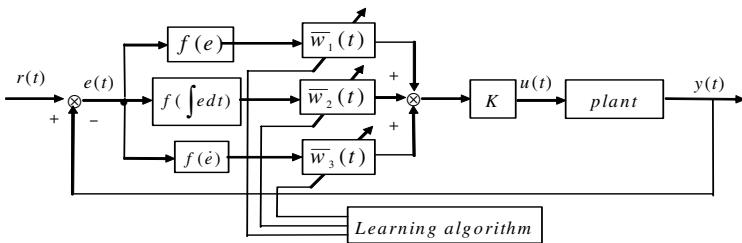
According to the neuron model and the learning strategy described as above, the neuron model-free control method is proposed as follows [5]

$$\begin{cases} u(t) = \frac{K \sum_{i=1}^n w_i(t) x_i(t)}{\sum_{i=1}^n w_i(t)}, \\ w_i(t+1) = w_i(t) + de(t)u(t)x_i(t). \end{cases} \quad (6)$$

where  $u(t)$ ,  $y(t)$  are the input and output of the plant respectively, and  $u(t)$  is the control signal produced by the neuron.  $e(t) = r(t) - y(t)$ ,  $r(t)$  is the set point, and the neuron inputs  $x_i(t)$  can be selected by the demands for the control system designs.

### 3.3 The Neuron Based Nonlinear PID Control Method

The neuron is used to construct the nonlinear PID controller by combining the advantages of the neuron controller and the conventional nonlinear PID controller. The neuron based nonlinear PID control system is set up and shown in Fig.2.



**Fig. 2.** The neuron based nonlinear PID control system

In this control system, the neuron based nonlinear PID controller is constructed by selecting the neuron inputs, and the control action of the nonlinear PID controller are determined by modifying the neuron weights on-line. The neuron based nonlinear PID

controller produce the control signal in model-free way. It is designed as follows, the inputs are selected as

$$x_1(t) = f(e(t), \alpha_P, \delta_P), \quad x_2(t) = f(\int e(t)dt, \alpha_I, \delta_I), \quad x_3(t) = f(\dot{e}(t), \alpha_D, \delta_D). \quad (7)$$

where, the nonlinear function  $f(\cdot)$  is defined as Eq.(5),  $e(t) = r(t) - y(t)$ . From Eqs.(6) and (7), we have

$$\begin{aligned} u(t) &= K\bar{w}_1(t)x_1(t) + K\bar{w}_2(t)x_2(t) + K\bar{w}_3(t)x_3(t) \\ &= K_P(t)f(e(t), \alpha_P, \delta_P) + K_I(t)f(\int e(t)dt, \alpha_I, \delta_I) + K_D(t)f(\dot{e}(t), \alpha_D, \delta_D). \end{aligned} \quad (8)$$

Where,  $\bar{w}_i(t) = w_i(t)/\sum_{i=1}^3 w_i(t)$ ,  $K_P(t) = K\bar{w}_1(t)$ ,  $K_I(t) = K\bar{w}_2(t)$ ,  $K_D(t) = K\bar{w}_3(t)$ .

Thus, the Eqs.(6) and (7) construct the neuron based nonlinear PID controller. According to the description as above, the neuron based nonlinear PID control is proposed as follows

$$\left\{ \begin{array}{l} u(t) = \frac{K \sum_{i=1}^3 w_i(t)x_i(t)}{\sum_{i=1}^3 w_i(t)}, \\ w_i(t+1) = w_i(t) + d_i e(t)u(t)x_i(t), \\ x_1(t) = f(e(t), \alpha_P, \delta_P), \quad x_2(t) = f(\int e(t)dt, \alpha_I, \delta_I), \\ x_3(t) = f(\dot{e}(t), \alpha_D, \delta_D). \end{array} \right. \quad (9)$$

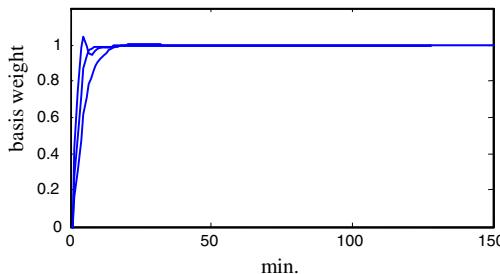
where,  $K$ ,  $d_i$  ( $i=1,2,3$ ),  $\alpha_P, \delta_P, \alpha_I, \delta_I, \alpha_D, \delta_D$  are the constants to be determined.

## 4 Simulation Tests and Results

The plant with big uncertainties of a paper-making process is taken as an example to verify the proposed nonlinear PID control method in Eq.(5). The dynamic characteristics of the plant can be written as follows [8]

$$G(z) = \begin{cases} \frac{0.2719}{z(z-0.8187)} & \text{when } 80 \text{ g/m}^2 \text{ paper is made,} \\ \frac{0.4484}{z(z-0.7788)} & \text{when } 100 \text{ g/m}^2 \text{ paper is made,} \\ \frac{0.7087}{z(z-0.7165)} & \text{when } 120 \text{ g/m}^2 \text{ paper is made.} \end{cases} \quad (10)$$

The experiments of using the proposed nonlinear PID controller are made. the controller parameters are selected as follows:  $K = 1.1$ ,  $d_1 = 15$ ,  $d_2 = 1$ ,  $d_3 = 10$ ,  $\alpha_P = 1$ ,  $\alpha_I = 0.5$ ,  $\alpha_D = 0.5$ ,  $\delta_P = 0.1$ ,  $\delta_I = 0.4$ ,  $\delta_D = 0.3$ . The simulation results are shown as in Fig.3. From the simulation results, it is known that the response curves of using the neuron based nonlinear PID controller are close to the same shape.



**Fig. 3.** The simulation results of the proposed controller

## 5 Conclusions

This paper work illustrates that the neuron based nonlinear PID controller can efficiently control a plant with big uncertainties. The main advantages of this model-free control method have strong adaptability and robustness. It shows good performance of the control system when different kinds of paper are made.

**Acknowledgments.** This paper is supported by National Science Foundation of China under grant 60421002 and 70471052.

## References

1. F. G. Shinskey.: Process control systems: application, design and adjustment. McGraw-Hall Book Company, New York (1988)
2. F.J. Jiang., *et al.*: An application of nonlinear PID control to a class of truck ABS problems. Proc. of the 40th IEEE Conference on Decision and Control, Orlando (2001) 516-521
3. A.A.S. Ibrahim.: Nonlinear PID controller design using fuzzy logic. Proc. of the 11th Mediterranean Electrotechnical Conference, Menouf (2002) 595-599
4. C.L. Chen., F.Y. Chang.: Design and analysis of neural/fuzzy variable structure PID control systems. IEE Proc. Control and Application, Vol.143, No.2 (1996) 200-208
5. N. Wang., J. Tu., J. Chen.: Intelligent Control Using the Single Adaptive Neuron. J. Hangzhong Univ. of Sci. & Tech, Vol.21, no.3 (1993) 31-35
6. N. Wang., J.Tu., J. Chen.: Neuron intelligence control for electroslag remelting processes. ACTA Automatica SINICA, vol.19, no.5 (1993) 634-636
7. N. Wang., Chen J., Wang J. C.: Neuron intelligent control for hydraulic turbine generators. Proceedings of the IEEE International Conference on Industrial Technology (1994) 288-292
8. Q.G. Wang.,*et al.*: A multiple-model based design method for the control of uncertain multivariable systems with paper machines. ACTA Automatica SINICA, vol.17, no.1 (1991) 68-76

# An Image Retrieval System Based on Colors and Shapes of Objects

Kuo-Lung Hong<sup>1</sup>, Yung-Fu Chen<sup>2,\*</sup>, Yung-Kuan Chan<sup>3</sup>, and Chung-Chuan Cheng<sup>1</sup>

<sup>1</sup> Department of MIS, Chaoyang University of Technology, Taichung 413, Taiwan  
`{klhung, s9314638}@cyut.edu.tw`

<sup>2</sup> Department of CSIE, National Chin-Yi Institute of Technology, Taichung County 411, Taiwan  
`yfchen@ncit.edu.tw`

<sup>3</sup> Department of MIS, National Chung Hsing University, Taichung 402, Taiwan  
`ykchan@nchu.edu.tw`

**Abstract.** This paper proposes a color-shape based method (CSBM) based on color, area, and perimeter intercepted lengths of segmented objects in an image. It characterizes the shape of an object by the intercepted lengths obtained by intercepting the object perimeter by eight lines with different orientations passing through the object center. The experimental results show that CSBM provides a better performance than fuzzy color histogram (FCH) and conventional color histogram (CCH). Besides, it is insensitive to translation, rotation, distortion, scaling, and hue variations, but impressionable to contrast and noise variations.

**Keywords:** Content-based image retrieval, Perimeter intercepted length, Fuzzy c-mean histogram (FCH), Conventional color histogram (CCH).

## 1 Introduction

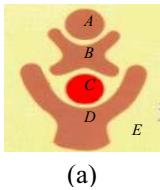
Context-based image retrieval has been studied for more than two decades, which generally works with features including color, texture, and shape [1]. Color histogram is one of the most commonly adopted features for designing image retrieval systems [2]. The advantages of color histogram are simple in operation and easy for calculation. In order to improve the efficacy of the conventional color histogram (CCH), a fuzzy-based technique, namely fuzzy color histogram (FCH) [3], is proposed for color histogram construction. It considers the degree of color similarity for each pixel to be associated with all the histogram bins using fuzzy memberships, in which the memberships are calculated based on the fuzzy c-means algorithm [3]. It is demonstrated to be more robust than the CCH in dealing with quantization errors and changes in light intensity. CCH and FCH, however, embed a significant drawback that they can delineate only the global properties of an image. To overcome this problem, this paper proposes a color-shape based method (CSBM) based on the colors, areas, and intercepted lengths obtained by intercepting the object perimeter by eight lines with different orientations passing through the object center for segmented objects. The intercepted lengths are demonstrated to be effective in discriminating shapes of the objects and are immune to translation and rotation variations.

---

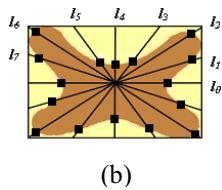
\* Corresponding author.

## 2 Color-Shape Based Method

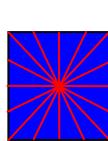
If the possible pixel colors in a color image, for example trademark, cartoon, flag, traffic sign, and synthesized images, are reduced to fewer colors, it usually turns into an image consisting of several large regions that each is made up of a set of pixels having the same color. Since it is very difficult to segment the objects contained in an image, this paper regards a region consisting pixels with identical color as an object. Two similar images generally contain several bigger objects that have similar colors and shapes. Figure 1(a) shows an image containing five objects, in which *A* and *B* are usually treated as different objects because of different shapes even with identical color. Objects *A* and *C* are also different because of distinctive colors although having the same shapes and sizes.



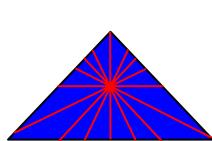
(a)



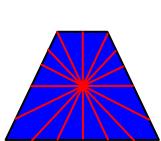
(b)



(a)



(b)



(c)

**Fig. 1.** (a) An image and (b) the 8 perimeter intercepts of an object bounded by an MBR

**Fig. 2.** Objects having the same area but different shapes (PILs)

### 2.1 Feature Extraction and Image Matching

CSBM first classifies the pixels of images into  $K$  clusters using the  $K$ -mean clustering algorithm [4]. The mean value of each cluster is regarded as a representative color in a color palette, namely common color palette (*CCP*), for all images including database and query images. For an image  $I$ , its color-reduced image  $I'$  containing only  $K$  colors is generated by replacing each pixel in  $I$  with a closest color found in *CCP*. Every object has its distinctive color, area, and shape. For an object  $O$  with an area greater than a threshold value  $Th_A$ , features including color, area, and perimeter intercepted lengths (PILs) are calculated and recorded; otherwise, it is treated as a noise and ignored. As shown in Fig. 1(b), a minimal bounding rectangle (MBR) with its sides parallel to the X and Y axes is applied to enclose the object. The area is featured as the number of pixels within  $O$ , whereas the shape as a set of 8 perimeter intersected lengths obtained by 8 lines passing through the central pixel of the MBR with different orientations intersecting the perimeter of object  $O$ . The orientations of these eight lines are separated by  $22.5^\circ$  from  $0^\circ$  to  $180^\circ$ . For object  $O$ , CSBM records its color, area, and PILs into a database. The difference between the PILs of two objects is demonstrated to be a good indicator of shape variation. As shown in Fig. 2, although the areas of all three objects are very close, their shapes are greatly different. To achieve rotation invariant, two objects are compared in 8 different orientations that the smallest Euclidean distance is used as an indicator for shape differentiation.

Let  $I$  be an image with the size of  $H \times W$  pixels. To remedy the problem caused by scale variation, this paper normalizes the features by dividing each value of PILs by

( $H \times W$ ) and the object area by ( $H \times W$ ), respectively. Let  $O_{hi}^q$  ( $1 \leq i \leq n_h^q$ ) and  $O_{hj}^d$  ( $1 \leq j \leq n_h^d$ ) be the objects with the  $h$ -th color in CCP obtained from  $I'_q$  and  $I'_d$ , color-reduced images of the query image  $I_q$  and database image  $I_d$ , respectively. Also,  $n_h^q$  and  $n_h^d$  denote the total numbers of objects in  $I'_q$  and  $I'_d$  with  $h$ -th color in CCP. CSBM calculates the distance  $d_{ij}$  between objects  $O_{hi}^q$  and  $O_{hj}^d$  according to Eq. (1):

$$d_{ij} = d_{ij}^L + w \times d_{ij}^A \quad (1)$$

in which  $w$  is a weighted constant, whereas  $d_{ij}^L$  and  $d_{ij}^A$  are the distances for the PIL and area, respectively, between  $O_{hi}^q$  and  $O_{hj}^d$ . With PILs denoted as  $L^q = \{l_0^q, l_1^q, \dots, l_7^q\}$  and  $L^d = \{l_0^d, l_1^d, \dots, l_7^d\}$  and areas as  $A_i^q$  and  $A_j^d$  for  $O_{hi}^q$  and  $O_{hj}^d$ ,  $d_{ij}^L$  and  $d_{ij}^A$  in Eq. (1) can be calculated from the following equation with  $z$  is a given constant.

$$d_{ij}^L = \sqrt{z \min \left( \sum_{r=0}^7 \sum_{s=0}^7 |l_s^q - l_{(r+s) \bmod 8}^d|^z \right)} \quad \text{and} \quad d_{ij}^A = |A_i^q - A_j^d| \quad (2)$$

### Algorithm OBJ\_Match()

```

Input:  $O_{h1}^q, O_{h2}^q, \dots, O_{hn_h^q}^q$  and  $O_{h1}^d, O_{h2}^d, \dots, O_{hn_h^d}^d$ ,
Output:  $MMD^h$ 
 $MMD_{0,0}^h = 0$ 
for  $i = 0$  to  $n_h^q$ 
     $MMD_{i,0}^h = \text{Penalty}(O_{hi}^q) + MMD_{i-1,0}^h$ 
for  $j = 0$  to  $n_h^d$ 
     $MMD_{0,j}^h = \text{Penalty}(O_{hj}^d) + MMD_{0,j-1}^h$ 
for  $i = 1$  to  $n_h^q$ 
    for  $j = 1$  to  $n_h^d$ 
         $MMD_{i,j}^h = \min(MMD_{i-1,j}^h + \text{Penalty}(O_{hi}^q), MMD_{i,j-1}^h + \text{Penalty}(O_{hj}^d), MMD_{i-1,j-1}^h + d_{i,j})$ 

```

CSBM adopts the dynamic programming method, as described in **Algorithm OBJ\_Match()**, to calculate the minimal matching distance between two sets of objects  $O_h^q$  and  $O_h^d$ , in which the objects are sorted by area in descending order. In the algorithm, a two-dimensional  $n_h^q \times n_h^d$  matrix  $MMD^h$  is used to record the minimal matching distance between  $O_h^q$  and  $O_h^d$ . An element in the matrix is represented as  $MMD^h(i, j)$  which indicates the minimal matching distance between any one object in  $\{O_{h1}^q, O_{h2}^q, \dots, O_{hi}^q\}$  and any one in  $\{O_{h1}^d, O_{h2}^d, \dots, O_{hj}^d\}$ ; therefore, the element  $MMD^h(n_h^q, n_h^d)$  denotes the final minimal matching distance between  $O_h^q$  and  $O_h^d$ . For an object  $O_{hm}$  in either a query with  $1 \leq m \leq n_h^q$  or a database image with  $1 \leq m \leq n_h^d$  which does not have its corresponding matched object in its counterpart, a penalty is given according to the equation defined in Eq. (3) with  $A_{hm}$  indicating the area of  $O_{hm}$ .

$$\text{Penalty}(O_{hm}) = \sqrt{z \sum_{r=0}^7 (l_r)^z} + A_{hm} \times w \quad (3)$$

Finally, the image matching distance  $Dist$  between  $I_q$  and  $I_d$  is defined as  
 $Dist = \sum_{h=1}^K MMD^h(n_h^q, n_h^d)$ .

## 2.2 Experimental Images and System Evaluation

In the experiment, an image set consists of 100 true-color images are used as the query images. Additionally, a database containing images with variations, including rotation, distortion, noise, scaling, hue, luminance, and contrast, are generated in Adobe Photoshop 7.0 package from each query image is constructed. In Table 1, examples of query images and their geometry (rotation, scaling, and translation), color (luminance, contrast, and hue), noise, blurring, and distortion variations are demonstrated for testing the performance of three methods.

**Table 1.** Examples of query images and their corresponding images with variations

Query	Trans. & Rot.	Lumin.	Contrast	Noise	Distort.	Scaling	Hue

To evaluate the system performance, average normalized modified retrieval rank ( $ANMRR$ ), as shown in Eq. (4), proposed by MPEG-7 [5] is used as a benchmark. It not only reflects the recall rate and precision information of the retrieved images, but also shows the ranks of all the retrieved images.

$$ANMRR = \frac{Q}{T + 0.5 - 0.5 \times I(q)} \quad \text{with} \quad AVR(q) = \sum_{t=1}^{I(q)} R(t) / I(q) \quad (4)$$

in which  $T$  is  $\text{Min}\{4 \times I(q), 2 \times GTM\}$  with  $GTM$  representing  $\text{Max}\{I(q)\}$  for each query  $q$ . Also  $I(q)$  indicates the number of returned images which are most similar to the query image and  $Q$  indicates total number of queries for evaluation. Notice that smaller  $ANMRR$  value indicates better retrieval performance.

## 3 Experimental Results

Since CSBM can achieve the best performance if the parameters,  $w$ ,  $z$ ,  $K$  and  $Th_A$ , have been set to 0.009, 1.1, 27 and 5, respectively, hence the parameters are set accordingly for the experiment. Table 2 compares the performances among CSBM, FCH, and CCH. The results show that CSBM is immune to geometry, distortion, and hue variations; slightly susceptible to luminance variation; and very sensitive to contrast and noise variations. FCH and CCH, on the other hand, can resist scale variation, slightly to noise and distortion variations, and very susceptible to luminance and

contrast variations. Moreover, FCH is very vulnerable to geometric variations and CCH to hue variation. The average *ANMRR* for CSBM (0.147) is smaller than FCH (0.206) and CCH (0.231), which indicates that it provides better retrieval performance.

**Table 2.** Comparisons among CSBM, FCH, and CCH retrieval performance ( $T=2$ )

<b><i>ANMRR</i> (Rank 1, Rank 2)</b>	<b>Tran &amp; Rot</b>	<b>Lumi- nance</b>	<b>Con- trast</b>	<b>Noise</b>	<b>Dis- tortion</b>	<b>Scaling</b>	<b>Hue</b>	<b>Mean</b>
<b>CSBM</b>	0.000 (100,100)	0.145 (80,87)	0.435 (53,60)	0.320 (63,73)	0.030 (96,98)	0.075 (89,96)	0.025 (97,98)	0.147
<b>FCH</b>	0.090 (86,96)	0.535 (43,50)	0.330 (61,73)	0.190 (80,82)	0.145 (83,88)	0.000 (100,100)	0.150 (83,87)	0.206
<b>CCH</b>	0.095 (84,97)	0.455 (51,58)	0.405 (54,65)	0.205 (76,83)	0.110 (86,92)	0.065 (89,98)	0.280 (67,77)	0.231

## 4 Discussions and Conclusions

The method adopts the colors, areas, and PILs of segmented objects to describe the characteristics of an image. The PILs of an object can be used to effectively characterize the shape of the object. Since the same objects with different luminance or color contrasts might be regarded as different objects, all the three methods cannot achieve good performance for these variations. Noises may generate various small isolated objects which will affect the minimal matching distance between two groups of objects having identical color; CSBM is hence very sensitive to noise variation. In conclusion, CSBM is more robust in resisting geometric, luminance, distortion, and hue variations, and is more susceptible to noise variation than the FCH and CCH. It provides better retrieval performance than the other two methods based on mean *ANMRR*.

**Acknowledgment.** This work was funded in part by National Science Council of Taiwan under Grant NSC93-2213-E-212-038 for Y. F. Chen.

## References

1. Smeulders, A. W. M., Worring, M., Santini, S., Gupta, A., and Jain, R.: Content-Based Image Retrieval at the End of the Early Years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 22 (2000) 1349-1380.
2. Swain, M. J. and Ballard, D. H.: Color indexing. *International Journal of Computer Vision*, Vol. 7, (1991) 11-32.
3. Han, J. and Ma, K. K.: Fuzzy color histogram and its use in color image retrieval. *IEEE Transactions on Image Processing*, Vol. 11 (2002) 944-952.
4. Su, M. C. and Chou, C. H.: A Modified Version of the K-means Algorithm with a Distance Based on Cluster Symmetry. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (2001) 674-680.
5. Manjunath, B. S., Ohm, J. R., Vasudevan, V. V., and Yamada, A.: Color and texture descriptors. *IEEE Transaction On Circuits and Systems for Video Technology*, Vol. 11, (2001) 703-715.

# A Hybrid Mood Classification Approach for Blog Text

Yuchul Jung, Hogun Park, and Sung Hyon Myaeng\*

School of Engineering, Information and Communications University, South Korea  
119, Munjiryo, Yuseong-gu, Daejeon, 305-732, Korea  
{enthusia77, gsgphg, myaeng}@icu.ac.kr

**Abstract.** As an effort to detect the mood of a blog, regardless of the length and writing style, we propose a hybrid approach to detecting blog text's mood, which incorporates commonsense knowledge obtained from the general public (ConceptNet) and the Affective Norms English Words (ANEW) list. Our approach picks up blog text's unique features and compute simple statistics such as term frequency, n-gram, and point-wise mutual information (PMI) for the SVM classification method. In addition, to catch mood transitions in a given blog text, we developed a paragraph-level segmentation based on a mood flow analysis using a revised version of the GuessMood operation of ConceptNet and an ANEW-based affective sensing module. For evaluation, a mood corpus comprised of real blog texts has been built semi-automatically. Our experiments using the corpus show meaningful results for 4 mood types: happy, sad, angry, and fear.

## 1 Introduction

A blog is a web site, where anybody can write about his or her own personal experiences and thoughts on a voluntary basis. As a result, it reflects user's personality and cultural biases, sometimes forming a unique society. Since blog texts often carry the emotions of the writers, they should lend themselves for automatic categorization based on moods. Compared to topicality-based classification of text, mood classification is challenging in many aspects.

A recent approach to mood classification of text uses Support Vector Machine (SVM) with 6 features: frequency counts, lengths, semantic orientations, Point-wise Mutual Information for Information Retrieval (PMI-IR), emphasized words, and special symbols [1]. While this approach of using surface level features can allow for reasonable accuracy, it seems to have a limit because its inability to deal with idiosyncratic aspects of moods and blogs. For example, although an author is under a certain mood when starting to write a blog document, the initial mood may not be maintained all the way to the end. Some blogs are so intertwined that even human readers would have difficulty in identifying the mood, not to mention the statistically motivated method using surface level features.

To detect the mood of blog text more accurately, we propose a hybrid approach to mood classification that incorporates commonsense knowledge obtained from the

---

\* Corresponding author.

general public (ConceptNet) [2] and the Affective Norms English Words (ANEW) list [3]. ConceptNet is an easily usable, freely available commonsense knowledge base and natural language processing toolkit which support many practical textual-reasoning tasks over real-world documents, including topic-gisting, affect-sensing, analogy-making, and other context-oriented inferences. The knowledge base is a semantic network presently consisting of over 1.6 million assertions of commonsense knowledge covering the spatial, physical, social, temporal, and psychological aspects of everyday life.

The ANEW list, created from a psychological study, contains 1,034 unique terms with affective valence (unpleasant ~ pleasant), arousal (calm ~ excited), and dominance (submissive ~ dominance) scores. It can be used to identify different mood types based on lexical analysis by mapping terms in text to those in the list.

Our approach is hybrid in the sense that several tools are integrated: the SVM classification model [4] that has shown superior performance over other existing classification models in many application domains, the GuessMood function of ConceptNet [2], and an affective sensing model based on ANEW [3], and Open Mind Common Sense (OMCS)<sup>1</sup>[5]. We observed that some features like term frequency, PMI-IR, emoticons, abbreviated words, and mood-specific terms contribute to detection of the mood of a given text. In addition, a paragraph level segmentation and a mood flow analysis were applied to handle various blog texts of different lengths and writing style.

For an evaluation of our hybrid mood classification approach, we have built a mood corpus based on a large number of blog documents extracted from Live Journal.com. More than 50GB text has been processed to semi-automatically classify the documents into four categories: *happy*, *sad*, *angry*, and *fear*.

## 2 Proposed System

Our system includes two steps as in Fig. 1. In the first step, when a blog document comes in, the system performs statistical analyses to obtain term frequency, n-gram, and PMI-IR[7] sequentially. Based on the statistical features, the system applies SVM<sup>2</sup> based mood classification to assign a mood category to the document.

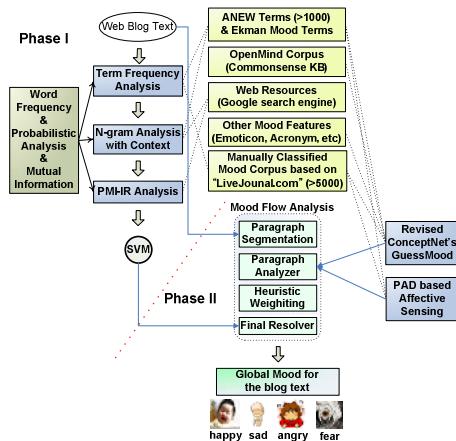
In the second step, the system initiates a mood flow analysis to identify a global mood for the given blog document. As in the details found in Fig. 2, our mood flow analyzer segments a blog document into paragraphs. After that, in paragraph analysis, the number of mood terms is counted to select a scheme between a revised ConceptNet's GuessMood [2] and a PAD [3] based affective sensing module. If the number of mood terms is bigger than the experimental threshold, the latter is chosen. If a mood is sustained without transitions throughout the whole blog document, the “final resolver” module only checks the consistency and assigns the mood as the final result. When some paragraphs have different mood types, heuristically measured weights are multiplied into the results of paragraph analysis. A global mood score is

---

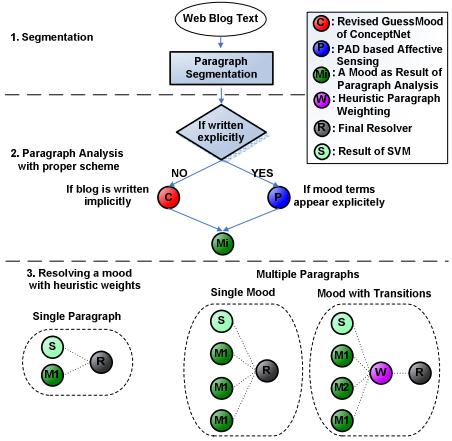
<sup>1</sup> <http://commonsense.media.mit.edu/cgi-bin/search.cgi>

<sup>2</sup> <http://svmlight.joachims.org/>

calculated by averaging the weighted sum of every paragraph analysis in the final resolution phase. The heuristic weights were obtained through several hundreds of trials with our training corpus (Fig. 2).



**Fig. 1.** Overall Flow



**Fig. 2.** Details of Mood Flow Analysis

### 3 A Blog Mood Corpus

We have collected over 50GB of blog text from “LiveJournal.com” semi-automatically, which are appropriate for our mood categories (happy, sad, angry, and fear) and the ANEW list to build a trustable mood corpus that can be used for training/testing purposes. In order to select blogs which fall into our mood categories, we adopted LiveJournal’s mood hierarchy<sup>3</sup>. In this process, 108,892 blog documents were extracted from 50G blog text. Because the authors’ mood “annotation” sometimes doesn’t match the real content, we performed a specially designed refinement process for more trustable mood corpus. It uses a keyword spotting technique to remove non-affective parts of the text, which contain few or no mood key terms that were predefined. In addition, to avoid irregular, meaningless, and ambiguously long blog documents, it ignores blog texts whose length is less than 5 or more than 40 sentences. Finally, now we have 10,479 blog documents (about 10MB) as a highly refined mood corpus.

### 4 Experiments and Discussion

For improved accuracy of mood classification, diverse features of blog documents were considered. In addition, each of the two methods, one with GuessMood of ConceptNet and the other with the PAD based affective text sensing model. Internal

<sup>3</sup> <http://www.livejournal.com/moodlist.bml?moodtheme=140&mode=tree>

parameters were tuned based on experimental work. The main goal of the experiments was to evaluate the hybrid mood classification model. In case of the *happy* and *sad* mood categories, classification results approached to the range of 89~92%. Table 2 shows classification accuracy comparisons between the baseline SVM and our proposed approaches. A total of randomly chosen 20,000 blog documents (*happy*: 5,000, *sad*: 5,000, *angry*: 5,000, and *fear*: 5,000) and semi-automatically refined 5,000 blog documents (*happy*: 2000, *sad*: 1800, *angry*: 600, *fear*: 600) were used for training and testing, respectively. 5-fold cross validation was taken for SVM classifier's evaluation. In Table 2, "random" and "refined" mean the corpus with 20,000 documents and that with 5,000 documents, respectively.

**Table 2.** Mood classification: SVM vs. Our Approach

Testing Type	Category	SVM	Our Approach
Type 1. Training: Random + Testing: Random	Happy	59.30	55.14 (-4.16)
	Sad	35.60	31.94 (-3.66)
	Angry	44.80	38.28 (-6.52)
	Fear	42.10	35.12 (-6.98)
		Average	45.45%
Type 2. Training: Refined + Testing: Random	Happy	77.80	76.79 (+9.99)
	Sad	43.15	38.37 (-4.78)
	Angry	26.67	23.25 (-3.42)
	Fear	27.23	25.40 (-1.83)
		Average	43.71%
Type 3. Training: Refined + Testing: Refined	Happy	90.13	92.85 (+2.72)
	Sad	89.58	89.27 (-0.31)
	Angry	80.77	83.12 (+2.35)
	Fear	57.68	61.97 (+4.29)
		Average	79.54%
			81.80%

Although the semantic network of ConcpetNet consists of 1.6million assertions, it contains lots of needless commonsense knowledge that is not required for processing mood related concepts. Thus, we have re-organized the semantic network by filtering out unnecessary concepts, forming the "refined" corpus.

**SVM:** Randomly selected training data were not sufficient in constructing a classifier; with the limited coverage, its performance was only 45.45% on average. When the SVM classifier was tested with randomly selected testing data, which contains inconsistent lexical features, it almost failed in getting a reasonable level of accuracy except for the *happy* mood. In case of Type 2, although highly refined training data were used as the training corpus, classification result became worse due to heterogeneity between the training and testing corpora. However, highly refined training data helped in achieving average 79.54% of accuracy if well refined testing data were also used.

**Our Hybrid Approach:** In every testing, classification accuracy for the *happy* mood was enhanced when the refined corpus was used in training. However, in case of Type 1&2, when randomly chosen testing data were used, there were ups and downs. On the other hand, our hybrid approach obtained 81.80% of accuracy on average when well refined training and testing data were used. While the revised GuessMood and the PAD based affective text sensing modules caused noisy classification results because the semantic network is not comprehensive enough to cover all the mood-related terms that appear in the blog text, the experimental result with the refined data indicates the resulting is promising. Even the relatively small size of training corpus of refined data allowed for quite reasonable performance.

## 5 Conclusion and Future Work

This paper presents a hybrid model for mood classification of blog text, which uses statistical features, an informal commonsense reasoning with ConceptNet, and a PAD based affective text sensing method. In addition, a semi-automatically refined mood corpus has been built and used to evaluate our proposed model.

Mood classification of blog documents is a very difficult task because of diverse situations and expressions of authors. Although we can hardly catch author's internal, emotional status correctly, at least we could perceive a global mood for a given blog text at the surface level if statistical features and commonsense knowledge are incorporated. With the experiments using the blog text, we have developed a firm belief that sophisticated preprocessing and a specially designed hybrid mood classifier are quite feasible for mood classification of blog text.

## References

1. Mishne, G.: Experiments with Mood Classification in Blog Posts. *Style2005 – the 1<sup>st</sup> Workshops on Stylistic Analysis of Text for Information Access, at ACM SIGIR 2005*, (2005)
2. Liu, H. and Singh, P.: ConceptNet – A practical commonsense reasoning tool-kit. *BT Technology Journal*, (2004), 211-226.
3. Bradley, M.M., and Lang, P.J.: Affective norms for English words (ANEW). Gainesville, FL. The NIMH Center for the Study of Emotion and Attention, University of Florida, (1999).
4. Joachims, T.: Text Categorization with Support Vector Machines: Learning with Many Relevant Features. *Proc. of ECML-98, 10th European Conf. on Machine Learning*, Springer-Verlag, (1998), 137-142
5. Singh, P., Lin, T., Mueller, E. T., Lim, G., Perkins, T., and Li Zhu, W.: Open Mind Common Sense: Knowledge acquisition from the general public. *Proc. of the First Int. Conf. on Ontologies, Databases, and Applications of Semantics for Large Scale Information Systems*, (2002).
6. Liu, H., Lieberman, H., and Selker.: A Model of Textual Affect Sensing using Real-World Knowledge. *Proc. of the 2003 Int. Conf. on Intelligent User Interfaces, IUI 2003*, (2003).
7. Read, J.: Recognising affect in text using pointwise-mutual information. Master's thesis, University of Sussex, (2004).
8. Mehrabian, A.: Framework for a comprehensive description and measurement of emotional states. *Genetic, Social, and General Psychology Monographs*, 121(3), (1995), 339-361.

# Modeling and Classification of Audio Signals Using Gradient-Based Fuzzy C-Means Algorithm with a Mercer Kernel

Dong-Chul Park<sup>1</sup>, Chung Nguyen Tran<sup>1</sup>, Byung-Jae Min<sup>1</sup>, and Sancho Park<sup>2</sup>

<sup>1</sup> Dept. of Information Engineering, Myong Ji University, Korea  
`{parkd, tnchung, mbj2000}@mj.ac.kr`

<sup>2</sup> Davan Tech Co., Seongnam, Korea  
`sancho@davan.co.kr`

**Abstract.** In this paper, we propose a noble classification algorithm for content-based audio signal retrieval. The algorithm uses the Gradient-Based Fuzzy C-Means with a Mercer Kernel (GBFCM(MK)) to perform clustering of Gaussian Probability Density Function (GPDF) data of a Gaussian Mixture Model (GMM). The GBFCM(MK) algorithm incorporates a kernel method into the GBFCM to implicitly perform non-linear mapping of the input data into a high-dimensional feature space. Experiments and results for several audio data sets have shown that the GBFCM(MK)-based classification algorithm has accuracy improvements of 3.14%-7.49% over classification algorithms employing the traditional k-means and the Fuzzy C-Mean (FCM), respectively.

## 1 Introduction

In recent years, various automatic and computerized methods based on the content-based analysis of audio data have been proposed for audio data classification and retrieval. Essentially, content-based classification of audio data can be performed by using a pattern recognition approach with a consideration of two issues: feature extraction and classification of extracted features. Acoustical features, such as loudness, pitch, brightness, bandwidth, and harmony have been most widely used to discriminate speech and music signals [1]. To further specifically classify music signals, music-oriented features, such as timbral texture, rhythmic content, and pitch content have been proposed [2].

In order to model and classified extracted features, statistical model based on a Gaussian Mixture Model (GMM) are widely used because of its computational simplicity [1,2]. The GMM is estimated from audio data by considering audio data as mixtures of Gaussian Probability Density Function (GPDF) data. The GPDF data are presented by a mean vector and a covariance matrix, which are obtained by using a clustering algorithm to distill natural groupings of data from a large data set. For clustering, the k-means and Fuzzy C-Means (FCM) algorithms are widely used [3,4]. As an improvement of the FCM, Gradient Based Fuzzy C-Mean (GBFCM) [5] exploits the characteristics of the Kohonen's Self Organizing Map [6] to improve the speed and computational complexity

of the FCM. However, these algorithms lack the ability to deal with data in which boundaries among clusters are nonlinear, and this shortcoming leads to inefficiencies in forming the mixtures of the GMM.

In this paper, we propose a noble model that can efficiently classify audio signals by employing a Gradient Based Fuzzy C-Mean with a Mercer Kernel (GBFCM(MK)) for the clustering of GPDF data of GMMs. The GBFCM(MK) algorithm employs a kernel method [7] called the Mercer kernel to implicitly perform nonlinear mapping of the input data into a high-dimensional feature space. By doing so, complex nonlinear classification boundaries in the input space with the original dimensions can more likely be treated linearly in the expanded feature space [7]. By incorporating the kernel method into the GBFCM, the GBFCM(MK) has the ability in dealing with nonlinear boundary clusters in GPDF data. When applied to the clustering of GPDF data of GMMs, the GBFCM(MK) can form mixtures of GMMs more efficiently than the conventional k-means or the FCM.

The remaining of this paper is organized as follows: A brief review of the GBFCM is given in Section 2. Section 3 introduces the GBFCM(MK) algorithm. Section 4 presents our experiments and results for several audio data sets. The conclusions and closing remarks are presented in Section 5.

## 2 Gradient Based Fuzzy C-Means (GBFCM)

The GBFCM was first introduced in [5]. The algorithm attempts to improve the FCM algorithm by minimizing the objective function using one input data at a time instead of the entire input data. Given one data  $\mathbf{x}_i$  and  $c$  clusters with centers at  $\mathbf{v}_j$ , ( $j = 1, 2, \dots, c$ ), the objective function to be minimized is:

$$J_i = \mu_{1i}^2(\mathbf{v}_1 - \mathbf{x}_i)^2 + \mu_{2i}^2(\mathbf{v}_2 - \mathbf{x}_i)^2 + \dots + \mu_{ci}^2(\mathbf{v}_c - \mathbf{x}_i)^2 \quad (1)$$

with the following constraint:

$$\mu_{1i} + \mu_{2i} + \dots + \mu_{ci} = 1 \quad (2)$$

The group centers are updated as follows:

$$\mathbf{v}_{k+1} = \mathbf{v}_k - 2\eta\mu_{ki}^2(\mathbf{v}_k - \mathbf{x}_i) \quad (3)$$

where  $\eta$  is a learning constant and the membership grades are defined as:

$$\mu_{ki} = \frac{1}{\sum_{j=1}^c (\frac{d_i(\mathbf{x}_k)}{d_j(\mathbf{x}_k)})^2} \quad (4)$$

More detail about GBFCM can be found in [5].

### 3 Gradient Based Fuzzy C-Means with a Mercer Kernel

The objective function in the FCM or GBFCM with a kernel can be rewritten in feature space with the mapping function  $\Phi$ :

$$J_i^\Phi = \sum_{k=1}^c \mu_{ki}^2 \|\Phi(\mathbf{v}_k) - \Phi(\mathbf{x}_i)\|^2 \quad (5)$$

By using the kernel substitution [7] with the Gaussian kernel function, the objective function becomes:

$$J_i^\Phi = 2 \sum_{k=1}^c \mu_{ki}^2 (1 - K(\mathbf{v}_k, \mathbf{x}_i)) \quad (6)$$

In order to minimize the objective function with a kernel, we use the steepest gradient descent algorithm. The learning rule can be summarized as follows:

$$\Delta \mathbf{v}_k = \eta (\mathbf{v}_k - \mathbf{x}_i) = \eta \frac{\partial J_i^\Phi}{\partial \mathbf{v}_k} \quad (7)$$

In the case of the Gaussian kernel function, the objective function in Eq.(6) can be rewritten as:

$$J_i^\Phi = 2 \sum_{k=1}^c \mu_{ki}^2 (1 - e^{-\frac{\|\mathbf{v}_k - \mathbf{x}_i\|^2}{\sigma^2}}) \quad (8)$$

By substituting Eq.(8) into Eq.(7), the group centers are updated as follows:

$$\mathbf{v}_{k+1} = \mathbf{v}_k - \eta \mu_{ki}^2 \sigma^{-2} K(\mathbf{v}_k, \mathbf{x}_i) (\mathbf{v}_k - \mathbf{x}_i) \quad (9)$$

By solving optimization condition in Eq.(6) with respect to the constraint in Eq.(2) using the Lagrange multiplier, the membership grades are updated as follows:

$$\mu_{ki} = \frac{1}{\sum_{j=1}^c \left( \frac{1-K(\mathbf{v}_k, \mathbf{x}_i)}{1-K(\mathbf{v}_j, \mathbf{x}_i)} \right)} \quad (10)$$

More detail on GBFCM(MK) can be found in [8].

### 4 Experiments and Results

A data set of 2,100 audio signals was used for experiments, including speech data and music data. The speech data consisted of 300 excerpts for males and 300 excerpts for females. The music data comprised of 5 genres, including country, folk, jazz, hip-hop, and rock. Each signal was a 30s-long excerpt, totaling more than 17 hours of audio data. To extract features from audio signals, we used an open source framework, Marsyas, which is provided by Tzanetakis [2].

**Table 1.** Classification accuracy (%) of speech/music classifier using different algorithms and 3 code vectors

	Speech	Music	Overall
K-means	95	87	<b>91.0</b>
FCM	96	89	<b>92.5</b>
GBFCM(MK)	98	93	<b>95.5</b>

**Table 2.** Classification accuracy (%) of speech classifier (male/female) using different algorithms and 3 code vectors

	Male	Female	Overall
K-means	99	70	<b>84.5</b>
FCM	99	73	<b>86</b>
GBFCM(MK)	100	79	<b>89.5</b>

The audio signals were classified into a hierarchy of genres. First, the audio signals were classified into speech or music signals, using the speech/music classifiers. Furthermore, the speech signals were classified into male speech signals and female speech signals, while the music signals were classified into 5 genres, including country, folk, jazz, hip-hop, and rock.

Table 1 compares the classification accuracies of classification algorithms using the conventional k-means, the FCM, and the GBFCM(MK) for a speech/music classifier. The results were obtained by using 400 speech excerpts and 400 music excerpts for training. The remaining 200 speech excerpts from the speech data and 200 music excerpts are for testing. As can be seen from Table 1, the classification model using the GBFCM(MK) outperforms those using the conventional k-means and the FCM in every case.

After classifying audio signals into speech or music signals, the speech signals were further classified into male or female speech signals. A summary of classification accuracy for speech classifier using 3 code vectors is given in Table 2. As can be seen from Table 2, overall classification accuracies of 84.5%, 86%, and 89.5% were achieved by using the conventional k-means, the FCM, and the GMFCM(MK), respectively.

To classify music signals into 5 genres, 200 excerpts from each genre were used for training, while the remaining 100 excerpts from each genre were used for testing. The classification accuracy of the music classifier using 5 code vectors is given in Table 3. As can be seen from Table 3, overall classification accuracies of 64.2%, 65.2%, and 69.4% were achieved by using the conventional k-means, the FCM and the GMFCM(MK), respectively. Table 4 shows the confusion matrix, which describes classification results for each genre in details. One significant point that can be inferred from the confusion matrix is that hip-hop and rock can be well discriminated from the others while country is quite likely to be confused.

**Table 3.** Classification accuracy of different algorithms (unit:%), using 5 code vectors

	Country	Folk	Jazz	Hiphop	Rock	Overall
K-means	42	45	62	97	75	<b>64.2</b>
FCM	42	48	58	99	79	<b>65.2</b>
GBFCM(MK)	47	57	69	99	79	<b>69.4</b>

**Table 4.** Confusion matrix of audio genres, using 5 code vectors

	Country	Folk	Jazz	Hiphop	Rock	Accuracy
Country	<b>47</b>	23	9	9	12	<b>47%</b>
Folk	9	<b>57</b>	13	7	14	<b>57%</b>
Jazz	3	14	<b>65</b>	11	7	<b>65%</b>
Hiphop	0	0	0	<b>99</b>	1	<b>99%</b>
Rock	2	5	10	4	<b>79</b>	<b>79%</b>

## 5 Conclusions

In this paper, a new approach for modeling and classification of audio signals using the GBFCM(MK) algorithm for clustering of GPDF data of GMMs is proposed. The GBFCM(MK) algorithm was formulated by incorporating the kernel methods with the GBFCM to manage nonlinear separation boundaries among clusters in GPDF data. The GMM using the GBFCM(MK) for clustering of GPDFs was applied to the audio signal classification problem. Our experiments and results for several audio signal data sets shown that respective improvements of as much as 7.49% and 3.14% can be archived over the conventional k-means and the FCM.

## References

1. Lu, L., Zhang, H.J., Jiang, H.: Content Analysis for Audio Classification and Segmentation. *IEEE Trans. on Speech and Audio Processing* 10(7) (2002) 504-516
2. Tzanetakis, G., Cook, P.: Musical Genre Classification of Audio Signals. *IEEE Trans. Speech and Audio Processing* 10(5) (2002) 293-302
3. Hartigan, J.: Clustering Algorithms. New York, Wiley (1975)
4. Bezdek, J.C.: Pattern Recognition with Fuzzy Objective Function Algorithms. New York, Plenum (1981)
5. Park, D.C., Dagher,I.: Gradient Based Fuzzy c-means Algorithm. *IEEE Int. Conf. on Neural Networks*, Vol. 3, ICNN-94 (1994) 1626-1631
6. Kohonen,T.: The Self-Organizing Map. *Proc. IEEE*, Vol. 78 (1990) 1464-1480
7. Muller, K.-R., Mika, S., Ratsch, G., Tsuda, K., Scholkopf, B.,: An Introduction to Kernel-Based Learning Algorithms. *IEEE Transactions on Neural Networks* 12(2) (2001) 181-201
8. Park, D.C., Tran, C.N., Park, S.: Gradient Based Fuzzy C-Means Algorithm with a Mercer Kernel. *Lecture Notes in Computer Science (LNCS)*, Vol. 3971, Springer-Verlag, Berlin Heidelberg New York (2006) 1038-1043

# A Quick Rank Based on Web Structure

Hongbo Liu, Jiaxin Wang, Zehong Yang, and Yixu Song

State Key Lab of Intelligent Technology and System,  
Department of Computer Science and Technology, Tsinghua University,  
Beijing, P.R. China  
[1hb00@mails.tsinghua.edu.cn](mailto:1hb00@mails.tsinghua.edu.cn)

**Abstract.** Hyperlink structure of the Web provides valuable information for ranking query results and has been used in some famous search engines. The development of search engines such as personalized and topic-sensitive search intensifies the need of quick rank algorithms. In this paper, a link based rank called ExpRank is proposed. It can converge quickly and reserve the fundamental features of PageRank. Experimental results comparing with PageRank on real dataset are also discussed.

## 1 Introduction

Search engines are successful applications of web mining technology and play more and more crucial roles on the Internet. PageRank proposed by Sergey Brin and Larry Page [1] is one of the most important link-based ranking algorithms and becomes the basis of ranking system used by Google search engine. One advantage of link-based algorithm is that it is query independent and content independent. It can be computed offline using only the link structure and is used when users submit queries to the search engine. The simplicity, robustness and effectiveness of link-based ranking method have been witnessed with the great success of Google.

Since WWW contains billions of pages, it is necessary to accelerate the speed of ranking computation. Recent research on the personalized and topic-sensitive search [2,3] greatly intensified the need of faster ranking algorithm. Some accelerated methods [4,5] have been proposed to speed up the computation. However, these methods are limited by the intrinsic property of slow convergence of power method. In this paper, we proposed a new link-based ranking method called ExpRank which could dramatically accelerates the iterative computation and reserves the fundamental features of PageRank. In the next section, main idea and computation of ExpRank will be introduced. Some experimental results comparing with PageRank will be presented in section 3.

## 2 ExpRank and Its Computation

### 2.1 Matrix Representation and PageRank

Consider web pages as nodes and the hyperlinks as directed links, the hyperlink structure of WWW can be represented as a directed graph  $G$ . In digraph  $G$ , let

$\langle u, v \rangle$  denotes a link from node  $u$  to node  $v$ , and let  $c(u)$  denotes the outdegree of node  $u$ . Let  $n$  be the number of nodes in the digraph,  $G$  can be represented with a  $n \times n$  dimensional transition matrix  $P$  whose element  $P_{uv}$  is  $1/c(u)$  if there is a link from  $u$  to  $v$  and is zero otherwise. For example, the transition matrix of a small fraction of web containing six pages linked shown in Fig. 1 can be represented as

$$\mathbf{P} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/2 & 1/2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1/2 & 1/2 \\ 0 & 0 & 0 & 1/2 & 0 & 1/2 \\ 1/2 & 0 & 0 & 1/2 & 0 & 0 \end{pmatrix}.$$

The rank vector on the graph represents the importance of each node. The basic conception underlying link based rank was that the link  $\langle u, v \rangle$  can be viewed as evidence that  $v$  is important for  $u$ . In PageRank, the rank vector  $\mathbf{w}^T$  was defined recursively as

$$\mathbf{w}^T = \alpha \mathbf{w}^T P + (1 - \alpha) \mathbf{v}^T, \quad (1)$$

where  $\mathbf{w}^T$  and  $\mathbf{v}^T$  are both  $n$  dimensional row vectors and  $\alpha$  is the damping factor and  $0 \leq \alpha < 1$ .  $\mathbf{v}^T$  is known as personalized vector and can be assigned  $n$  dimensional uniform row vector generally. The iterative form of equation (1) can be written as

$$\mathbf{w}^{(k)T} = \alpha \mathbf{w}^{(k-1)T} P + (1 - \alpha) \mathbf{v}^T. \quad (2)$$

Start from  $\mathbf{w}^{(0)T}$ ,  $\mathbf{w}^{(k)T}$  can be calculated using power method.

## 2.2 ExpRank

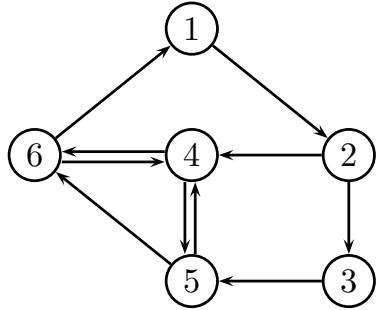
From Eq. (2), we can obtain

$$\mathbf{w}^{(k)T} = \mathbf{w}^{(0)T} (\alpha P)^k + (1 - \alpha) \mathbf{v}^T (I + \sum_{i=1}^{k-1} (\alpha P)^i). \quad (3)$$

Since the spectral radius of  $P$  equals to 1,  $(\alpha P)^k$  converges to zero when  $k \rightarrow \infty$ . Thus, when  $k \rightarrow \infty$ , Eq. (3) can be written as

$$\mathbf{w}^T = (1 - \alpha) \mathbf{v}^T (I + \sum_{i=1}^{\infty} (\alpha^i P^i)) = (1 - \alpha) \mathbf{v}^T (I - \alpha P)^{-1}. \quad (4)$$

Eq. (4) is the stable solution of PageRank algorithm, which can also be achieved by solving the linear equation of Eq. (1). Thus Eq. (4) is the ideal



**Fig. 1.** Digraph representing web containing six pages

rank result attained from PageRank algorithm. It can be viewed as the accumulated effect of the vector sequence produced along with the transition of  $\mathbf{v}^T$  on digraph  $G$ . Thus, the generalization form of PageRank can be written as

$$\mathbf{w}^T = \mathbf{v}^T(I + \sum_{i=1}^{\infty} (\Gamma_i P^i)), \quad (5)$$

where  $\Gamma$  is a sequence of coefficients converging to zero.

Let  $\Gamma = \{\beta^n/n! | n \in N, \beta \geq 0\}$ , and this leads to our ExpRank algorithm. In ExpRank, Eq. (5) becomes

$$\mathbf{w}^T = \mathbf{v}^T(I + \sum_{i=1}^{\infty} (\frac{\beta^i}{i!} P^i)) = \mathbf{v}^T e^{\beta P}. \quad (6)$$

$\beta$  is the decay factor used to adjust the decay rate of  $\Gamma$ , and  $\mathbf{v}^T$  is a personalized vector to customize ExpRank for the purpose of spam or user-dependent applications.

---

**Algorithm 1.** ExpRank Computation

---

```

 $k \leftarrow 0$ 
 $\mathbf{t}^T \leftarrow \mathbf{v}^T$ 
 $\mathbf{w}^T \leftarrow \mathbf{v}^T$ 
repeat
     $k \leftarrow k + 1$ 
     $\mathbf{t}^T \leftarrow \beta \mathbf{t}^T P$ 
     $\mathbf{w}^T \leftarrow \mathbf{w}^T + \mathbf{t}^T / k!$ 
     $\delta = \|\mathbf{t}^T / k!\|_1$ 
until  $\delta < \varepsilon$ 

```

---

For the computation of ExpRank, the exponential of matrix does not need to be computed apparently, and only the results of vector-matrix multiplication were required. The process of ExpRank computation is described in Algorithm 1. Algorithm 1 is matrix-free and only  $nnz(P)$  multiplications are needed for each iteration, where  $nnz(P)$  is the number of non-zeros in  $P$ . Since generally there is no more than 10 non-zeros per row in  $P$ ,  $O(nnz(P)) \approx O(n)$ . Only the storage of two vector  $\mathbf{t}^T$  and  $\mathbf{w}^T$  is required at each iteration. Thus, this algorithm is suitable for the size and sparsity of web matrix.

### 3 Experimental Results

In digraph  $G$ , the nodes without outlinks were called dangling nodes. In PageRank, dangling nodes need to be pretreated to avoid rank sinking. This pretreatment is not necessary for ExpRank calculation, and ExpRank can be calculated with the transition matrix  $P$  including dangling nodes directly. However, for the

convenience of comparison with PageRank, all dangling nodes were removed in our experiments.

The dataset used in our experiments was generated from a crawl within China in 2004. The link graph contains 92382 nodes and 238430 links. After removing the dangling node recursively, there are 33519 nodes and 164110 links left for the ExpRank and PageRank computation. In the following results of our experiments,  $\beta = 3$ ,  $\alpha = 0.85$  and  $\mathbf{v}^T$  is uniform if not specified. The uniform vector was also assigned to  $\mathbf{w}^{(0)T}$  in PageRank calculation.

ExpRank vector  $\mathbf{w}_{\text{exp}}^T$  and PageRank vector  $\mathbf{w}_{\text{page}}^T$  were plotted in semilog plots in Fig. 2 with  $x$ -axis indicates labels of nodes.  $\mathbf{w}_{\text{exp}}^T$  and  $\mathbf{w}_{\text{page}}^T$  are both normalized. As most rank values were very small,  $y$ -axes of  $\mathbf{w}_{\text{exp}}^T$  and  $\mathbf{w}_{\text{page}}^T$  plots were logarithmically scaled.  $\mathbf{r}^T = \mathbf{w}_{\text{exp}}^T - \mathbf{w}_{\text{page}}^T$  was also plotted in this figure. The average of  $\mathbf{r}^T$  is  $-6.398 \times 10^{-19}$  and the variance is  $6.586 \times 10^{-10}$ .

We measure the rates of convergence using the L1 norm of the residual vector, i. e.,

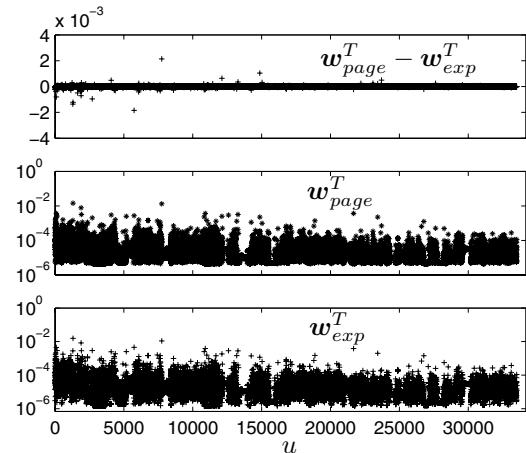
$$\Delta^{(k)} = \|\mathbf{w}^{(k)T} - \mathbf{w}^{(k-1)T}\|_1.$$

Since  $P$  is a stochastic matrix and  $\mathbf{v}^T$  is uniform, from Algorithm 1 we can obtain that  $\Delta_{\text{exp}}^{(k)} = \beta^k / k!$ . Thus, the convergence rate of ExpRank is much faster than the convergence rate of PageRank  $O(\alpha^k)$ . The convergence rates of ExpRank and PageRank in our experiments were plotted on semi-log graph shown in Fig. 3.

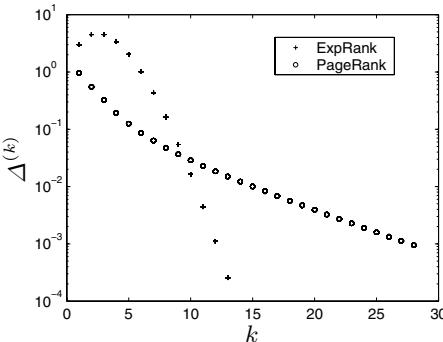
As the order of nodes is important in a rank, the percent of nodes reversed can be used as a valuable evidence to evaluate the difference between ExpRank and PageRank. Let  $\mathbf{r}_1$  and  $\mathbf{r}_2$  are two different rank containing  $m$  nodes, we define  $\Psi(\mathbf{r}_1, \mathbf{r}_2)$  as the ratio of reversed order between two rank, i.e., the number of node pairs reversed between two ranks divided by all possible reversed node pairs. If the order of a node changes from  $a$  to  $b$ , there are  $|a - b|$  reversals. Thus,  $\Psi(\mathbf{r}_1, \mathbf{r}_2)$  can reflect the quantity of changed

node pairs and the amplitude of the change of one node's rank order simultaneously.  $\Psi(\mathbf{r}_{\text{exp}}^T, \mathbf{r}_{\text{page}}^T)$  of top  $m$  nodes in ExpRank vs.  $m$  when  $m \leq 1000$  is plotted in Fig. 4. The relatively high difference in the low region of  $m$  is due to the small value of  $m$  which highlights the reversals of rank order.

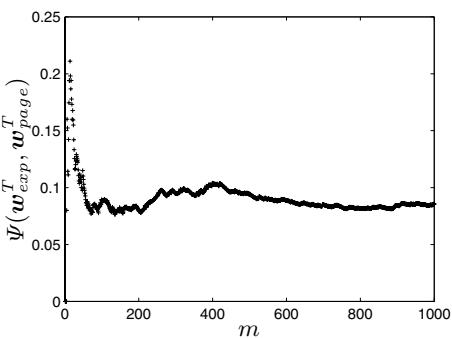
From Fig. 2 and Fig. 4, we can infer that the difference between the results of ExpRank and PageRank is not very obvious, especially for nodes with high rank, which is more meaningful for the users of search engines.



**Fig. 2.** Comparison of ExpRank and PageRank



**Fig. 3.** The convergence rates of ExpRank and PageRank



**Fig. 4.** The ratio of reversed order for high ranked nodes

## 4 Conclusion

From the above discussion, we can conclude that in ExpRank the main features of PageRank were reserved such as query independent, content independent, simplicity, suitable for the huge and extreme sparse web matrix, etc. The convergence rate of ExpRank is much faster and ExpRank calculation can also be achieved using Krylov subspace approximation method [6]. So it could be a good candidate for the increasing needs of quick rank algorithms, along with the rapid development of World Wide Web and the progression of specific and customized search engines.

## References

1. Brin, S., Page, L., Motwani, R., Winogard, T.: The pagerank citation ranking: Bring order to the web. Technical report, Stanford University (1999) Available at <http://dbpubs.stanford.edu:8090/pub/1999-66>.
2. Jeh, G., Wiedom, J.: Scaling personalized web search. Technical report, Stanford University (2002) Available at <http://dbpubs.stanford.edu:8090/pub/2002-12>.
3. Haveliwala, T.H.: Topic-sensitiv pagerank. In: Proc. of the Eleventh International World Wide Web Conference, New York: ACM Press (2002) 517–526
4. Kamvar, S.D., Haveliwala, T.H., Manning, C.D., Golub, G.H.: Extrapolation methods for accelerating pagerank computations. In: Proc. of the Twelfth International World Wide Web Conference, New York: ACM Press (2003) 261–270
5. Langville, A.N., Meyer, C.D.: Deeper inside pagerank. Internet Mathematics **1** (2003) 335–380
6. Hochbruck, M., Lubich, C.: On krylov subspace approximations to the matrix exponential operator. SIAM J. Numer. Anal. **34**(5) (1997) 1911–1925

# A Biologically-inspired Computational Model for Perceiving the TROIs from Texture Images

Woobeam Lee and Wookhyun Kim

Department of Computer Engineering, Yeungnam University  
214-1 Dae-dong, Gyeongsan-si, Gyeongbuk-do 712-749, Republic of Korea  
[{beomlee, whkim}@ynu.ac.kr](mailto:{beomlee, whkim}@ynu.ac.kr)

**Abstract.** This paper presents a biologically-inspired method of perceiving the TROIs(: Texture Region Of Interest) from various texture images. Our approach is motivated by a computational model of neuron cells found in the primary visual cortex. An unsupervised learning schemes of SOM(: Self-Organizing Map) is used for the block-based image clustering, plus 2D spatial filters referring to the response properties of neuron cells is used for extracting the spatial features from an original image and segmenting any TROI from the clustered image. To evaluate the effectiveness of the proposed method, various texture images were built, and the quality of the extracted TROI was measured according to the discrepancies. Our experimental results demonstrated a very successful performance.

## 1 Introduction

Texture analysis using a biologically-motivated Gabor scheme is the most effective technique and represents the sate-of-the-art in this area. Two major approaches have been studied for using merely one filter in this literature.

One is the supervised method that uses a bank of Gabor filters[1,2]. These methods are restricted within a supervised problem of fore-knowledge and a high computational complexity issue. The other is the unsupervised method that designs a single Gabor filter, which is distinctly responding to the specific texture component[3,4]. Although it is an unsupervised methods, optimal filtering has focused on detecting only pertinent texture component and using the texture information inherent to a particular image with fore-knowledge.

Accordingly, this paper proposes a biologically-inspired method like the human behavior recognizing the TROIs in an image without fore-knowledge, and providing a useful edge-information for object recognition from a query image.

This paper focuses on implementing 2D spatial filters corresponding to the receptive field of neuron cells such as retinal ganglion cell( $c_G$ ) and two types of simple cell( $c_{S1}$ ,  $c_{S2}$ ) found in the primary visual cortex.  $c_{S1}$  cells extract a orientation selective spatial feature from an original image, and an unsupervised learning scheme of SOM clusters an original image into the block-based parts. Also, A  $c_{S2}$  cell responds to the selective attention for segmenting any TROI from the clustering results automatically without fore-knowledge, and A  $c_G$  cell responses to the contrast extraction for detecting the edge of any TROI.

## 2 A Self-organized Clustering Model(: $c_{S1}$ Cell)

A  $c_{S1}$  cell is defined by using the asymmetrical difference of two Gaussian function with a preferred orientation  $\phi$ , as follows:

$$c_{S1}(x', y', \phi) = \left( \exp\left(-\frac{x'^2}{2\sigma_e^2}\right) - \frac{\sigma_e}{\sigma_i} \exp\left(-\frac{x'^2}{2\sigma_i^2}\right) \right) \cdot \exp\left(-\frac{y'^2}{2\sigma_{en}^2}\right) \quad (1)$$

where  $(x', y') = (x \cos \phi + y \sin \phi, -x \sin \phi + y \cos \phi)$  are rotated coordinates,  $\sigma_e$  and  $\sigma_i$  represent the space constants of excitatory and inhibitory regions respectively, and  $\sigma_{en}$  determines the sensitivity of preferred orientation of the filter. This filter corresponds to a simple cell receptive field found in mammal's visual cortex domain. Simple cells are excellent at detecting the presence of simple visual features, such as lines and edges of a particular orientation[5]. The output of  $c_{S1}$  cell is given by

$$u_{S1}(x, y, \phi) = \varphi \left[ \frac{1 + \int \int_{A_{S1}} c_S^+(\xi, \eta, \phi) \cdot i(x + \xi, y + \eta) d\xi d\eta}{1 + \int \int_{A_{S1}} c_S^-(\xi, \eta, \phi) \cdot i(x + \xi, y + \eta) d\xi d\eta} - 1 \right], \quad (2)$$

where  $A_{S1}$  denotes the radius of receptive field satisfying  $|\xi|^2 + |\eta|^2 \leq |A_{S1}|^2$ ,  $c_S^+(\cdot)$  and  $c_S^-(\cdot)$  represent the strength coefficients of excitatory and inhibitory connections respectively,  $i(x, y)$  is a gray level intensity in image, and  $\varphi[\cdot]$  is a step function. After response of the  $c_{S1}$  cells is completed for each preferred orientation in an image, the image is divided into the equal-sized block parts by the unsupervised learning scheme, which is proposed by Kohonen[6,7].

## 3 A Selective Attention Model(: $c_{S2}$ Cell)

A  $c_{S2}$  cell corresponds to another simple cell receptive field found in mammal's visual cortex domain.  $c_{S2}$  cells are more appropriate for a selective attention of an image containing a very specific frequency and orientation characteristic[5]. Thus, this cell acts as a local bandpass filter. This cell is defined by using a 2D Gabor function form, as follows:

$$\begin{aligned} c_{S2}(x, y; \sigma, u_0, v_0, \lambda, \phi) \\ = g(x', y'; \sigma) \cdot \exp(-2\pi i(u_0 x + v_0 y)) = g(x', y'; \sigma) \cdot \exp(-2\pi i f_0 x') \\ = g(x', y'; \sigma) \cdot (\cos(2\pi f_0 x') - i \sin(2\pi f_0 x')), \end{aligned} \quad (3)$$

where

$$g(x, y; \sigma) = \left( \frac{1}{2\pi\lambda\sigma^2} \right) \cdot \exp\left(-\frac{(x/\lambda)^2 + y^2}{2\sigma^2}\right),$$

$(x', y') = (x \cos \phi + y \sin \phi, -x \sin \phi + y \cos \phi)$  are rotated coordinates,  $\lambda (= b/a)$  specifies the aspect ratio,  $\sigma$  is the standard deviation of the Gaussian envelope. Also, the radial center frequency  $f_0$  can be calculated as  $f_0 = \sqrt{u_0^2 + v_0^2}$ ,

and  $\lambda$ ,  $\phi$ , the center frequency  $(u_0, v_0)$  of the Gabor function is defined as follows:

$$u_0 = f_u/N, \quad v_0 = f_v/M, \quad \lambda = 0.5, \quad \phi = \theta(= \tan^{-1}(v_0/u_0)). \quad (4)$$

Here  $N$  and  $M$ , considering as  $N = M$  generally, are the spatial resolution of the perceived TROI, and  $1/N$  is the frequency sample interval. Tuning frequency  $(f_u, f_v)$  for a selective attention of a TROI is then defined as follows:

$$(f_u, f_v) = \arg \left\{ \max_{(1 \leq k \leq m)} \left\{ \sum_{i=1(i \neq t)}^n \|FS_t^k - FS_i^1\| \right\} \right\} \quad (5)$$

where  $FS_t^k$  is the  $k$ -th maximum frequency in the  $TROI_t$ , which corresponds to the  $k$ -th center frequency that is ordered by the Fourier spectrum of the  $TROI_t$ , and  $FS_i^1$  are the highest center frequency in the  $TROI_i$ . Also,  $m$  is the number of the ordered spatial frequency candidate in the  $TROI_t$ , and  $n$  is the number of the clustered TROIs in an image[7].

The response of a  $c_{S2}$  cell,  $u_{S2}(\cdot)$  can be defined in the form of Eq. (6).

$$u_{S2}(x, y) = \left[ \left( \int \int_{A_{S2}} c_{S2}^R(\xi, \eta) \cdot i(x + \xi, y + \eta) d\xi d\eta \right)^2 + \left( \int \int_{A_{S2}} c_{S2}^I(\xi, \eta) \cdot i(x + \xi, y + \eta) d\xi d\eta \right)^2 \right]^{1/2} \quad (6)$$

where  $A_{S2}$  denotes the distance of receptive field satisfying  $|\xi/a|^2 + |\eta/b|^2 \leq |A_{S2}|^2$ ,  $a$  and  $b$  denote the two scale parameters, and  $c_{S2}^R(\cdot)$  and  $c_{S2}^I(\cdot)$  represent the strength coefficients of the real and imaginary parts respectively.

#### 4 A Contrast Extraction Model(: $c_G$ Cell)

The edge detection of a TROI requires a binary image of the  $u_{S2}(\cdot)$  output. Thus, the below Eq. (7) is applied to the  $u_{S2}(\cdot)$  output of any TROI ,

$$u_B(x, y) = \varphi[u_{S2}(x, y)] = \begin{cases} 1 & \text{if } (\lceil \omega \times H \rceil / \omega) \leq x \leq (\lfloor \omega \times L \rfloor / \omega) \\ 0 & \text{otherwise.} \end{cases} \quad (7)$$

where  $H$  and  $L$  are the highest and lowest response value, respectively, for  $u_{S2}(\cdot)$ , in any clustered TROI,  $\omega$  is the precision coefficient, and  $\lceil \cdot \rceil$  and  $\lfloor \cdot \rfloor$  denote the *ceiling()* function and *floor()* function, respectively, for the truncation using the integer transformation. Thus, without any fore-knowledge or heuristic decision, the upper and lower bound for the binary image transformation of the  $u_{S2}(\cdot)$  can be automatically determined.

The final segmentation is achieved by the contrast-extracting property of a retinal ganglion cell. This cell corresponds to an on-center and off-surround receptive field of ganglion cells found in the retina of the visual pathways[5]. A  $c_G$

cell is defined by using a set of 2D circular symmetric difference of two Gaussian function, as follows:

$$c_G(x, y) = \frac{1}{2\pi\sigma_e^2} \exp\left(-\frac{x^2 + y^2}{2\sigma_e^2}\right) - A \frac{1}{2\pi\sigma_i^2} \exp\left(-\frac{x^2 + y^2}{2\sigma_i^2}\right). \quad (8)$$

where  $\sigma_e$  and  $\sigma_i$  represent the space constants of the excitatory and inhibitory regions, respectively, and the ratio of the space constants  $\sigma_e/\sigma_i = 1.6$ . The ratio yields a good approximation of the ideal Laplacian operator. The output of a  $c_G$  cell can be defined by

$$u_G(x, y) = \psi\left[\int \int_{A_G} c_G(\xi, \eta) \cdot u_{S2}(x + \xi, y + \eta) d\xi d\eta\right]. \quad (9)$$

where  $\psi[\cdot]$  is a function for finding the zero crossing of the  $c_G$  cell correspondence.

## 5 Computer Simulation

To evaluate the quality of the segmentation performance, given more than 100 texture images obtained from the Brodaz texture book , the segmentation quality was measured according to the discrepancies based on the number of mis-segmented pixels[9], as defined below:

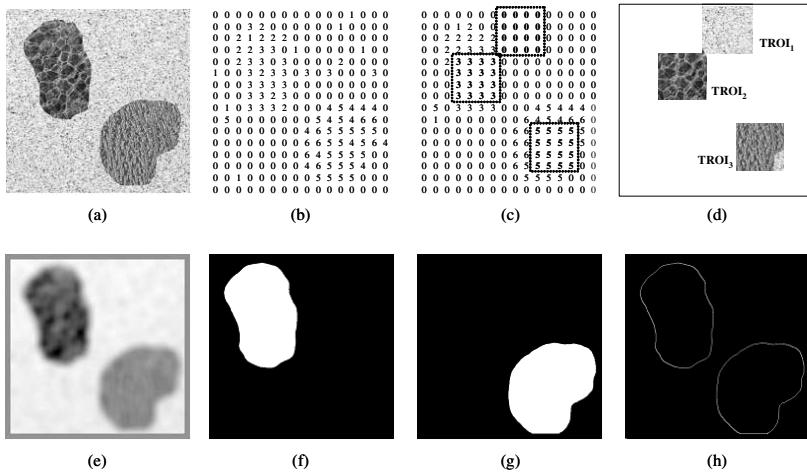
$$D_k = 100 \times \left[ \left( \sum_{i=1}^N C_{ik} \right) - C_{kk} \right] / \left[ \sum_{i=1}^N C_{ik} \right] \quad (10)$$

where  $C_{ij}$  represents the number of cluster  $j$  pixels classified as cluster  $i$  in the segmentation results. The results were measured as close to 5% for the two-texture problem, and 7% for three-texture problem. this results show that the performance of the proposed system was very successful.

## 6 Conclusions

A biologically-inspired computational model was presented for unsupervised perceiving the TROIs from a texture image. This paper focuses on (1) implementing a 2D spatial filters corresponding to the receptive field of neuron cells such as ganglion cell in the retina and two types of simple cells found in the primary visual cortex, (2) proposing an unsupervised learning scheme for clustering the TROIs without fore-knowledge, and (3) detecting edge of any TROI from the perceived results automatically.

However, several problems remain for future work: the size of a block unit and the number of preferred orientation. In particular, the selection method of the appropriate parameters, such as the orientation, phase, and aspect ratio, is an important task when tuning a selective attention. Consequently, when these problems are solved, and the object-shape definition of the extracted TROI is also included, the proposed method has potential application for the development of a real image query system.



**Fig. 1.** Experimental Results: (a) Collage of the Brodaz textures Background(Sand), D112(Plastic bubbles) and D24(Pressed calf leather), where the size of original image is 512 x 512 pixel. [ $u_{S1}(\cdot)$  and SOM]: (b) Clustered map and (c) Merged map, where one of the block unit is scaled to a size of 32x32 pixel. Bounding box corresponds to the maximum square including the TROI and preserving the block-based connectivity with respect to the same label. (d) Results of browsing the TROIs. [ $u_{S2}(\cdot)$ ]: (e) Result of a selective attention by tuning a spatial frequency of the TROI(D112). [ $u_B(\cdot)$ ]: (f) Binary image of extracted TROI(D112). (g) Binary image of extracted the TROI(D24). [ $u_G(\cdot)$ ]: (h) Edge detected image of the TROIs(D112 and D24) by zero-crossing.

## References

1. Manthalkar, R., etc.: Rotation invariant texture classification using even symmetric Gabor filters. *Pattern Recognition Letters* **24** (2003) 2061-2068
2. Idrissa, M., Achery, M.: Texture classification using Gabor filters. *Pattern Recognition Letters* **23** (2002) 1095-1102
3. Tsai, D., etc.: Optimal Gabor filter design for texture segmentation using stochastic optimazation. *Image and Vision Computing* **19** (2001) 299-316
4. Clausi, D. A., Jernigan, M.: Designing Gabor filters for optimal texture seperability. *Pattern Recognition* **33** (2000) 1835-1849
5. Marr, D.: Vision: A computational investigation into the human representation and processing of visual information. W. H. Freedom&Company (1982)
6. Kohonen, T.: The self-organizing map. *Proc. IEEE* **78 (9)** (1990) 1464-1480
7. Lee, W.B., Kim, W.H.: Texture Segmentation by Unsupervised Learning and Histogram Analysis using Boundary Tracing. In: Yue, H., et al (eds.): Computational Intelligence and Security. Lecture Notes in Artificial Intelligence, Vol. 3801. Springer-Verlag, Berlin Heidelberg New York (2005) 25-32
8. Fukushima, K.: Neural network model for extracting optical flow. *Neural Networks* **18** (2005) 549-556
9. Zhang, Y. J.: A survey on evaluation methods for image segmentation. *Pattern Recognition* **29 (8)** (1996) 1335-1346

# A Computer-Assisted Environment on Referential Understanding to Enhance Academic Reading Comprehension

Wing-Kwong Wong<sup>1</sup>, Jian-Hau Lee<sup>1</sup>, Yu-Fen Yang<sup>2</sup>, Hui-Chin Yeh<sup>2</sup>,  
Chin-Pu Chiao<sup>2</sup>, and Sheng-Cheng Hsu<sup>3,4</sup>

<sup>1</sup> Graduate School of Computer Science & Information Engineering,

<sup>2</sup> Graduate School of Applied Foreign Languages,

<sup>3</sup> Graduate School of Engineering Science & Technology,

National Yunlin University of Science & Technology, Douliou, Yunlin, Taiwan  
{wongwk, g9317708, yangy, hyeh,  
chiaocp, g9010802}@yuntech.edu.tw

<sup>4</sup> Department of Information Management, Nan Kai Institute of Technology,  
Tsaotun, Nantou, Taiwan

**Abstract.** To comprehend English-written texts successfully, readers have to construct a referential map of textual information. Referential device is one of the important means for helping readers comprehend a text. The purpose of this study is to develop a computer-assisted environment to enhance EFL college students' comprehension. Four modules, natural language processing (NLP), user interface, recording, and feedback module are included. Among these four modules, the feedback module compares students' initial maps with the expert's. The results of comparison will inform students what referents are incorrect and offer them appropriate scaffoldings. The recording module records all of students' behavioral data. From the data, the teacher can identify the difficulties students encounter and different performance levels among students with various reading proficiencies.

## 1 Introduction

To comprehend a text successfully, the reader has to construct a comprehensible and coherent mental representation of the textual information in his/her memory [1]. The mental representation shows a network of the text, with nodes that indicate individual text elements and connections which show the meaningful relations [1]. The ability of integrating textual information is one of the essential skills for reading comprehension [2] [3].

Halliday and Hansan [4] proposed five cohesive ties to help the reader integrate textual meaning and form a coherent mental picture toward the information presented in a text; they are reference, substitution, ellipsis, conjunction, and lexical cohesion. Among these five cohesive ties, referential device accounts for 75% of the variation for students who learn English as a Foreign Language (EFL) in academic reading comprehension [5]. Three major types of references are included: personal, demonstrative, and comparative references. Personal reference refers to individuals or objects by specifying their functions or roles in the speech situation [4], such as "I", "me", and "you." Demonstrative reference acts as a form of verbal locating, such

as “this”, “these”, and “that.” The speaker figures out the referent by means of location, on a scale of proximity. Comparative reference refers to the indirect reference for indicating identity or similarity.

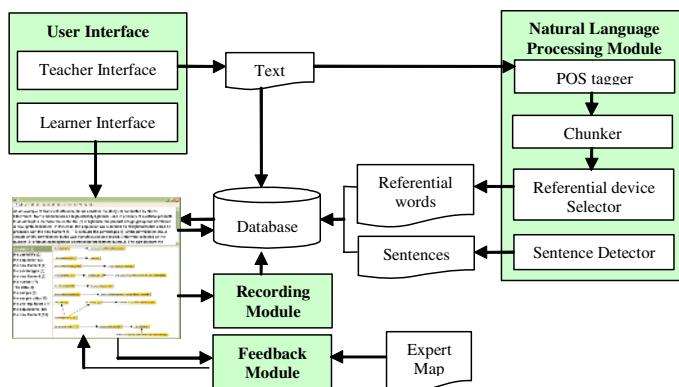
As cohesive signals for connecting sentences in texts occur less often in Chinese [6], most Taiwanese EFL students are found to be less aware of them while reading English texts. They rarely use cohesive devices or referential words for integrating textual information [7][8]. The difficulty in identifying references results in their lower English reading proficiency.

EFL college students are expected to be equipped with sufficient English reading skills since most of them have to acquire domain-specific knowledge by reading texts. That is, they are required to apply basic skills to read English for Specific Purpose (ESP) texts, such as statistics, physics, or chemistry. These basic skills include understanding relations within a sentence and between sentences, using cohesive and discourse markers, predicting, inferring, guessing, processing, and evaluating the information during reading [9].

This study aims to develop a computer-based reference instruction for enhancing their comprehension in reading texts and exploring the relationship between referential understanding and reading comprehension among EFL college students.

## 2 English Learning Environment

The English learning environment built in this study includes four modules: *natural language processing (NLP)*, *user interface*, *recording*, and *feedback modules*. Fig. 1 shows the system architecture of the learning environment.



**Fig. 1.** System Architecture

The teacher designs the course, select texts for students to read, and input the texts to *NLP module* through a *teacher interface*. The *NLP module* picks all of the referential devices from each text and segments the text into sentences. The selected referential devices and the sentences are then saved in the database.

The *recording module* records student's keyboard and mouse actions while he/she constructs a referential map of the text. These recorded behavioral data are then

reported back to the teacher who identifies the difficulties the student encounter and the performances among groups of different reading proficiencies. After the student has constructed his/her initial referential map, the *feedback module* compares the map with the expert's map. It then reports their differences back to the student.

## 2.1 Natural Language Processing Module

To search all referential devices and segment a text into sentences automatically, we choose OpenNLP [10] to help us do *part-of-speech tagging*, *chunking*, and *sentence detection*. It provides a variety of Java-based tools which perform sentence detection, tokenization, pos-tagging, chunking and parsing, named-entity detection [10]. For example, the referential devices of the text in Fig. 2 are detected in Fig. 3.

*Dr. Chen is interested in using the sample data to make an inference about the average hours of useful life for the populations of all lightbulbs that could be produced with the new filament.*

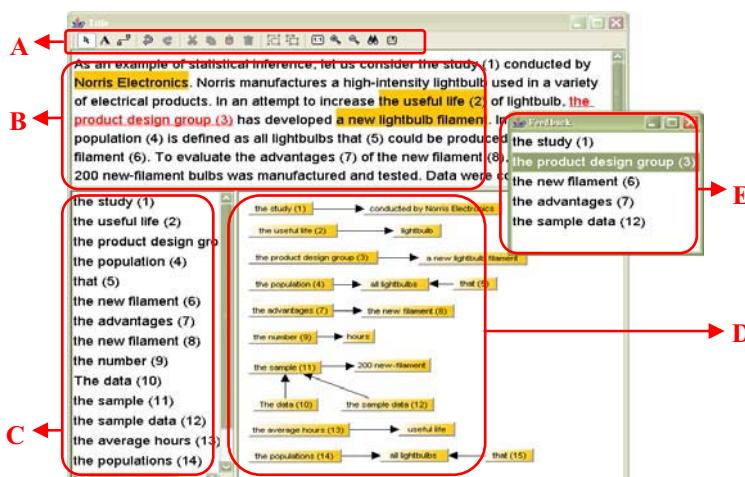
**Fig. 2.** An example text

[NP the sample data]  
[NP the average hours]  
[NP the populations]  
[NP that]  
[NP the new filament]

**Fig. 3.** Referential devices detected by NLP module

## 2.2 User Interface

The *user interfaces* include *teacher interface* and *learner interface*. With the *teacher interface*, the teacher manages course data, input texts which students should read, and access students' behavioral data. With the *learner interface*, students input the relationships among referents. Fig. 4 shows the *learner interface*.



**Fig. 4.** Learner interface and Feedback received by students

**A. Toolbar.** Toolbar includes many graphic tools. Text tool can add text elements in the canvas; connection tool can establish meaningful relations among referential devices. Other tools in helping students manage the canvas are cut, copy, paste, erase, group, ungroup, zoom in, zoom out, undo, redo, and request hint.

**B. Text field.** Text field shows a reading text. Students can select a word or a sentence as a text element and drag into the canvas directly when they comprehend the relations among the text. The entire sentence will be highlighted when students click at a word.

**C. Referential device list.** All referential devices found by the *NLP module* are listed in this area. Students are asked to figure out what these referential devices refer to, drag and drop them to the canvas. Students can also add other referential devices from the text to the list by themselves. Moreover, when a referential device is selected, the referential device will also be highlighted in the text.

**D. Canvas.** Students draw arrows indicating the relations among referents in the canvas. They can *add*, *erase*, *drag* and *drop* elements in the canvas, and establish relations among these text elements. The map shows a simplified network about the text, with nodes of individual text elements and connections which show the relations between elements in the text [1].

**E. Feedback frame.** The feedback frame will inform students what referents are incorrect and offer him/her some clues about the correct referents. The details of the *Feedback module* will be discussed later in this article.

### 2.3 Recording Module

While students construct the referential map, the system records all of their actions by *recording module*. Students' behavioral data and maps will be saved by the system in the database so that the teacher can check out these data later. The module uses some predicates to record students' behavioral data. Table 1 shows some of the predicates.

**Table 1.** Student's actions

Predicates	Description
select_sentence(T)	Select a sentence T which a student is reading in text field.
element(X)	Add an element X to the canvas.
erase_element(X)	Erase an element X in the canvas.
refer_to(X, Y)	Add a relation between X and Y.
erase_reference(X, Y)	Erase a relation between X and Y.
...	...

### 2.4 Feedback Module

After a student has constructed his/her initial referential map, he/she can request an evaluation of the map. The *feedback module* will compare the student's map with the expert's map and then inform the student what referents are incorrect and offer him/her some clues about the correct referents. Fig. 4 shows the feedback received by

a student. When an incorrect referent is chosen by the student, the clues about the referent will be highlighted in the *text field*. The types of mistakes can be divided into missing referents, incorrect referents, and incorrect direction of referential relation.

### 3 Conclusion

In this study, a computer-assisted environment is developed to enhance EFL college students' reading comprehension. Natural language processing technology is used to automatically search referential devices and segment the texts into sentences. The *recording module* records students' actions and then reports back to the teacher. From these data, the teacher can further identify students' reading strengths and weaknesses. This would be helpful for the teacher to adjust their teaching in order to reduce students' reading difficulties. After students construct their initial referential map, the *feedback module* will compare the map with the expert's map. If there are mistakes in the student's map, the *feedback module* will offer them a second chance in correcting the referents. Based on the results of comparison, students are told what referents are incorrect, and get hints on candidate referents.

In the future, this computer-assisted learning environment will be empirically tested by EFL college students, and the experimental results will be reported in subsequent studies. The system also plans to add a module which automatically generates the expert's map from a text. We will use the lexical resource of *WordNet* [11] to help us to resolve referential ambiguities.

### Acknowledgement

This project is supported by the National Science Council, Taiwan (NSC 94-2520-S-224-001) and the Center for English Teaching Resources (MOE 0940052270).

### References

1. Broek, P. V. D., Kremer, K. E.: The Mind in Action: What It Means to Comprehend During Reading. In Taylor, B. M., Graves, M. F., & Broek, P. V. D. (Ed.) *Reading for Meaning: Fostering Comprehension in the Middle Grades*. Allyn & Bacon (1998) 1-31
2. Peterson, C.: Identifying Referents and Linking Sentences Cohesively in Narration. *Discourse processes*, 16 (1993) 507-524
3. Grabe, W., Stoller, F. L.: *Teaching and Researching Reading*. Pearson Education (2002)
4. Halliday, M. A. K., Hasan, R.: *Cohesion in English*. Longman Group Ltd (1976)
5. Huang, S. H.: Assessing the Relationship between Referential Understanding and Academic Reading Comprehension among EFL College Students. A thesis of institute of applied foreign languages, National Yunlin University of Science and Technology, Taiwan (2005)
6. Chu, H. C. J., Swaffar, J., Charney, D. H.: Cultural Representation of Rhetorical Conventions: the Effects on Reading Recall. *TESOL Quarterly*, 36(4), (2002) 511-541

7. Chen, L. T.: Improving High School Students' Performance on "Discourse Structure" Tests through Instruction of Text Structure and Think-aloud Modeling. A thesis of institute of English department, National Taiwan Normal University, Taiwan (2003)
8. Sharp, A.: Reading Comprehension and Text Organization. U. S. The Edwin Mellen Press, Ltd (2003)
9. Dudley-Evans, T., St. John, M. J.: Developments in ESP a Multi-Disciplinary Approach. Cambridge university press (1998)
10. OpenNLP, available electronically from <http://opennlp.sourceforge.net/>
11. WordNet, available electronically from <http://wordnet.princeton.edu/>

# An Object-Oriented Framework for Data Quality Management of Enterprise Data Warehouse

Wang Li and Li Lei

Software Institute of Sun Yat-Sen University  
510630 Guangzhou, China  
Mr.wangli@163.com

**Abstract.** Enterprise data warehousing technology aims at providing integrated, consolidated and historical data for users to analyze businesses and make decisions. In order to obtain the correct results, the high data quality is required. In this paper, we analyze the quality problems of enterprise data warehouse and present an object-oriented framework for data quality management. In this framework, an object-oriented data quality model (OODQM) is built. The data quality requirements, the participants, the data quality checking object, and the possible data quality problems, form the core components of OODQM. The method we provide is a goal-driven method. Once the data quality goal is built, we manage data quality by the interaction of those components of OODQM.

## 1 Introduction

Effective, real, and accurate data is a necessary condition for enterprise data warehouse (EDW) building. We should build a series of managing mechanisms to guarantee the data quality before the data is used to make decision.

Many experts study on the data quality, and the issues include data quality requirements analysis and modeling [4], data quality defining [5, 10], data quality assessment [7], and data quality management [1, 2, 3, 8, 9]. R.Y. Wang builds the framework of data quality analyzing [10], and provides the attribute-based method [2] which attaches the data quality measure to attribute and build the data quality model based on attributes. Yang W. Lee [3] provides an approach to manage data quality based on context-reflective which includes paradigm, role, goal, time, and place. DWQ [1, 9] extends the architecture of EDW by building the metadata model of data quality and embedding them into the concept model, logic model, and physical model of EDW respectively. The quality data is acquired, stored, and maintained in the metadata model. These researches build and develop the foundation of theories and techniques for guaranteeing data quality.

In this paper, we build a novel data quality management framework for EDW according to the actual experience of project implementing. EDW has special data environments which integrate many homogeneous or heterogeneous data. We analyze the origins of data quality problems in these environments, establish an object-oriented data quality model, and provide the data quality processing method.

## 2 Definition of Data Quality and Data Environments of EDW

The definitions of data quality are different for different categories [5, 10]. For EDW, we define the definition of data quality from six dimensions according to the actual projects we did. We measure every dimension quantitatively.

*Completeness*— There are two kinds of completeness, one is *business rule completeness (BRC)*, and the other is *value of attribute completeness (VAC)*. BRC is the percent of data which satisfy business requirements. VAC is the percent of records which have right value.

*Correctness*— Correctness is the percent of data in accord with facts.

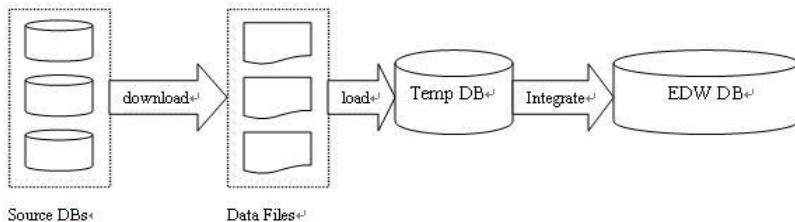
*Usability*—Usability is the percent of data which can be used.

*Currency*— Currency is the percent of data which are current for the application of EDW, namely, the percent of data refreshed during the accepted tolerance of delay.

*Consistency*— Consistency is the percent of data which don't conflict with others.

*Relevance*— Relevance is the percent of data which can be related with other data according to business requirements.

Data environments are the environments where the data exist. The generally data environments of EDW are shown in figure 1. The data quality problems are mostly caused by the source data. Logical data model (LDM) and ETL design would produce wrong data too. Errors would occur when the data are downloaded, transmitted, loaded, or integrated by ETL program. Data quality problems will be produced during the process of EDW running.



**Fig. 1.** Data Environments

## 3 Framework for Data Quality Management of EDW

### 3.1 Object-Oriented Data Quality Model (OODQM)

The object-oriented theory has been popularized since 1970s. In this section, we build the object-oriented data quality model (OODQM) based on the object-oriented theory.

The mainly classes included in OODQM are defined as follows.

1. *Goal*— Goal is a class which represents the requests for data quality coming from users or the system. Goal contains kinds of attributes related with data quality dimensions. The Goal class includes six subclasses: *Completeness*, *Correctness*, *Usability*, *Currency*, *Consistency*, and *Relevance*, furthermore, the Completeness

subclass includes *BRC* subclass and *VAC* subclass. The meanings of these subclasses are shown in section 2. The main functions of the operations defined in these subclasses are acquiring the measures about data quality.

2. *Data Carrier*—Data Carrier is a class which represents all kinds of being forms of data and their corresponding data quality information. According to the data environments of EDW, the Data Carrier class has five subclasses: *Source DB*, *Data File*, *Temp DB*, *EDW LDM* and *EDW DB*.

*Source DB* subclass represents the databases of source systems and contains the data quality information of source systems. The main functions of the operations defined in *Source DB* are collecting the data quality information of source systems by mining the relations between all source systems or mining the business rules within a source system.

*Data File* subclass represents the data files downloaded from source system or the data files stored in the ETL server and contains their corresponding data quality information. The main functions of the operations defined in *Data File* are collecting the data quality information by checking the key features of data files, such as the names of data files, the date of data, the size of a file, the number of records in data files, the length of a record, and so forth.

*Temp DB* subclass represents the temp database (or temp space) built during the process of ETL and contains the data quality information of the temp database. Generally, the temp database has the same data structure as source data in source system. The main functions of the operations defined in *Temp DB* are collecting the data quality information by checking key features of the table in the temp database, such attributes' domain, reference integrity, data consistency within a system or between systems, correctness of indexes (by computing some sample).

*EDW LDM* subclass represents the logic data model of EDW and contains the data quality information about entities, attributes, or the relevant mapping rules from source systems to EDW. The main functions of the operations defined in *EDW LDM* are collecting the data quality information by checking the LDM.

*EDW DB* subclass represents the EDW database and its data quality information. The main functions of operations defined in *EDW DB* are the same as the operations of *Temp DB*.

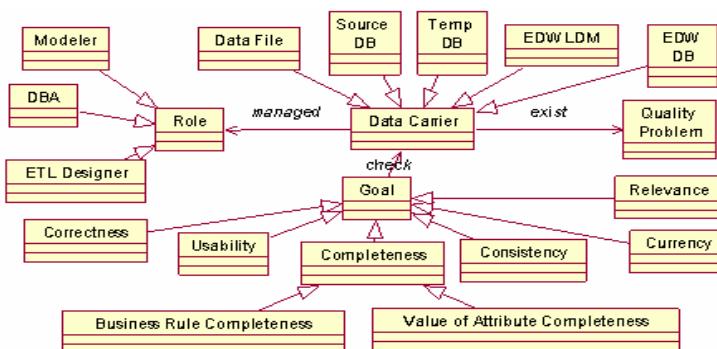
3. *Quality Problem*— Quality Problem is a class which represents all kinds of the data quality problems, such as null, incorrect data, incurrent data, inconsistent data, etc.. The main functions of operations defined in the Quality Problem are the methods used to solve corresponding data quality problems, such as the null processing methods (for example, using the sample average to replace null), noise data processing methods (for example, the binning approach), and so on. The Quality Problem can be specialized into multi subclasses according to the types of data quality problems.
4. *Role*—Role is a class which represents the participants who participate in data quality management. There are three subclasses in Role: *Modeler*, *ETL designer*, and *DBA* (database administrator). The functions of the operations defined in these subclasses are participating in data quality management. The Modeler takes charge of the correctness of LDM. The ETL designer takes charge of the data quality

management of data files, temp database, and EDW. The DBA takes charge of monitoring and maintaining the data quality in EDW when EDW running.

The mainly associations of OODQM are defined as follow:

1. “Check” between Goal and Data Carrier. Goal checks the situations of data quality of Data Carrier.
2. “Managed” between Data Carrier and Role. Data Carrier is managed by Role.
3. “Exist” between Data Carrier and Quality Problem. What problems the Data Carrier has.

The mainly generalizations of OODQM are shown in the figure 2, for example, the relations between Modeler and Role. There aren’t dependencies between the classes defined as above.



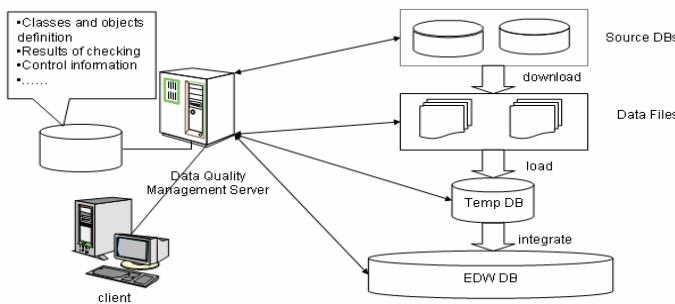
**Fig. 2. OODQM Primary Class Diagram**

Figure 2 illustrates the primary class diagram of OODQM. The other concepts of OODQM, such as massage, polymorphism, inheritance, etc., are the same as the general definitions of the object-oriented theory, and we don’t show their definitions here again.

### 3.2 Data Quality Processing in EDW

The procedure of data quality processing in EDW is a goal-driven course. We first built the goal which represents the requirements of data quality. Once the goal is built, we manage data quality by the interaction of those components of OODQM.

Figure 3 illustrates the architecture of data quality management framework of EDW. The requirements of data quality information are input by the clients and stored as Goal objects in the data quality management server. Management server checks the data quality information in the Data Carrier objects and finds out the data quality problems by comparing the information with Goal. Server informs the participants to select the right methods to solve problems. The framework has been partly implemented in our actual project. The remaining key work is complementing and optimizing the data quality checking and solving methods.



**Fig. 3.** Architecture for Data Quality Management framework of EDW

## 4 Conclusion

In this paper, we presented an object-based framework for data quality management of EDW. Our method emphasizes the importance of data quality goal, the data quality checking object, the participants, and the types of data quality problems in the process of managing data quality. The framework we built is flexible. We can trim the objects of OODQM by the actual applications.

## References

1. M. Jarke, M.A. Jeusfeld, et al.: Architecture and Quality in Data Warehouses: An Extended Repository Approach. *Information Systems* 24 (3) (1999) 229—253
2. R.Y. Wang, M.P. Reddy, H.B. Kon.: Towards quality data: an attribute-based approach. *Decision Support Systems*, 13(4) (1995) 349—372
3. Yang W. Lee.: Crafting Rules: Context-Reflective Data Quality Problem Solving. *Journal of Management Information Systems/Winter 2003-4*, Vol. 20, No. 3 (2004) 93—119
4. R.Y. Wang.: Data Quality Requirements Analysis and Modeling. Ninth International Conference on Data Engineering Vienna, Austria April (1993)
5. Strong D.; Lee, Y., Wang, R. Data quality in context. *Communications of the ACM*, 40.5 (1997) 103—110
6. Wand Y., Wang R.: Anchoring data quality dimensions in ontological foundations. *Communications of the ACM*, 39, 11 (1996) 86—95
7. Pipino L., Lee Y., Wang, R.: Data quality assessment. *Communications of the ACM*, 45, 4 (2002) 211—218
8. Wang R., Lee Y., Pipino L., Strong D.: Manage your information as a product. *Sloan Management Review*, 39, 4 (1998) 95—105
9. M. Jarke, Y. Vassiliou.: Foundations of data warehouse quality: a review of the DWQ project. In Proc. 2nd intl. Conf. Informantion Quality(IQ-97). Cambridge, Mass. (1997)
10. R.Y. Wang, V.C. Storey, C.P. Firth.: A framework for analysis of data quality research. *IEEE Trans. Knowledge and Data Engineering*, 7(4) (1995) 623-640

# Extending HPSG Towards HDS as a Fragment of pCLL

Erqing Xu

Shanghai International Studies University,  
200083 Shanghai, China  
[xuerqing@shisu.edu.cn](mailto:xuerqing@shisu.edu.cn)

**Abstract.** Rebuilding Minimalist Grammars (MG) into Categorial Minimalist Grammars (CMG) as extension of MG towards partially commutative linear logic (pCLL) is of significance. But the bijective syntax-semantics interface established in CMG sometimes fails to obtain some semantic proof trees which involves more than one quantifier. Satisfactory solution has not been found yet. Aimed at this problem, we keep the type *psoa* of HPSG (Head-Driven Phrase Structure Grammar) that the glue semantics approach removes and extend HPSG towards a HPSG Deductive System (HDS) as a fragment of pCLL the way MG are rebuilt. We handle the Subcategorization Principle, the Trace Principle, and the Semantic Principle of HPSG with HDS; establish the correspondence between syntax and semantics; obtain with this correspondence the problematic semantic proof tree, and thus solve the above-mentioned problem.

## 1 Introduction

Since Minimalist Grammars (MG) are relatively recent and largely informal, while linear logic provides a valuable common ground upon which proposals in MG can be formulated clearly, redundancies and notational variants can be identified, and compatibility with the data can have the deciding role it deserves, [4] rebuilding MG into Categorial Minimalist Grammars (CMG) as extension of MG towards partially commutative linear logic (pCLL) [1][2] is of significance.

In CMG, syntactic analyses can be regarded as pCLL proofs. Using the embedding of intuitionistic logic into pCLL [4], semantical  $\lambda$ -terms which by Curry-Howard isomorphism are intuitionistic proofs can also be viewed as pCLL proofs [4]. Thus, both the semantic homomorphic image of the syntactic analysis and the semantic recipes are pCLL proofs, and the semantic recipe of an utterance can be computed [4]. This gives rise to the establishment of a bijective syntax-semantics interface, which enables the handling of the interaction between syntax and semantics. [1][2]

But this interface sometimes fails to work for certain cases where more than one quantifier is involved. To be more specific, some semantic proof trees involving more than one quantifier are hard to obtain. For example, the semantic proof tree of “ $\exists x.[\text{poem}(x) \wedge \forall y.[\text{student}(y) \rightarrow \text{knows}(y, x)]]: t'$ ” (denoting one of the two readings of “every student knows a poem”) is hard to obtain. In order to solve this problem, the tool of CLLS (Constraint Language for Lambda Structures) is introduced, which helps to generate all the possible semantic proof trees. [2] But CLLS is too flexible

[2] and the bijective correspondence between syntactic proof trees and semantic proof trees are lost. Further work on this problem is needed. [2]

The glue semantics for HPSG (Head-Driven Phrase Structure Grammar) [5] is able to deal with multi-quantifiers and generate all the possible semantic proof trees. But the problem with the glue semantics approach for HPSG is similar to that of CLLS, i.e. it is too flexible and the correspondence between syntax and semantics cannot be found. We argue that this is because the glue semantics approach removes from HPSG the type *psoa* (parameterized state of affairs), namely the useful quantifier-handling information.

In order to solve the above-mentioned problem, the idea of this paper is to keep the type *psoa* that the glue semantics approach removes and extend HPSG towards a HPSG Deductive System (HDS) as a fragment of pCLL the way MG are rebuilt. Then the correspondence between syntax and semantics can be established and the above-mentioned problem can be solved.

## 2 A Logical Analysis of the Syntax of HPSG

In a headed phrase, the SUBCAT value of the head daughter is the concatenation of the phrase's SUBCAT list with the list of SYNSEM values of the complement daughter. [3] This is referred to as the Subcategorization Principle of HPSG.

Suppose in a HPSG schema, there are a Head-daughter and its Complement-sister, which can be designed as mere list of features. The head daughter H is  $z_1/\boxed{1}$ , and its complement-sister C is  $\boxed{1}$ , where  $z_1$  is a sublist of head features of H,  $/\boxed{1}$  is on the SUBCAT list of H, which is to be satisfied by  $\boxed{1}$ .  $\boxed{1}$  is the head features of C. We have:  $z_1/\boxed{1} \boxed{1} \vdash z_1$ . Here  $/\boxed{1}$  and  $\boxed{1}$  are both cancelled. With the operator  $/$ , we are able to attach a syntactic object to the right of another. If the head selects an additional syntactic object, it will be attached on the left, and therefore we shall have necessarily to use  $\backslash$ . We assume the usual elimination rules for  $/$  and  $\backslash$ , which are the rules of classical categorial grammars. Now we consider that syntactic features of a lexical entry or a syntactic object are put together by a non-commutative product  $\bullet$ . The rules of  $\bullet$  of HDS include [ $\bullet I$ ] and [ $\bullet E$ ] as follows respectively:

$$\frac{\Gamma \vdash x: \boxed{1} \quad \Delta \vdash y: \boxed{2}}{(\Gamma; \Delta) \vdash xy: \boxed{1} \bullet \boxed{2}}$$

$$\frac{\Gamma \vdash \omega: \boxed{1} \bullet \boxed{2} \quad \Delta[(x: \boxed{1}; y: \boxed{2})] \vdash z: \boxed{3}}{\Delta[\Gamma] \vdash \text{let } (x; y) = (\pi_1(\omega); \pi_2(\omega)) \text{ in } z: \boxed{3}}$$

Here  $\Delta[]$  denotes a position in context  $\Delta$ . [4]  $\omega$ ,  $x$ , and  $y$  are all phonological labels.  $\boxed{1}$ ,  $\boxed{2}$ , and  $\boxed{3}$  are tagged features.  $\pi_1(\omega)$  and  $\pi_2(\omega)$  are projections. ( ) expresses order. Product  $\bullet$  is also associative. We have an example with the lexicon: she ::=  $\vdash /she/: \boxed{1}$ ; walks ::=  $\vdash /walks/: \boxed{1} \backslash \boxed{2}$ . We build the syntactic proof tree with [ $\bullet E$ ]:

$$\frac{\vdash /she/: \boxed{1} \quad \vdash /walks/: \boxed{1} \backslash \boxed{2}}{\vdash /she \text{ walks}/: \boxed{2}}$$

### 3 A Logical Analysis of the Trace Principle

In HPSG, every trace must be subcategorized by a substantive head, [3] which is referred to as the Trace Principle. Refer to the following example of WUDC (weak Unbounded Dependency Construction): “I<sub>1</sub> am easy to please \_\_\_\_<sub>1</sub>.” In this example, the nominative “I” is coindexed with the accusative “\_\_\_\_”. Here the phenomenon is not so simple as it was where the hypotheses discharged in conformity with the use of the [ $\bullet$ E] rule when the factors of the product type/feature were in the same order as the hypotheses to discharge. Here relevant head features of the complement to satisfy the features on the SUBCAT list of the head are not at the right place where they can delete the features on the SUBCAT list of the head by means of our usual rules. This problem can be solved if we introduce two products  $\bullet$  (which expresses order) and  $\otimes$  (which ignores order). The rules of  $\otimes$  of HDS include [ $\otimes$ I] and [ $\otimes$ E] respectively:

$$\frac{\Gamma \vdash x: \boxed{1} \quad \Delta \vdash y: \boxed{2}}{\Gamma, \Delta \vdash \{x, y\}: \boxed{1} \otimes \boxed{2}}$$

$$\frac{\Gamma \vdash \omega: \boxed{1} \otimes \boxed{2} \quad \Delta[(x: \boxed{1}, y: \boxed{2})] \vdash z: \boxed{3}}{\Delta[\Gamma] \vdash \text{let}\{x, y\} = \{\pi_1(\omega), \pi_2(\omega)\} \text{ in } z: \boxed{3}}$$

Here {} ignores order. And then the rule [entropy] is a direct result:

$$\frac{\Gamma[(\Delta_1; \Delta_2)] \vdash \boxed{1}}{\Gamma[(\Delta_1, \Delta_2)] \vdash \boxed{1}}$$

The rule [ $\otimes$ E] can act on non-adjacent features and they can be employed to handle the trace+coindexion problem. We try the following example with the lexicon: I ::=  $\vdash /I/: \boxed{3} \otimes \boxed{5}$ ; am ::=  $\vdash /am/: \boxed{3} \setminus \boxed{2} / \boxed{4}$ ; easy ::=  $\vdash /easy/: \boxed{4} / \boxed{1}$ ; to\_please ::=  $\vdash /to\_please/: \boxed{1} / \boxed{5}$ , and build the following syntactic proof tree:

$$\begin{array}{c} \vdash /to\_please/: \boxed{1} / \boxed{5} \quad x: \boxed{5} \vdash x: \boxed{5} \\ \vdash /easy/: \boxed{4} / \boxed{1} \quad x: \boxed{5} \vdash /to\_please/x: \boxed{1} \\ \vdash /am/: \boxed{3} \setminus \boxed{2} / \boxed{4} \quad x: \boxed{5} \vdash /easy\_to\_please/x: \boxed{4} \\ y: \boxed{3} \vdash y: \boxed{3} \quad x: \boxed{5} \vdash /am\_easy\_to\_please/x: \boxed{3} \setminus \boxed{2} \\ \vdash /I/: \boxed{3} \otimes \boxed{5} \quad y: \boxed{3}, x: \boxed{5} \vdash y/am\_easy\_to\_please/x: \boxed{2} \\ \pi_1(/I)/am\_easy\_to\_please/\pi_2(/I): \boxed{2} \end{array}$$

We give the set of rules and the structure of lexical entries to express our work in this paper. Notice lexical entries are axioms. And the tagged features are in the type-logical sense. Let us call HDS (HPSG Deductive System) the fragment of pCLL [1]. We thus obtain rules ([/E], [E], [ $\bullet$ I], [ $\bullet$ E], [ $\otimes$ I], [ $\otimes$ E], [entropy], along with [ $\neg$ I] and [ $\neg$ E] in Section 4) and a lexicon of which each lexical entry consists in an axiom  $\vdash w: \mathcal{T}$  where  $\mathcal{T}$  is a type: (( $\boxed{2} \setminus (\boxed{3} \dots (\boxed{n} \setminus (\boxed{n+1} \otimes \boxed{n+2} \otimes \dots \otimes \boxed{n+m} \otimes \boxed{n+m+1})))$ )) $/\boxed{1}$ .

## 4 Semantic Interpretation

The following work in this section simplifies the complicated Semantic Principle [3] of HPSG. And we employ HDS to deal with the semantic calculus for HPSG. Refer to the example sentence: “Every student knows a poem.” [3] We have the lexicon:

every ::= $\vdash / \text{every}/: \boxed{1} / \boxed{7}$	$\lambda P. \lambda Q. \forall x. P(x) \rightarrow Q(x)$	$(e^{\text{subj}} \multimap t) \multimap (e^{\text{subj}} \multimap t) \multimap t$
student ::= $\vdash / \text{student}/: \boxed{7}$	$\lambda x. \text{student}(x)$	$e^{\text{subj}} \multimap t$
knows ::= $\vdash / \text{knows}/: \boxed{1} \backslash \boxed{3} / \boxed{2}$	$\lambda x. \lambda y. \text{knows}(x, y)$ or $\lambda y. \lambda x. \text{knows}(x, y)$	$e^{\text{subj}} \multimap (e^{\text{obj}} \multimap t)$ $e^{\text{obj}} \multimap (e^{\text{subj}} \multimap t)$
a ::= $\vdash / a/: \boxed{8}$	$\lambda P. \lambda Q. \exists x. P(x) \wedge Q(x)$	$(e^{\text{obj}} \multimap t) \multimap (e^{\text{obj}} \multimap t) \multimap t$
poem ::= $\vdash / \text{poem}/: \boxed{8} \backslash \boxed{2}$	$\lambda x. \text{poem}(x)$	$e^{\text{obj}} \multimap t$

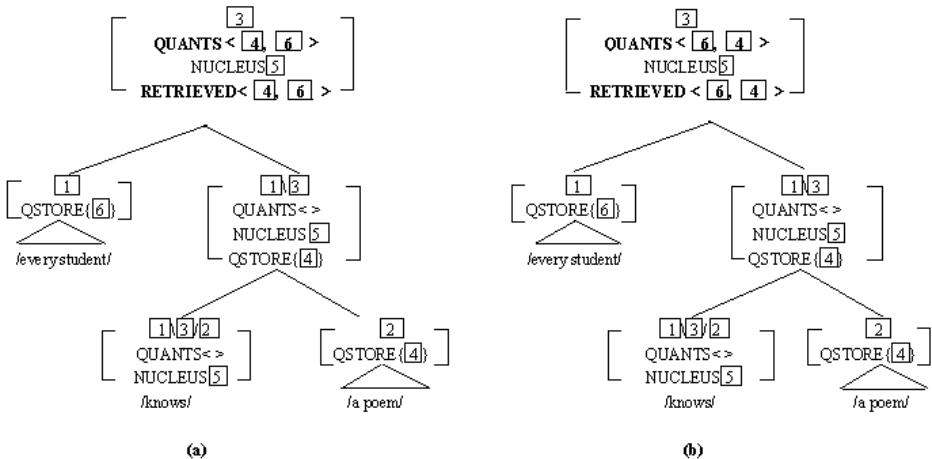
Here  $t$  bears a truth-denoting or proposition meaning, while  $e$  bears entity-denoting meanings, and  $e^{\text{subj}}$  and  $e^{\text{obj}}$  are subtypes of  $e$ . [2] The semantic rules  $[\multimap \circ I]$  and  $[\multimap \circ E]$  also belong to HDS. With  $[\backslash E]$  and  $[/ E]$  we build the syntactic proof tree:

$$\frac{\vdash / \text{every}/: \boxed{1} / \boxed{7} \quad \vdash / \text{student}/: \boxed{7} \quad \vdash / \text{knows}/: \boxed{1} \backslash \boxed{3} / \boxed{2} \quad \vdash / \text{poem}/: \boxed{8} \backslash \boxed{2}}{\frac{\vdash / \text{every student}/: \boxed{1} \quad \vdash / \text{knows a poem}/: \boxed{1} \backslash \boxed{3}}{\vdash / \text{every student knows a poem}/: \boxed{3}}}$$

In HPSG, each lexical entry contains both syntactic and semantic information in its AVM. If we highlight the semantic information as part of CONTENT whose value is of type *psoa* [3], we have two structure trees in Figure 1 [3]. The value of “knows” should be selected accordingly to ensure that the corresponding semantic proof does not crash, and we have the following two semantic proof trees for tree (a) and tree (b) in Figure 1 respectively:

$$\begin{array}{c}
 \frac{\lambda Q. \forall y. [\text{student}(y) \rightarrow Q(y)]: (e^{\text{subj}} \multimap t) \multimap t \quad z: e^{\text{obj}} \lambda y. \lambda x. \text{knows}(x, y): e^{\text{obj}} \multimap (e^{\text{subj}} \multimap t)}{\lambda y. [\text{student}(y) \rightarrow \text{knows}(y, z)]: t} \\
 \frac{\lambda Q. \exists x. [\text{poem}(x) \wedge Q(x)]: (e^{\text{obj}} \multimap t) \multimap t \quad \lambda z. \forall y. [\text{student}(y) \rightarrow \text{knows}(y, z)]: e^{\text{obj}} \multimap t}{\exists x. [\text{poem}(x) \wedge \forall y. [\text{student}(y) \rightarrow \text{knows}(y, x)]]: t} \\
 \\[10pt]
 \frac{\lambda Q. \exists x. [\text{poem}(x) \wedge Q(x)]: (e^{\text{obj}} \multimap t) \multimap t \quad z: e^{\text{subj}} \lambda x. \lambda y. \text{knows}(x, y): e^{\text{subj}} \multimap (e^{\text{obj}} \multimap t)}{\lambda y. [\text{student}(y) \rightarrow Q(y)]: (e^{\text{subj}} \multimap t) \multimap t \quad \lambda y. \text{knows}(z, y): e^{\text{obj}} \multimap t} \\
 \frac{\lambda Q. \forall y. [\text{student}(y) \rightarrow Q(y)]: (e^{\text{subj}} \multimap t) \multimap t \quad \lambda z. \exists x. [\text{poem}(x) \wedge \text{knows}(z, x)]: e^{\text{subj}} \multimap t}{\forall y. [\text{student}(y) \rightarrow \exists x. [\text{poem}(x) \wedge \text{knows}(y, x)]]: t}
 \end{array}$$

The above two semantic proof trees can be built through the rules  $[\multimap \circ I]$  and  $[\multimap \circ E]$  of HDS, and thus the complicated Semantic Principle [3] is simplified. The above two distinct semantic proof trees are obtained due to the difference in the orders of the retrievals of the stored quantifiers tagged  $\boxed{4}$  and  $\boxed{6}$ . By the above analysis, it is revealed that the sentence “every student knows a poem” has two readings.



**Fig. 1.** Two structure trees (a) and (b). In tree (a) the quantifier tagged **[6]** is RETRIEVED from QSTORE before the quantifier tagged **[4]** is RETRIEVED. And the other way round for tree (b).

## 5 Conclusion

In this paper, we extend HPSG towards HDS as a fragment of pCLL and successfully obtain certain problematic semantic proof tree that involves more than one quantifier.

## References

1. Lecomte, A.: Rebuilding MP on a Logical Ground. Research on Language and Computation. 2 (2004) 27-55
2. Amblard, M.: Synchronisation Syntax-Sémantique, des Grammaires Minimalistes Catégorielles aux Constraint Languages for Lambda Structures. RECITAL 2005, Dourdan, France, 6-10 Juin 2005
3. Pollard, C., Sag, I.A.: Head-Driven Phrase Structure Grammar. The University of Chicago Press, Chicago London (1994)
4. Rétoré, C., Stabler, E.: Generative Grammars in Resource Logics. Research on Language and Computation. 2 (2004) 3-25
5. Asudeh, A., Crouch, R.: Glue Semantics for HPSG. Proceedings of the 8<sup>th</sup> International Conference on Head-Driven Phrase Structure Grammar. CSLI, Stanford (2001) 1-19

# Chinese Multi-document Summarization Using Adaptive Clustering and Global Search Strategy

Dexi Liu<sup>1,2</sup>, Yanxiang He<sup>2</sup>, Donghong Ji<sup>3</sup>, Hua Yang<sup>2</sup>, and Zhao Wu<sup>1,2</sup>

<sup>1</sup> School of Physics, Xiangfan University, Xiangfan 441053, P.R. China

<sup>2</sup> School of Computer, Wuhan University, Wuhan 430079, P.R. China

<sup>3</sup> Institute for Infocomm Research, Heng Mui Keng Terrace 119613, Singapore  
dexiliu@gmail.com

**Abstract.** Multi-document summarization has become a key technology in natural language processing. This paper proposes a strategy for Chinese multi-document summarization based on clustering and sentence extraction. As for clustering, we propose two heuristics to automatically detect the proper number of clusters: the first one makes full use of the summary length fixed by the user; the second is a stability method, which has been applied to other unsupervised learning problems. We also discuss a global searching method for sentence selection from the clusters. To evaluate our summarization strategy, an extrinsic evaluation method based on classification task is adopted. Experimental results on news document set show that the new strategy can significantly enhance the performance of Chinese multi-document summarization.

## 1 Introduction

The sentence extraction strategy ranks and extracts representative sentences from the multiple documents. Radev et al. [1] described an extractive multi-document summarizer which extracts a summary from multiple documents based on the document cluster centroids. Although sentence extraction method is not the best one for a readable summary, some sentences extracted from the documents can describe part contents in a certain extent. Extraction-based summarization is still a promising solution especially when the speed is concerned. In the sentence extraction strategy, clustering is frequently used to eliminate the redundant information resulted from the multiplicity of the original documents [2]. However, two problems should be solved for Chinese multi-document summarization based on clustering and sentence extraction strategy. The first problem is how many clusters are appropriate for the sentences in the documents collection. In this paper, we try to use two strategies to automatically infer the cluster number: the first one makes full use of the summary length fixed by the user; the other is a stability-based strategy [3], which has been applied to other unsupervised learning problems. The second problem is how to select representative sentences from the clusters. In this paper, we give some formalization of the local and global strategies, and compare their performance.

## 2 Sentence Clustering

In this paper, two kinds of methods are proposed to detect  $K$  automatically. The first method is simple and inspired by the limit of summary length fixed by the user. On the one hand, summary length is usually fixed by the user, so the number of extracted sentences is approximatively fixed at the same time. On the other hand, to generate an anti-redundant summary, summarizer usually extracts only one sentence from each cluster. So, the number of sentences in fixed-length-summary is an acceptable value for the number of clusters. The most probable number of sentences in a fixed-length-summary is the length of summary divided by the average length of sentences in document collection. Thus, we determine the approximate number of clusters as:

$$K = \text{LSM}/\text{avg}(LS) . \quad (1)$$

Where LSM denotes the summary length fixed by the user, avg(LS) denotes the average length of sentences in the document collection.

In contrast to the former method, the second one is suitable for the condition in which the summary length is absent. We adopt a stability method, which has been applied to other unsupervised learning problems [4,5]. Formally, let  $K$  be the cluster number, we need to find  $K$  which meets (2).

$$K = \arg \max_k (F(k)) . \quad (2)$$

Where  $F(k)$  is the evaluation function based on resampling based stability. Let  $P^\mu$  be a subset sampled from full sentence set in the document collection  $P$  with size  $\delta|P|$  ( $\delta$  is set as 0.9 in this paper),  $C$  (or  $C^\mu$ ) be  $|P| \times |P|$  (or  $|P^\mu| \times |P^\mu|$ ) connectivity matrix based on the clustering result on  $P$  (or  $P^\mu$ ). Each entry  $c_{ij}$  (or  $c_{ij}^\mu$ ) of  $C$  (or  $C^\mu$ ) is calculated in the following way: if the entry pair  $p_i, p_j \in P$  (or  $P^\mu$ ) belong to the same cluster, then  $c_{ij}$  (or  $c_{ij}^\mu$ ) equals 1, otherwise, 0. Then the stability is defined as:

$$M(C^\mu, C) = \frac{\sum_{p_i, p_j \in P^\mu, p_i, p_j \in P} \eta}{\sum_{p_i, p_j \in P^\mu} \xi} . \quad (3)$$

$$\eta = \begin{cases} 1, & \text{if } \{C_{i,j}^\mu = C_{i',j'} = 1, p_i = p_{i'}, p_j = p_{j'}\}, \\ 0, & \text{else} \end{cases}, \quad \xi = \begin{cases} 1, & \text{if } C_{i,j}^\mu = 1 \\ 0, & \text{else} \end{cases}$$

Intuitively,  $M(C^\mu, C)$  denotes the consistency between the clustering results on  $C^\mu$  and  $C$ . The assumption is that if  $k$  is actually the “natural” number of the clusters, then the clustering results on the subset  $P^\mu$  generated by sampling should be similar to the clustering result on full sentence set  $P$ . Obviously, the above function satisfies  $0 \leq M \leq 1$ .

It is noticed that  $M(C^\mu, C)$  tends to decrease when increasing the value of  $k$ . Therefore to avoid the bias that small value of  $k$  is to be selected as cluster number,

we use the cluster validity of a random predictor  $\rho_k$  to normalize  $M(C^\mu, C)$ . The random predictor  $\rho_k$  achieved the stability value by assigning uniformly drawn labels to objects, in other words, splitting the data into  $k$  clusters randomly. Furthermore, for each  $k$ , we tried  $q$  times. So, the normalized object function can be defined as (4).

$$F(k) = M_k^{\text{norm}} = \frac{1}{q} \sum_{i=1}^q M(C_k^{\mu_i}, C_k) - \frac{1}{q} \sum_{i=1}^q M(C_{\rho_k}^{\mu_i}, C_{\rho_k}) . \quad (4)$$

Normalizing  $M(C^\mu, C)$  by the stability of the random predictor can yield values independent of  $k$  [4].

After the number of optimal clusters has been chosen, we adopted the k-means algorithm for the clustering phase. Each of the output sentence clusters is supposed to denote one topic in the document collection. For the sake of running efficiency, we limited cluster number  $k$  varies from 8 to 12 in the following experiments.

### 3 Representative Sentence Selection

For each sentence cluster, we need to select one sentence to represent the topic denoted by the cluster. Now that the terms extracted from the texts (sentence cluster or the whole document collection) are supposed to denote the main concepts in the texts, we weight the sentence based on the terms included in the sentence. Local search strategies select the representative sentences based on the clusters themselves. We try 3 methods to select the representative sentence: centroid sentence, TF\*IDF [6], TF.

For global search strategy, we select a sentence according to its contribution to the performance of the whole summary. To do that, we need a global criterion to measure the summary. The criterion is defined as follows:

$$w_{\text{summary}} = \frac{\sum_{t \in \text{summary}} (\log(1 + f_t^D) * \log(1 + l_t))}{\log(1 + l_{\text{summary}})} . \quad (5)$$

Where  $t$  is the term in the summary,  $f_t^D$  is term frequency in document collection,  $l_t$  is the term length. Intuitively, the criterion reflects the global term density of a summary. In general, we expect the summary to contain more terms, more longer terms, and as short as possible in each selecting step.

### 4 Experiments and Results

We adopt the extrinsic method [7] to evaluate the quality of summarization by evaluating the results of classifying task. The training and testing data set are the document collection and their summaries produced by the summarizer to be evaluated. For this classifying task, the document collection  $D$  and their summaries set  $S$  are divided into

two equal parts  $D_1$ ,  $D_2$ ,  $S_1$  and  $S_2$  respectively. The effectiveness of summarization can be evaluated through comparing the effectiveness of following 4 subtasks:

- i) S1D2: classify  $S_1$  using the classifier trained with  $D_2$ ;
- ii) S1S2: classify  $S_1$  using the classifier trained with  $S_2$ ;
- iii) D1S2: classify  $D_1$  using the classifier trained with  $S_2$ ;
- iv) D1D2: classify  $D_1$  using the classifier trained with  $D_2$ ;

The data set we used is the news pages snapped from sohu website ([www.sohu.com](http://www.sohu.com)) in 2004. All these 1912 news have been classified into 7 categories: economics (number: 312), science & technology (283), law (150), politics (409), military (256), sports (324), entertainment (178). We extract the content from these pages using XML parser and save them as text files.

Table 1 suggests objectively that the summaries can express the main idea of the original documents more accurately and fairly perfect, and the performance of classification by training with the summaries is better than that with the corresponding documents. The main reason is that most of the little-informative sentences have been removed while the themes of the original documents are covered in the summaries.

**Table 1.** Evaluation of summarization based on classification task and comparison of different sentence selection strategies (CS – Centroid Sentence ranking, TFIDF – TF\*IDF ranking, TF – Term Frequency ranking, GS – Global Search)

	Macro-averaged F <sub>1</sub>				Micro-averaged F <sub>1</sub>			
	CS	TFIDF	TF	GS	CS	TFIDF	TF	GS
S1D2	0.8045	0.8103	0.8162	0.8156	0.8127	0.8211	0.8095	0.8126
S1S2	0.9183	0.9647	0.9718	0.9630	0.9198	0.9647	0.9710	0.9605
D1S2	0.8061	0.8412	0.8433	0.8459	0.8002	0.8335	0.8562	0.8531
D1D2	0.7847	0.7847	0.7847	0.7847	0.7818	0.7818	0.7818	0.7818

The results also demonstrate that TF\*IDF ranking, term frequency ranking and the global search get very similar scores, while the centroid sentence ranking produces the summaries with lower performance. The reason may be that the information of terms frequency, their cluster frequency, as well as their length, is useful to select better representative sentences.

To check whether automatic cluster number detection help to improve the quality of summary, we design three experiments using different cluster number determination methods. Firstly, we fix the summary length as 10%, 30% and 50% of the average document length of its original document collection, and then the number of clusters can be calculated using formula (1) in section 2. Secondly, we fix the number of the clusters as 8, 10 and 12 respectively. In the third experiment, the number of the clusters automatically detected using formula (2) in section 2. For each  $K$ , we use term frequency ranking sentence selection strategy after k-means clustering, and then evaluate the generated summaries through classification task. The macro-averaged F<sub>1</sub> scores of classification result are listed in Table 2.

**Table 2.** Comparison of different cluster number determination methods

	Fixed summary length			Fixed cluster number			Auto detection
	10%	30%	50%	K=8	K=10	K=12	
S1D2	0.7526	0.8169	0.8083	0.8005	0.8106	0.7961	0.8162
S1S2	0.8209	0.9683	0.9467	0.9562	0.9437	0.9546	0.9718
D1S2	0.7314	0.8327	0.8059	0.8255	0.8370	0.8359	0.8433
D1D2	0.7847	0.7847	0.7847	0.7847	0.7847	0.7847	0.7847

We can see that the algorithm with automatic cluster number detection outperforms the fixed cluster number method. The reason is that for each document collection, the optimal number of the clusters should be different. With fixed cluster number, there would produce non-optimal cluster structures, which would affect the overall performance.

## 5 Conclusion

In this paper, we propose a cluster-based method for Chinese multi-document summarization. It mainly consists of two steps: sentence clustering and sentence selection. For sentence clustering, we propose two strategies to determine the number of clusters automatically: one strategy makes full use of the summary length fixed by the user while the other one is stability based, which can infer the optimal cluster number automatically. For sentence selection, we present a global search method and compare this method with other local methods. Experiment results show that our summarization strategy is effective and efficient for classification tasks. Our future work is to utilize more Chinese text features in sentence selection and generation.

## References

1. Radev, Dragomir R., Jing, HY., Budzikowska, M.: Centroid-Based Summarization of Multiple Documents: Sentence Extraction, Utility-Based Evaluation and User Studies. *Information Processing and Management*, Vol. 40(6) (2004) 919–938
2. Boros, E., Kantor, P., Neu, David J.: A Clustering Based Approach to Creating Multi-Document Summaries. [http://www-nlpir.nist.gov/projects/duc/pubs/2001papers/rutgers\\_final.pdf](http://www-nlpir.nist.gov/projects/duc/pubs/2001papers/rutgers_final.pdf) (2001)
3. Lange, T., Braun, Mikio L., Roth, V., Buhmann, Joachim M.: Stability-Based Model Selection. *Advances in Neural Information Processing Systems*, Vol. 15. MIT Press (2003)
4. Levine, E., Domany, E.: Resampling Method for Unsupervised Estimation of Cluster Validity. *Neural Computation*, Vol. 13 (2001) 2573–2593
5. Niu, ZY., Ji, DH., Tan, CL.: Document Clustering Based on Cluster Validation. *CIKM'04*, Washington, DC, USA (2004)
6. Baeza-Yates, R., Ribeiro-Neto, B.: *Modern Information Retrieval*. Addison Wesley, New York (1999) 27–30
7. Hand, T. F.: A Proposal for Task-based Evaluation of Text Summarization Systems. In *ACLEACL-97 Summarization Workshop* (1997) 31–36

# Genetic Algorithm Based Multi-document Summarization

Dexi Liu<sup>1,2</sup>, Yanxiang He<sup>2</sup>, Donghong Ji<sup>3</sup>, and Hua Yang<sup>2</sup>

<sup>1</sup> School of Physics, Xiangfan University, Xiangfan 441053, P.R. China

<sup>2</sup> School of Computer, Wuhan University, Wuhan 430079, P.R. China

<sup>3</sup> Institute for Infocomm Research, Heng Mui Keng Terrace 119613, Singapore  
dexiliu@gmail.com

**Abstract.** The multi-document summarizer using genetic algorithm-based sentence extraction (SBGA) regards summarization process as an optimization problem where the optimal summary is chosen among a set of summaries formed by the conjunction of the original articles sentences. To solve the NP hard optimization problem, SBGA adopts genetic algorithm, which can choose the optimal summary on global aspect. To improve the accuracy of term frequency, SBGA employs a novel method TFS, which takes word sense into account while calculating term frequency. The experiments on DUC04 data show that our strategy is effective and the ROUGE-1 score is only 0.55% lower than the best participant in DUC04.

## 1 Introduction

The sentence extraction strategy ranks and extracts the representative sentences in the multiple documents. Radev et al. [1] described an extractive multi-document summarizer that extracts a summary from multiple documents based on the document cluster centroids. Knight and Marcu [2] introduced two algorithms for sentence compression based on noisy-channel model and decision-tree approach. Barzilay et al. [3] described an algorithm for information fusion, which tries to combine similar sentences across documents to create new sentences based on language generation technologies. In sentence extraction strategy, clustering is introduced to eliminate the information redundancy resulted from the multiplicity of the original documents [4]. However, the redundancy problem cannot be totally solved because the clustering process cannot maintain the disjunction between clusters. On the other hand, the critical problem of clustering method is how to find out the appropriate number of clusters. Some researchers adopted predefined cluster number or similarity threshold. Even if the cluster number is determined, sentences with high score in their clusters may be not the best one if we view the summary as a whole. To solve these problems, we adopt genetic algorithm (GA) [5] to extract appropriate sentences in our work. In addition, a novel method is adopted while calculating TF\*IDF, which takes word sense into account.

## 2 System Design

SBGA includes three modules: pre-processing module, statistical modules and sentence extraction module. In pre-processing module, documents are split into

paragraphs and sentences, common words (stop words and other non-important words) are deleted, word stems are calculated, and word synsets are extracted from WordNet. In statistical module, word frequency, word weight, sentence position, sentence type, sentence score, sentence similarity matrix are calculated. In sentence extraction module, the best summary is chosen from the summary population, which is generated by a genetic algorithm. Pre-processing Module contains four operators: i) Split documents into paragraphs and sentences. ii) Get the stem of each word by the Porter stemmer. SBGA uses stem instead of its original word in the statistical module. iii) Get the type of each sentence according to punctuation at the end. The interrogative sentence and dialog should have less chance to appear in summary. iv) Get synset of each word from WordNet.

## 2.2 Statistical Module

Three statistical variants, word score, sentence score and sentence similarity matrix, are calculated. Let  $D$  be a document cluster with common topics, and  $|D|$  be the number of documents in  $D$ ,  $d_k$  denote the  $k^{\text{th}}$  document in  $D$ ,  $s_{i,k}$  be the  $i^{\text{th}}$  sentence in document  $d_k$ ,  $p_{j,k}$  be the  $j^{\text{th}}$  paragraph in document  $d_k$ ,  $w$  be a word,  $\text{synset}(w)$  be the synset of word  $w$ .

### (1) Word-Score

The product of “Term Frequencies (TF)” and “Inverted Document Frequencies (IDF)” is commonly used as the weight of a word [6]. However, the TF\*IDF is not accurate enough because authors prefer to use different vocabularies to express the same meaning. In this paper, not only words themselves but also their lexical meaning (according to WordNet) is employed while calculating word frequency. If a word’s synonym occurs in the same document, this word should have higher frequency. Hence, the method TFS adds the weighted synonym frequency to term (or word) frequency in order to enhance the accuracy. Weight of a synonym is calculated according to its position in the synset.

$$\text{SCORE\_W}(w) = \text{TFS}(w) \cdot \text{IDF}(w) / |D|. \quad (1)$$

Where:  $\text{TFS}(w) = \sum_{y_i \in \{\{w\} \cup \text{synset}(w)\}} \lambda^i f(y_i, D)$ ,  $y_i$  is the  $i^{\text{th}}$  element in the union set of  $w$  and its synset.  $f(y_i, D)$  is the “Term Frequencies” of  $y_i$  in document cluster  $D$ .  $\lambda^i$  is position weight of synonym ( $\lambda = 0.5$  in SBGA).  $\text{IDF}(w) = \lg \frac{\# \text{documents}}{\# \text{documents contain } w}$  is the “Inverted Document Frequencies”, which is computed over the documents in the whole corpus (we use BNC corpus in this work).

The word has higher Word-Score is more informative. We define a keyword as:

$$\text{w is a keyword, if } \text{SCORE\_W}(w) \geq 10 \times \frac{\sum_{w \in D} \text{SCORE\_W}(w)}{n_w(D)}$$

Where  $n_w(D)$  is the number of words in document cluster  $D$ . We define keyword set of sentence  $s_{i,k}$  as:  $\text{keywords}(s_{i,k}) = \{w \mid w \in s_{i,k} \wedge w \text{ is a keyword}\}$ .

## (2) Sentence-Score

Four features are used when we check whether a sentence is suitable for the summary or not. They are Word-Feature  $F_W(s_{i,k})$ , Position-Feature  $F_P(s_{i,k})$ , Length-Feature  $F_L(s_{i,k})$ , Type-Feature  $F_T(s_{i,k})$ . The Sentence-Score is defined as:

$$\text{SCORE\_S}(s_{i,k}) = w_W \cdot F_W(s_{i,k}) + w_P \cdot F_P(s_{i,k}) + w_L \cdot F_L(s_{i,k}) + w_T \cdot F_T(s_{i,k}) \quad (2)$$

Where:  $w_W, w_P, w_L, w_T$  are weights of each feature, which satisfy:

$$w_W, w_P, w_L, w_T > 0, \quad w_W + w_P + w_L + w_T = 1.$$

We let  $w_W = 0.5, w_P = 0.3, w_L = 0.1, w_T = 0.1$  in SBGA.

## (3) Sentence-Similarity-Matrix

Similarity between two words is predefined as:

$$\text{sim}_W(w, w') = \begin{cases} 0, & \text{if } |\text{synset}(w) \cup \text{synset}(w')| = 0 \\ \frac{|\text{synset}(w) \cap \text{synset}(w')|}{|\text{synset}(w) \cup \text{synset}(w')|}, & \text{else} \end{cases}. \quad (3)$$

Let  $\text{SIM\_D}$  denote the sentence similarity matrix of document cluster. The element  $\text{sim}_{i,j}$  is computed as follow:

$$\text{sim}_{i,j} = \text{sim}_S(s_i, s_j) = \begin{cases} 0, & \text{if } |\text{keywords}(s_i)| \cdot |\text{keywords}(s_j)| = 0 \\ \frac{\sum_{w \in s_i} \sum_{w' \in s_j} \text{sim}_W(w, w')}{|\text{keywords}(s_i)| \cdot |\text{keywords}(s_j)|}, & \text{else} \end{cases}. \quad (4)$$

Where  $s_i, s_j$  are the  $i^{\text{th}}$  and the  $j^{\text{th}}$  sentence in sentence collection of document cluster.

## 2.3 Sentence Extraction Module

The sentence extraction module generates a summary by GA, which starts from a random solution, and then builds, in each stage, a set of solution and evaluates them through maximizing the evaluation function.

Suppose the original document cluster has  $N$  sentences, then, each summary is represented as a vector of  $N$  bits, where “1” in the  $i^{\text{th}}$  position means that the  $i^{\text{th}}$  sentence will be extracted, and “0” means otherwise. Initially, the algorithm starts with a random set of vectors (called initial population). Each vector (called genome or individual) has  $n$  1s and  $N-n$  0s. The evaluation function of GA is defined as:

$$E(S) = w_{\text{len}} \cdot \text{LEN}(S) + w_{\text{cov}} \cdot \text{COV}(S) + w_{\text{info}} \cdot \text{INFO}(S) + w_{\text{sim}} \cdot \text{SIM}(S) \quad (5)$$

Where:

- $S$  is the candidate summary generated according to an individual in the population.
- $\text{LEN}(S)$  associated to the “length” criteria been used in [7].

- $\text{COV}(S)$  associated to the “coverage” criteria:  $|\text{keywords}(S)|/|\text{keywords}(D)|$ .
- $\text{INFO}(S)$  associated to the “informativeness” criteria:

$$\text{INFO}(S) = \sum_{s_i \in S} \text{SCORE\_S}(s_i) / \max_S \sum_{s_i \in S} \text{SCORE\_S}(s_i). \quad (6)$$

Where  $\hat{S}$  denote all of the summaries in the summary population.

- $\text{SIM}(S)$  associated to the “anti-redundancy” criteria:

$$\text{SIM}(S) = 1 - \frac{\sum_{s_i \in S} \sum_{s_j \in S} \text{sim}_S(s_i, s_j)}{n_S(S) \times (n_S(S) - 1)}. \quad (7)$$

Where  $n_S(S)$  is the number of sentences in summary  $S$ .

- $w_{\text{len}}, w_{\text{cov}}, w_{\text{info}}, w_{\text{sim}}$  are the weights of the corresponding features, which satisfy :

$$w_{\text{len}}, w_{\text{cov}}, w_{\text{info}}, w_{\text{sim}} > 0, \quad w_{\text{len}} + w_{\text{cov}} + w_{\text{info}} + w_{\text{sim}} = 1$$

$$(w_{\text{len}} = 0.2, w_{\text{cov}} = 0.3, w_{\text{info}} = 0.4, w_{\text{sim}} = 0.1 \text{ in SBGA}).$$

Applying selection, crossover and mutation operators, genetic algorithm can generate optimal summary after hundreds of iterations. In this work, the implementation of genetic algorithm is GAlib released by Mattew [8].

A summary is generated by reordering all extracted sentences according to their position in the original documents and the chronology of the original documents.

### 3 Experimentation and Evaluation

We use the document set from DUC2003 and DUC2002 as the training data, and document set from DUC2004 as the testing data. It is claimed that ROUGE-1 consistently correlates highly with human assessments and has high recall and precision in significance test with manual evaluation results [9]. So we choose ROUGE-1 as the measurement of our experiment results.

Results show that SBGA is an effective system, which score is only 0.55% lower than the best participant. By the way, the average score differences of 34 participants is 2.95%, and more than 2/3<sup>rd</sup> of these scores only have about 0.30% difference. Compared with clustering based system, SBGA gets 1.94% higher scores, which implies that the genetic algorithm based method performs better than the clustering based, and the problems such as how to determine the number of clusters are not exist.

This work also tries to check whether the sentence similarity is helpful to improve summary quality. ExtraNews [7] uses GA as well, but it does not consider the sentence similarity while designing the GA evaluation function. SBGA gets 0.62% higher score than ExtraNews. To check whether the word sense is useful for improving the accuracy of Term Frequencies or not, we also evaluate the system SBGA and the system SBGA1. SBGA uses TFS as term frequency calculating method, whereas SBGA1 uses TF. Obviously, the summarizer performs better when word sense is taken into account.

**Table 1.** ROUGE-1 scores of different systems (Human average: the average score of the human summaries; Best system: the best participant system on DUC2004; SBGA: the system using GA-based sentence extraction; ExtraNews: the participant system on DUC2004, id=19; SBGA1: the system using TF; Clustering: the clustering based system which select one sentence from each cluster; Baseline system: the baseline system which selects the beginning 665 bytes from the latest document in each document cluster)

	ROUGE-1(F-score)	Human average	SBGA
Human average	0.40441	0	+6.76%
Best system	0.37917	-6.24%	+0.55%
SBGA	0.37709	-6.76%	0
SBGA1	0.37521	-7.22%	-0.36%
ExtraNews	0.37476	-7.33%	-0.62%
Clustering	0.36978	-8.56%	-1.94%
Baseline system	0.32095	-20.64%	-14.88%

## References

1. Radev, D., Jing, HY., Budzikowska, M.: Centroid-Based Summarization of Multiple Documents: Sentence Extraction, Utility-Based Evaluation and User Studies. *Information Processing and Management*, Vol. 40(6) (2004) 919-938
2. Knight, K., Marcu, D.: Summarization Beyond Sentence Extraction: a Probabilistic Approach to Sentence Compression. *Artificial Intelligence*, Vol. 139 (1) (2002) 91–107.
3. Barzilay, R., McKeown, Kathleen R., Michael, E.: Information Fusion in the Context of Multi-Document Summarization. the 37th Annual Meeting of the Association for Computational Linguistics, New Jersey: Association for Computational Linguistics (1999) 550-557.
4. MAN A-LO`PEZ, Manuel J.: Multi-document Summarization: An Added Value to Clustering in Interactive Retrieval. *ACM Transactions on Information Systems*, Vol. 22(2) (2004) 215–241.
5. Goldberg, DE.: *Genetic Algorithms in Search Optimization and Machine Learning*. Addison Wesley, New York (1989).
6. Baeza, Yates R., Ribeiro, Neto B.: *Modern Information Retrieval*. Addison Wesley, New York (1999) 27-30.
7. Jaoua, Kallel F., Jaoua, M.: Summarization at LARIS Laboratory. <http://duc.nist.gov/pubs/2004papers/larislabs2.jaoua.pdf> (2004).
8. Matthew, W.: GAlib: A C++ Library of Genetic Algorithm Components. <http://lancet.mit.edu/ga/> (1996).
9. Lin, CY., Hovy, E.: Automatic Evaluation of Summaries Using N-gram Co-occurrence Statistics. <http://www.isi.edu/~cyl/papers/NAACL2003.pdf> (2003).

# MaxMatcher: Biological Concept Extraction Using Approximate Dictionary Lookup\*

Xiaohua Zhou, Xiaodan Zhang, and Xiaohua Hu

College of Information Science & Technology, Drexel University

3141 Chestnut Street, Philadelphia, PA 19104

xiaohua.zhou@drexel.edu, {xzhang, thu}@ischool.drexel.edu

**Abstract.** Dictionary-based biological concept extraction is still the state-of-the-art approach to large-scale biomedical literature annotation and indexing. The *exact dictionary lookup* is a very simple approach, but always achieves low extraction recall because a biological term often has many variants while a dictionary is impossible to collect all of them. We propose a generic extraction approach, referred to as *approximate dictionary lookup*, to cope with term variations and implement it as an extraction system called *MaxMatcher*. The basic idea of this approach is to capture the significant words instead of all words to a particular concept. The new approach dramatically improves the extraction recall while maintaining the precision. In a comparative study on GENIA corpus, the recall of the new approach reaches a 57% recall while the *exact dictionary lookup* only achieves a 26% recall.

## 1 Introduction

A biological concept is a unique meaning in biological domain. It represents a set of synonymous terms. For example, *C0020538* is a concept about the symptom of hypertension in Universal Medical Language System (UMLS) [13]; it represents a set of synonymous terms including *high blood pressure*, *hypertension*, and *hypertensive disease*. In comparison with individual words, a concept is more meaningful; in comparison with multi-word phrases, a concept well solves polysemy and synonymy problems [12]. Therefore, using biological concepts can improve the performance of many applications such as large-scale biomedical literature retrieval, clustering, and summarization.

There are volumes of work addressing the issue of biological concept extraction in literature. However, most of them utilize the special naming conventions or patterns to identify a few types of biological concepts such as genes, proteins and cells [1, 3, 4, 7, 8, 9, 10]. In general, those approaches are designed for very specific types of concepts, and work efficiently and effectively if the types of biological concepts have unique naming patterns. Many large-scale biomedical applications such as literature retrieval, clustering, and summarization, however, are interested in many rather than a few types of biological concepts most of which do not have unique naming patterns.

\* This research work is supported in part from the NSF Career grant (NSF IIS 0448023). NSF CCF 0514679 and the research grant from PA Dept of Health.

For example, UMLS covers 135 semantic types of biological concepts; a typical genomic IR system will index all of them.

The dictionary-based biological concept extraction is still the state-of-the-art approach to large-scale biomedical literature annotation and indexing [6, 11, 12]. Its major advantage over the pattern-based approach is that it not only recognizes names, but also identifies unique concept identities. Among dictionary-based approaches, the *exact dictionary lookup* is the simplest one, but always achieves low extraction recall because a biological term often has many variants such as morphological variants, syntactic variants, and semantic variants [2] while a dictionary is impossible to collect all of them.

In this paper, we propose a new approach, referred to as *approximate dictionary lookup*, to the biological concept extraction. The basic idea is to capture the significant words rather than all words of a concept. For example, the word *gyrb* is significant to the concept “*gyrb protein*”; we will recognize it as a concept name even if the word *protein* is not present. Using UMLS Metathesaurus [13] as the dictionary, we implement this approach as an extraction system called *MaxMatcher*. We test the new approach on GENIA corpus [14]. As expected, the new approach dramatically raises the recall from 26% to 58%.

## 2 The Concept Extraction Approach

To overcome the limitation of *exact dictionary lookup*, we introduce an approximate dictionary lookup technique. The basic idea of this technique is to capture significant words rather than all words in a concept name. For example, the word *gyrb* is obviously very significant to the concept “*gyrb protein*”; we treat it as a concept name even if the word *protein* is not present. So the problem is reduced to measuring the significance of any word to given concept names. In particular, we propose a relative significance score measure in this paper. Suppose a concept ( $c$ ) has  $n$  concept names denoted as  $s_1, \dots, s_n$ , respectively. Let  $N(w)$  denotes the number of concepts whose variant names contain word  $w$ , and let  $w_{ji}$  denotes the  $i$ -th word in the  $j$ -th variant name of the concept, the significance of  $w$  to the concept is defined as follows:

$$I(w, c) = \max\{I(w, s_j) \mid j \leq n\} \quad (2.1)$$

where :

$$I(w, s_j) = \begin{cases} 0 & w \notin s_j \\ \frac{1/N(w)}{\sum_i 1/N(w_{ji})} & w \in s_j \end{cases}$$

We use UMLS Metathesaurus 2005AA version [13] as the dictionary to train the significance score of each word to biological concepts containing that word. The UMLS Metathesaurus has a table called normalized string index, which record all normalized names of each concept. We remove normalized strings containing more than ten words and then use the remaining 2,573,244 strings to build the significance score matrix. A huge matrix, 509,170 rows (words) by 998,774 columns (concepts), is

```

Find next starting word  $t_s$ 
 $k = 0$ 
 $C = \{c \mid t_s \in T(c)\}$  /*  $T(c)$  is the set of words appearing in names of concept  $c$  */
For each  $c \in C$   $S_c = I(t_s, c)$  /*  $I(t_s, c)$  is the score of word  $t_s$  to concept  $c$  */
While next word  $t$  is not boundary word AND  $k < skip$ 
     $N = \{c \mid t \in T(c) \wedge c \in C\}$ 
    IF  $N = \emptyset$  Then  $k = k + 1$ 
    Else
         $C = N$ 
        For each  $c \in C$   $S_c = S_c + I(t, c)$ 
    End If
Wend
 $C = \{c \mid S_c > threshold \wedge c \in C\}$ 
If  $|C| > 0$  Then
    return concept name and candidate concepts  $c \in C$ 
End If

```

**Fig. 1.** The algorithm for extracting one concept name and its candidate concept IDs. The *threshold* is set to 0.95; the maximum number (*skip*) of skipped words is set to 1.

obtained. Because for each word, only a few concepts contain it, we use sparse matrix to make the storage and search more efficiently.

During the stage of extraction, we use a set of simple rules to identify the boundary of a concept candidate. A biological concept name should begin with a noun, a number, or an adjective while ending with a noun or a number; it can not contain any boundary words including (1) punctuations (except hyphen, period, and single quote), verbs, and conjunctions and prepositions (except “of”). In other words, whenever a boundary word is encountered, a candidate concept name reaches its end. The detailed searching algorithm is shown in Figure 1.

The major advantage of *approximate dictionary lookup* is that even if a concept name changes the word ordering a little bit, inserts or deletes a couple of insignificant words, it is still can be recognized. According to its definition, the significance score of a concept name should be equal to or greater than 1.0 if no word is missing. Thus, the threshold of significance score should be close to 1.0. If the threshold is too small, our approach may falsely recognize “*high pressure*” as the concept name “*high blood pressure*”; if it is too high, our approach may fail to recognize “*gyrb*” as “*gyrb protein*”. We found that 0.95 as the threshold gave good results for UMLS-based biological concept extraction. Our approach is able to recognize concept names with a couple of insertions such as articles, pronouns, and even nouns. The parameter *skip* controls the maximum number of insertions. We found that *skip*=1 gave good results.

The searching results are concept names and corresponding concept IDs. If two or more concept IDs are returned, we need to further figure out the meaning the extracted concept name refers to. The words surrounding the extracted concept name are often indicative to the meaning [5]. Thus, we take surrounding words (4 to the left and 4 to the right) as the context and use the same algorithm as shown in Figure 1 to disambiguate the meaning of the extracted concept name if necessary.

### 3 Experimental Results

We evaluate both efficiency and effectiveness of the *MaxMatcher*. The effectiveness is evaluated on GENIA 3.02 corpus [14] which consists of 2,000 human annotated PubMed abstracts. We compare the result of *MaxMatcher* with that of two other *exact dictionary lookup* systems, *BioAnnotator* [8] and *ExactMatcher*. ExactMatcher is implemented by us. The machine-extracted terms are compared with human annotations. Because human annotation is kind of subjective, we provide exact-match based evaluation and approximate-match based evaluation, following the evaluation method in [8]. For approximate-match, the human annotation should be the substring of the machine annotation, or the opposite.

The comparison among three systems is presented in Table 1. For exact-match, *MaxMatcher* performs significantly better than the other two systems in terms of both precision and recall. For approximate match, the precision of *MaxMatcher* is comparable to that of the other two systems while the recall is significantly better than that of the other two.

**Table 1.** The effectiveness comparison. BioAnnotator [8] actually tested several configurations. But only the configuration with only dictionaries (i.e. exact dictionary lookup) is compared. BioAnnotator was evaluated on GENIA 1.1 (containing 670 human annotated abstracts of research papers). The dictionary used for BioAnnotator also includes LocusLink and GeneAlias in addition to UMLS.

IE Systems	Exact Match Eva.			Approximate Match Eva.		
	Recall	Precision	F-score	Recall	Precision	F-score
MaxMatcher	57.73	54.97	56.32	75.18	71.60	73.35
ExactMatcher	26.63	31.45	28.84	61.56	72.69	66.66
BioAnnotator	20.27	44.58	27.87	39.75	87.67	54.70

For efficiency comparison, we download first 10,000 PubMed abstracts published in 2005 and count the time for annotating these abstracts by *MaxMatcher* and *ExactMatcher*, respectively. It takes 510 seconds for *MaxMatcher* to annotate all 10,000 PubMed abstracts; the average annotation speed is 19.6 abstracts per second. *ExactMatcher* is faster. It only costs 320 seconds to process those abstracts; the average annotation speed is 31.3 abstracts per second. However, *ExactMatcher* consumes much more memory (765Megabytes) than *MaxMatcher* (362 Megabytes).

### 4 Conclusions

Dictionary-based biological concept extraction is still the state-of-the-art approach to the large-scale biomedical literature annotation and indexing. The *exact dictionary lookup* is very simple but always achieves low extraction recall because biological terms often have many variants while a dictionary is impossible to collect all of them. In this paper, we propose a generic approach, referred to as *approximate dictionary lookup*, to cope with the biological concept variation. The basic idea of the new approach is to capture the significant words of a biological concept rather than all of

them. A comparative study on GENIA corpus shows that the new approach can dramatically improve the extraction recall while maintaining the precision. However, the extraction efficiency of the new approach goes down a little bit in comparison with the *exact dictionary lookup*.

## References

1. Chang, J.T., Schütze, H., and Altman, R.B., "GAPSCORE: finding gene and protein names one word at a time", *Bioinformatics*, Vol. 20, No. 2, pp. 216-225, 2004.
2. Chiang, J.-H. and Yu, H.-C., "Literature extraction of protein functions using sentence pattern mining", *IEEE Transactions on Knowledge and Data Engineering*, 17(8), Aug. 2005 Page(s):1088 – 1098
3. Collier, N., Nobata, C., and Tsujii, J., "Extracting the names of genes and gene products with a Hidden Markov Model", *Proc. COLING 2000*, 201--207, 2000
4. Fukuda, K., Tamura, A., Tsunoda, T., and Takagi, T., "Toward information extraction: Identifying protein names from biological papers", In *Proceedings of Pacific Symposium on Biocomputing*, pages 707--718, Maui, Hawaii, January 1998.
5. Lesk, M., "Automatic Sense Disambiguation: How to Tell a Pine Cone from and Ice Cream Cone", *Proceedings of the SIGDOC'86 Conference*, ACM, 1986.
6. Rindfleisch, T.C., Tanabe, L., and Weinstein, J.N., "EDGAR: Extraction of Drugs, Genes and Relations from the Biomedical Literature", *Proceedings of Pacific Symposium on Bioinformatics*, Hawaii, USA, pp. 514-525, 2000.
7. Song, Y.-I., Kim, S.-B., and Rim, H.-C., "Terminology Indexing and Reweighting methods for Biomedical Text Retrieval", In *Proceedings of the SIGIR'04 Workshop on Search and Discovery in Bioinformatics*, Sheffield, UK, ACM, July 2004.
8. Subramaniam, L., Mukherjea, S., Kankar, P., Srivastava, B., Batra, V., Kamesam, P. and Kothari, R., "Information Extraction from Biomedical Literature: Methodology, Evaluation and an Application", In *the Proceedings of the ACM Conference on Information and Knowledge Management*, New Orleans, Louisiana, 2003.
9. Tanabe, L. and Wilbur, W., "Tagging gene and protein names in biomedical text", *Bioinformatics*, Vol. 18, No. 8, pp.1124-1132, 2002.
10. Zhou, G.-D., Zhang, J., Su, J., Shen, D., and Tan, C.-L., "Recognizing Names in Biomedical Texts: A Machine Learning Approach", *Bioinformatics*, 20(7), 1178-1190, 2004.
11. Zhou, X., Han, H., Chankai, I., Prestrud, A., and Brooks, A., "Converting Semi-structured Clinical Medical Records into Information and Knowledge", *Proceeding of The International Workshop on Biomedical Data Engineering (BMDE) in conjunction with the 21st International Conference on Data Engineering (ICDE)*, Tokyo, Japan, April 5-8, 2005.
12. Zhou, X., Hu, X. and Zhang, X., "Using Concept-based Indexing to Improve Language Modeling Approach to Genomic IR", *The 28th European Conference on Information Retrieval (ECIR' 2006)*, 10 - 12 April, 2006, London, UK.
13. UMLS, <http://www.nlm.nih.gov/research/umls/>
14. GENIA Corpus, <http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA/>

# Bootstrapping Word Sense Disambiguation Using Dynamic Web Knowledge

Yuanyong Wang and Achim Hoffmann

School of Computer Science & Engineering  
The University of New South Wales  
Sydney, Australia  
[wyw@cse.unsw.edu.au](mailto:wyw@cse.unsw.edu.au),  
[achim@cse.unsw.edu.au](mailto:achim@cse.unsw.edu.au)

**Abstract.** Word Sense Disambiguation(WSD) is one of the traditionally most difficult problems in natural language processing and has broad theoretical and practical implications. One of the main difficulties for WSD systems is the lack of relevant knowledge—commonly known as the knowledge acquisition bottleneck problem. We present in this paper a novel method that utilizes dynamic Web data obtained through Web search engines to effectively enrich the semantic knowledge for WSD systems. We demonstrated through a word sense disambiguation system the large quantity and good quality of the extracted knowledge.

## 1 Introduction

Word sense disambiguation problem is a problem in which the proper sense of a target word has to be determined in a given context. “All disambiguation work involves matching the context of the instance of the word to be disambiguated with either information from an external knowledge source (knowledge driven WSD), or information about the contexts of previously disambiguated instances of the word derived from corpora (data-driven or corpus-based WSD)”(Ide and Veronis,1998) [4]. Despite this seemingly simple principle the development in the field of word sense disambiguation over the years has always had difficulty obtaining enough prior knowledge whether it be “external knowledge source” or annotated corpora.

Recently, due to the enormous data available on the Web many researchers are drawn to the Web for a way of solving the knowledge acquisition bottleneck problem. Many statistical word sense disambiguation models are built on huge corpora collected from the Web (Turney,2004) [5]. Web search engines like Altavista usually have several billions of web pages indexed. The collection of all the Web pages if treated as a single corpus and the search engine as a tool to draw statistical information from this corpus offer a very encouraging solution to the problems caused by hand tagging and data sparseness for corpus based approaches. Our work is along the line of exploring the enormous potential of Web data and search engines as a gateway to this rich resource for WSD.

This paper is organized as follows. In section 2 we describe the proposed WSD model that bootstraps from minimum initial human input by extracting Web data through search engines. In section 3 the model will be evaluated on 35 selected nouns

and the results presented. Comparison with relevant works is conducted in section 4. Finally, in section 5 conclusions and a discussion about future development are given.

## 2 Our WSD Model

- **Step 1: Seeding**

The first step is called *seeding* during which a human is asked to nominate three typical phrases or "seed phrases" for each of the senses to be disambiguated for a target word. For example, for the target word "age" two senses are considered. The first sense refers to "a period in history". The second sense refers to "duration of life". Three seed phrases for the first sense are: "stone age", "digital age", "new age". For the second sense the seed phrases are: "old age", "age discrimination" and "at age". As shown above the seed phrases could be of any type (different types of noun phrases, prepositional phrases) as long as they are highly specific to only one of the senses disambiguated and contain the target word.

- **Step 2: Bootstrapping**

The second step is called *bootstrapping*. Each of the seed phrases are issued to the Web search engine Altavista as quoted queries. The snippets containing target words are extracted from the retrieved pages. Each snippet usually contains 30 to 40 words. For each seed phrase we automatically download approximately 1,000 such snippets. The snippets are pooled together for each sense so that each sense would have a snippet pool of 3,000 snippets. This way, the information contained in the three seed phrases specific to one sense is dramatically expanded to a sense specific snippet pool with texts of more than 10,000 words.

- **Step 3: Statistical knowledge extraction**

In the third step- *statistical knowledge extraction*, the statistical information is collected from the snippet pools on single word occurrences and 2-gram occurrences. For each sense we have two lists compiles. One list contains all the words that occur in its snippet pool ranked according to their frequency and the other list contains all the 2-gram phrases ranked likewise. All the functional words like "the", "a" ... would be removed from the word list. Then all the phrases that do not contain the remaining words in the word list would also be removed from the corresponding phrase list. The word list is thus used to filter the phrase list of semantically irrelevant phrases. The different phrase lists are also used to do mutual pruning, where one phrase that is present in different phrase lists with significantly different frequencies are removed from the list where its frequency is low. Eventually only the pruned phrase lists will be used.

- **Step 4: Disambiguation**

The fourth step is called *disambiguation*. As the name suggests, this is where the actual sense disambiguation is performed on the test data. Each test case is a whole news article that contain the target word in its title (and most of the time in its body as well) from Google News (as in news.google.com). The overlap found between sense specific phrase lists and the news article will be counted. One overlap is defined to be one 2-gram phrase that occurs both in the phrase list and in the news article. Basically the one sense specific phrase list that has more overlap with the news article indicates the correct sense of the target word.

- Step 5: **Second bootstrapping**

The fifth step is called *second bootstrapping*. It is thus named because the model does bootstrapping again from the knowledge extracted in the third step in the hope of further expanding sense specific information. Similar to the first *bootstrapping* new seed phrases need to be first determined. The top phrases in expanded phrase lists will undergo the disambiguation process(step4) just like a test case. As a result of the *disambiguation* some phrases will be ruled out as noisy phrases. From the rest phrases the first three will be selected. They are called “second seed phrases” and will be used as seed phrases for next round of expansion. For the ”age” example, the size of the sense specific phrase list for sense one increased from 24608-phrase to 41472-phrase to 57777-phrase after two iterations. It increased from 13267-phrase to 27154-phrase to 40112-phrase for sense two.

### 3 Evaluation

The WSD model is evaluated on a data set of 35 nouns. Among the 35 nouns 29 nouns are of two-way distinction, 5 nouns are of three-way distinction and 1 noun is of 4-way distinction. All the disambiguated senses are subsets of WordNet senses for these target words. These nouns are chosen so that their senses have substantial presence in the Google News articles. The nouns are summarized below.

- **2-way distinction nouns:** party, tissue, atmosphere, image, mouse, drug, deposit, age, degree, nature, player, rally, toll, stage, plant, yard, mine, bass, organ, treatment, offense, head, trial, memory, body, bug, power, room, trunk
- **3-way distinction nouns:** paper, channel, charge, course, heart
- **4-way distinction noun:** interest

The experiment results are summarized in Table 1. Precision and applicability measures are chosen to present the results because they are more informative than the com-

**Table 1.** Summary of the results over two iterations. ”#S” is the number of senses disambiguated. ”#N” is the number of nouns under the category. ”ave-prec” is average precision. Precision is the proportion of correct judgement in all the judgements made. ”ave-app” is average applicability . Applicability is the proportion of all the judgements made in all the test cases. ”iter-n” is the nth iteration of the algorithm. Each iteration consists of all 5 steps of the algorithm.

#S	#N	measure	iter-1	iter-2	default
2	29	ave-prec	92.8%	94.3%	87.2%
		ave-app	70.8%	76.1%	100%
3	5	ave-prec	81.4%	90%	77%
		ave-app	61.6%	64.3%	100%
4	1	ave-prec	74%	79.6%	57.2%
		ave-app	72.5%	67.5%	100%
	total	ave-prec	90.8%	93.4%	84.9%
		ave-app	69.5%	74.4%	100%

monly used precision/recall measure. At the second iteration the results of 93.3% precision and 74.4% applicability are comparable to 70% precision/recall. When a WSD system gives a judgement for every case its precision equals to its recall. Our system, if assumed to give judgement for every case (non-judgement cases are considered in correct), has 69.4%(multiply 93.3% by 74.4%) precision/recall. The results in the last column are simply obtained by adding the default sense heuristics to the second iteration disambiguation. Default senses that are predetermined by prior sense distribution statistics are assigned to those non-judgement cases.

## 4 Comparison with Related Works

(Yarowsky,1995) [1] as one the pioneer in unsupervised WSD systems designed a model featuring one sense per discourse and bootstrapping algorithm to expand the sense specific collocations. Ten words are tested in his experiment all with 2-way distinction. Precision as high as 96% has been achieved but no information about applicability is given. In this work the collocation is treated as a bag of words. In comparison, our model uses phrases instead of words to avoid the ambiguity among the collocational words themselves. In his work Yarowsky also mentioned the fundamental limitation of the model is coverage and in half of the examples no overlap is found between sense specific information and the collocational contexts. In our work, because of the use of Web search engine to collect highly concentrated sense specific information the data sparseness problem is greatly reduced. Carroll and (McCarty et al,2000) [3] in their unsupervised system for Senseval-2 WSD competition automatically extract subject-verb and verb-direct object dependencies from a 90 million word corpus as selectional preference evidence for WSD. They achieved 69.1% precision and 20.5% recall for the all word task. The low recall is an apparent indication of the low coverage of the automatically extracted selectional preference information. The 90 million word corpus, despite being large in its overall size, is not necessarily large at all for a randomly chosen word. This is still the notorious data sparseness problem that troubles most corpus based approaches. Agirre and (Martinez,2004) [2] also used Google snippets to compile Web corpus for their minimally supervised WSD model. Their model tested on 29 nouns from the Senseval-2 competition achieved 49.8% precision/recall. The seeds they use are monosemous words from WordNet. This, while eliminating the possible ambiguity from the seeds, again seriously limited the coverage of the sense specific information automatically extracted from the Web.

(Mihalcea et al,1999) [7] designed a simple and elegant WSD algorithm that utilizes WordNet glosses and word pairs from the disambiguation contexts to produce sense specific queries. The search engine hits of these queries are used as evidence to determine the correct sense. This model achieved a 80.1% precision over 384 word pairs manually extracted from Brown Corpus. No information about applicability is given. Many more WSD systems use Web in a different way and compile huge text corpora out of Web texts. (Turney,2004) [5] used a system that utilize syntactic and semantic feature vectors trained on the training data for disambiguation. Co-occurrence statistics extracted from a huge Web corpora are used to assist similarity score computation, which in turn helps assigning values to the semantic feature vectors. His fully

supervised system when applied to Senseval-3 lexical sample task achieved 75.9% precision/recall. (Mihalcea,1999) [6] proposed a model that utilizes WordNet synsets and definitions as seeds and bootstraps by retrieving through search engines Web documents with these seeds as the queries. The retrieved documents are manually tested for their relevancy. The model is limited in that it is still largely word oriented (its limitation in terms of possible high ambiguity in the seeds is also mentioned by the author).

## 5 Conclusion and Future Work

By comparing the experimental results of our proposed WSD model with related works, our approach appears very encouraging indeed. The proposed model, being minimally supervised, produced a comparable or favorable results to the best contemporary unsupervised and minimally supervised systems that attack tasks like Senseval-3 lexical sample task and similar tasks reported in other works. Using only phrases as seeds for bootstrapping and sense specific phrases for disambiguation should contribute the most to the high precision in our results. Using Web search engine to download large amounts of sense specific snippets as well as the snow balling effect of the iterative bootstrapping (with the assistance of first sense heuristics) are the determining factors of the high applicability in the results. One of the aspects of the model that apparently need to improve is the seed phrase selection process. We will explore more techniques to this end and to improve other aspects of our model in the future.

## References

1. Yarowsky. David. Unsupervised word sense disambiguation rivalring supervised methods. In *Proceedings of the 33rd Meeting of the Association for Computational Linguistics*, 1995.
2. Agirre. E and Martinez. D. Unsupervised wsd based on automatically retrieved examples: The importance of bias. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Barcelona, Spain, 2004.
3. Carroll. J and McCarthy. D. Word sense disambiguation using automatically acquired verbal preferences. In *Computers and Humanities*, pages 34,1–2, Netherlands, 1999.
4. I. Nancy and V. Jean. Word sense disambiguation: The state of the art. In *Ide, Nancy and Vronis, Jean (1998). Word sense disambiguation: The state of the art. Computational Linguistics, 24:1, 1-40.*, 1998.
5. Turney. D Peter. Word sense disambiguation by web mining for word co-occurrence probabilities. In *Proceedings of the 3rd International Worshop on the Evaluation of Systems for the Semantic Analysis of texts*, pages 239–242, Barcelona, Spain., 2004.
6. Mihalcea. Rada and Moldovan. Dan. An automatic method for generating sense tagged corpora. In *Proceedings of the American Association for Artificial Intelligence*, Orlando, FL, 1999.
7. Mihalcea. Rada and Moldovan. Dan. A method for word sense disambiguation of unrestricted text. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, College Park, MA., 1999.

# **Automatic Construction of Object Oriented Design Models [UML Diagrams] from Natural Language Requirements Specification**

G.S. Anandha Mala<sup>1</sup> and G.V. Uma<sup>2</sup>

Department of Computer Science and Engineering, College of Engineering,  
Anna University, Guindy, Chennai, Tamil Nadu, India-600025  
malamanosuke@yahoo.co.in, gvuma@annauniv.edu

**Abstract.** Application of natural language understanding to requirements gathering remains a field that has only limited explorations so far. This paper presents an approach to extract the object oriented elements of the required system. This approach starts with assigning the parts of speech tags to each word in the given input document. Further, to resolve the ambiguity posed by the pronouns, the pronoun resolutions are performed before normalizing the text. Finally the elements of the object-oriented system namely the classes, the attributes, methods and relationships between the classes, sequence of actions, the use-cases and actors are identified by mapping the ‘parts of speech-tagged’ words onto the Object Oriented Modeling Language elements using mapping rules which is the key to a successful implementation of user requirements.

## **1 Introduction**

As already several attempts have been made to semi automate the process of requirements capture, this is yet another approach of automatic construction of object oriented design model [UML diagram] from the natural language requirement specification. The paper begins with a review of the advances in the field of requirements engineering in section 2. The proposed methodology is explained in section 3. Our implementation and results are explained in section 4. The conclusion and future work is contained in Section 5.

## **2 Related Work**

The first relevant published technique attempting to produce a systematic procedure to produce design models from NL requirements was Abbot [1]. Abbot suggested a non-automatic methodology that only produces static analysis and design products obtained by an informal technique requiring high participation with that of users for decisions. Methods to bring out a justified relationship between the natural-language structures and OO concept is proposed by Sylvain [9] who show that computational linguistic tools, are appropriate for preliminary computer assisted OO analysis. Sawyer in their REVERE [5] makes use of a lexicon to disambiguate the word senses thus obtaining a summary of requirements from a natural language text but do not

attempt to model the system. Liwu Li [6] also presents a semi-automatic approach to translate a use case to a sequence diagram. It needs to normalize a use case manually. Overmyer [8], also present only a complete interactive methodology and prototype. However, the text analysis remains in good part a manual process. Liu [3] present an approach, which uses formalized use cases to capture and record requirements. Ke Li [4] also semi-automate the process of requirement elicitation where the text is matched with predefined statements. If there is no match then get help from user to clarify incomplete/ambiguous data. Participation of domain experts, customer are needed in class identification process in contrast to our fully automatic methodology.

### 3 The Proposed System

In all the earlier works mentioned, the requirement elicitations are not fully automatic. The proposed methodology includes the automatic reference resolution, which eliminates the user intervention as in the previous works. The system architecture is shown in fig. 1. The system named as ‘REQUIREMENTS ELICITOR’. The given input problem statement is split into sentences by the sentence splitter for sentence tagging. Then each sentence is subjected to tagging in order to get the parts of speech marker for every word. The noun and the verb phrases are identified for the tagged text by chunker based on simple phrasal grammars. To remove ambiguity posed by pronouns, they are resolved to their respective noun phrases by reference resolver. The text has to be simplified into the following constructs by the normalizer to ease the task of mapping the words onto the Object Oriented system constituents.

- Conditional: Conditional syntax is *If* aCondition transaction [*else* other Transitions]
- Iteration: Iteration syntax is *While* condition transactions *endwhile*.
- Concurrency: Concurrency syntax is

*Start concurrency* transaction 1 ... *concurrent* transaction k  
*end concurrency*

which executes transaction l, to transaction k concurrently.

- (Synchronization) A synchronization syntax is

*Start Synchronization* transaction 1 ... *synchronized* transaction k  
*end Synchronization*

Which requires synchronize transaction l, to transaction k.

All the transaction statements are simple. A number of patterns using conjunctions and their corresponding splits in the sentences are stored in the catalog. Each sentence is checked against the stored patterns and the corresponding split up is made.

For example “If the source and the destination of the request fall on the same route, the receptionist checks the seat that are available and issues the ticket to the passenger and blocks the seat” is normalized to

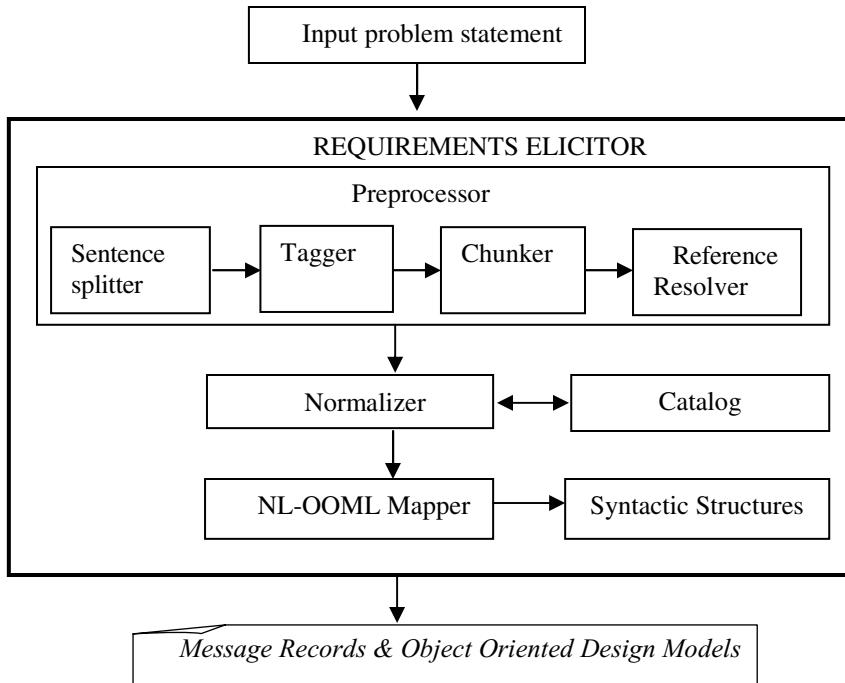
If the source and the destination of the request fall on the same route

The receptionist checks the seat.

The receptionist issues the ticket to the passenger

The receptionist blocks the seat

End if



**Fig. 1.** System Architecture

The NL-OOML mapper accepts a normalized problem description as input. Using the syntactic structures as in table 1. it translates each normalized sentence into a message record. Simple rule based approach is followed for identifying OO elements. The rules are,

- 1: Translating Nouns to Classes. A noun, which does not have any attributes, need not be proposed as a class.
- 2: Translating *Noun-Noun to Class-Property* according to position. When two nouns appear in sequence in the text, the first Noun is translated to Class and the following Noun is translated to properties of this Class.
- 3: A simple heuristic is used to decide which nouns are classes, and which form the attribute. In Noun-Noun, if the first noun is already been chosen as the class then the second noun is taken as the attribute. The attributes are decided based on the verb phrase.
- 4: Translating the lexical Verb of a non-personal noun to a Method of this noun. Decide the sender, receiver classes and argument to this method based on the Table 1.
- 5: Translating the lexical Verb of a personal noun to a use case (or part of a use case) linked with an actor defined by this noun.
- 6: Matching a Noun to a Personal Pronoun as the nouns of previous sentence.

**Table 1.** Syntactic Structures of Simple Sentences

Syntactic Structure	Sender	Receiver	Action	Argument
subject verb object	subject	object	verb	-
subject verb object (to) verb1 (object1)	subject	object	verb1 (+object1)	(object1)
subject verb object participle (object1)	subject	object	participle verb(+object1)	(object1)
subject verb object adjective	subject	object	be + adjective	
subject verb object conjunctive to verb1 (object1)		subject	verb	object,verb1 (+object1)
subject verb gerund (object)		subject	verb	gerund verb (+object)
subject verb object preposition object1	subject	object1	verb + object	(object)
subject verb object object1	subject	object	verb	object1
subject verb (for) complement		subject	verb	complement
subject verb		subject	verb	
subject be predicative		subject	be + predicative	
subject verb preposition object		subject	verb+preposition	object

## 4 Implementation and Results

The ‘Requirements Elicitor’ was implemented using JAVA and validated using 100 problem samples each of around 500 lines. The result produced by the system was compared with that of the human output. The human outputs were the results that were obtained by conducting the noun-verb analysis on the text. It was considered as the baseline and taken as expert judgment. The system does not miss to identify any of the classes and methods. But approximately 12.4% of additional classes and 7.4% of additional methods are identified in the entire sample taken, those that are removed by human by intuition that they may not be classes. Since system lacks that knowledge, they are listed as classes. The missed out methods occur only if the tagger assigns a wrong tag to the word. Also the system perfectly identifies all the attributes, usecases and actors with out any additional, missed or miss assignments.

## 5 Conclusion and Future Work

The project presents an approach to restructure the natural language text into a modelling language in order to elicit the stated requirements of a system. Further the work can be extended for identifying the different modules present in the requirement specification by properly segmenting the input text, which will help us to identify the packages. The deficiencies in the tagger and the reference resolver can be overcome by building a knowledge base which can also improve the effectiveness of generation of the system elements.

## References

1. Abbot.R.J.: "Program Design by informal English descriptions". *Communications of the ACM*, vol.26, (1983) 882 – 894.
2. Brill E.: "A simple rule-based part-of-speech tagger". *Proceedings of Third ACL Conference on Applied Natural Language Processing*, Trento, Italy, (1992) 152-155
3. Dong Liu, Kalaivani Subramaniam, Behrouz H. Far, Armin Eberlein: "Automating transition from use cases to class model", *MSc Thesis*, University of Calgary, (2003).
4. Ke Li: "Towards Semi-automation in Requirements Elicitation: mapping natural language and object-oriented concepts", *13th IEEE International Requirements Engineering Conference*, (2005)
5. Sawyer P., P Rayson, and R Garside: "REVERE: support for requirements synthesis from documents", *Information Systems Frontiers Journal*. Vol.4, (2002) 343 - 353.
6. Liwu Li: "A semi-automatic approach to translating use cases to sequence diagrams", *Proceedings of Technology of Object-Oriented Languages and Systems*, July (1999), IEEE CS Press, 184 –193
7. Mitkov. R: "Robust pronoun resolution with limited knowledge", Proceedings of the 18.th International Conference on Computational Linguistics (COLING'98)/ACL'98", Montreal, Canada, (1998) 869-875.
8. Overmyer ScottP., Lavoie.B.Rambow: "Conceptual modelling through linguistic analysis using LIDA", *Proceedings of the 23rd International Conference on Software Engineering, ICSE 2001*, Toronto (2001).
9. Sylvain Delisle, Ken Barker, Ismail Biskri: "Object-Oriented Analysis: Getting Help from Robust Computational Linguistic Tools, in G. Friedl, H.C. Mayr (eds) *Application of Natural Language to Information Systems*, Oesterreichische Computer Gesellschaft, (1999) 167-172.

# A Multi-word Term Extraction System

Jisong Chen, Chung-Hsing Yeh, and Rowena Chau

The Clayton School of Information Technology, Monash University, Clayton,  
Victoria 3800, Australia

{Jisong.Chen, ChungHsing.Yeh,  
Rowena.Chau}@infotech.monash.edu.au

**Abstract.** Traditional statistical approaches for identifying multi-word terms have to handle a large amount of noisy data and are extremely time consuming. This paper introduces a *multi-word term extraction system* for extracting multi-word terms from a set of documents based on the co-related *text-segments* existing in these documents. The system uses a short predefined stoplist as an initial input to segment a set of documents into text-segments, calculates the segment-weights of all text-segments, and then applies the short text-segments to segment the longer text-segments based on the weight values recursively until all text-segments cannot be further divided. The resultant text-segments can thus be identified as terms based on a specified threshold. The initial experimental result on a set of traditional Chinese documents shows that this system can achieve a minimum of 76.39% of recall rate and a minimum of 91.05% of precision rate on retrieving multiple occurrences terms, which include 18.30% of new identified terms.

## 1 Introduction

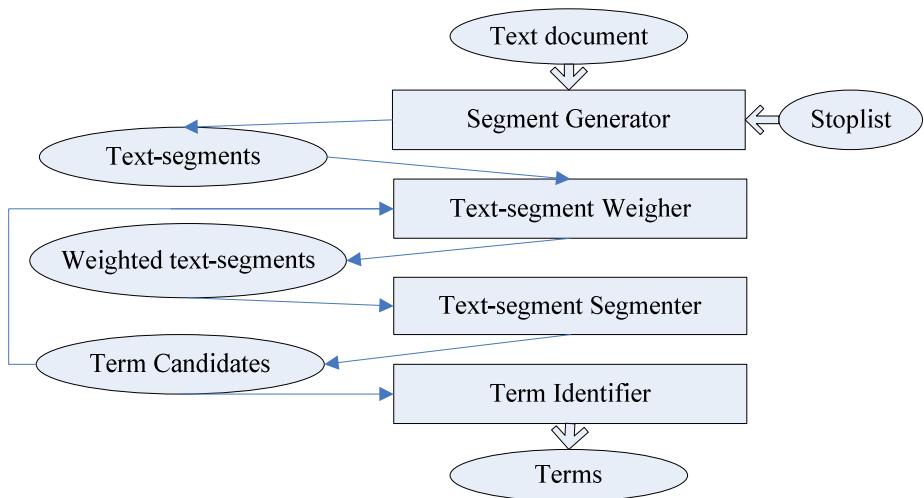
A term (a concept and its designation) can consist of a single word or multiple words. In general, a *multi-word term* may carry more meaning than a single-word term and can represent documents more actually. Statistical approaches to multi-word term identification are based on the detection of one or more lexical units in specialized documents with a frequency-derived value higher than a given threshold [1,2]. The concept is that documents are characterized by the repeated use of certain lexical units or morph-syntactic constructions.

Traditional statistical methods for multi-word *term identification* have to handle a large amount of noisy data and are very time consuming. In this paper, we suggest a *multi-word term extraction system* which uses a new and effective statistical method for identifying multi-word terms. The system uses a short predefined stoplist as an initial input to segment a set of documents into text-segments, calculates the segment-weights of all text-segments, and then applies the short text-segments to segment the longer text-segments based on the weight values. The system performs the weighting and segmenting tasks on the newly generated text-segments recursively until all text-segments cannot be further divided. The resultant text-segments can thus be identified as terms based on a specified threshold. The system has been experimented on a set of

traditional Chinese documents which was downloaded from the Hong Kong government website ([www.info.gov.hk](http://www.info.gov.hk)). The experimental result shows that this system can achieve a minimum of 76.39% of recall rate and a minimum of 91.05% of precision rate on retrieving multiple occurrences multi-word terms, which include 18.30% of new retrieved terms.

## 2 Proposal of the Multi-word Term Extraction System

Fig. 1 shows the proposed *multi-word term extraction system*. The system includes four components: a *text-segment generator* which uses a short predefined stoplist as an initial input to segment a set of text documents into text-segments; a *text-segment weigher* which calculates the *segment-weight* for each generated text-segments; a *text-segment segmenter* which segments all text-segments each other based on their segment-weights to generate new text-segments - term candidates; the term candidates can be re-input for further segmentation or directly input to the next component – a *term identifier* which identifies the resultant term candidates to be terms based on a specified threshold.



**Fig. 1.** A Proposed Multi-Word Term Extraction System

### 2.1 Generating Text-Segments

In a text document, some words, which have a very low discrimination value when it comes to Information Retrieval (IR) – known as stopwords, are existent. By removing these stopwords from each sentence, then a sentence may become one or multiple text-segments. A text-segment may include one or more words (a Chinese character is treated as a word). As this research focuses on multi-word term identification, a text-segment here only refers to a segment with multiple words.

During this stage, a term frequency ( $TF$ ) and a document frequency ( $DF$ ) for each *text-segment* are recorded. A term frequency for a *text-segment*  $t$  ( $TF_t$ ) represents the total occurrences of the *text-segment*  $t$ , and a document frequency for a *text-segment*  $t$  ( $DF_t$ ) refers the number of documents which the *text-segment*  $t$  occurs.

## 2.2 Weighing Text-Segments

To develop a scheme for weighing text-segments, the following factors have been considered: (a) a multi-word term with a longer *text-segment* is more descriptive than a short one. In other words, the *segment-length* of a *text-segment*, which indicates the number of words in the *text-segment*, must be included for estimating its *segment-weight*; (b) a new multi-word term such as a new terminology may frequently appear in certain documents which discuss this terminology. That is, this kind of terms should have a high  $TF$ ; (c) an existing multi-word term such as a commonly used phrase usually has a high  $DF$  and/or a high  $TF$ , due to its frequent usage.

Given a *text-segment*  $t \in C$  ( $C$  is a group of documents), the *segment-weight*  $W_t$  can be calculated in Equation (1). In Equation (1),  $TF_t$  is the term frequency of the *text-segment*  $t$ ,  $DF_t$  is the document frequency of *text-segment*  $t$ , and  $L_t$  denotes the *segment-length* of the *text-segment*  $t$ .

$$W_t = TF_t \times DF_t \times \sqrt{L_t} \quad (1)$$

Based on the Equation (1), a *text-segment* with longer *segment-length*, a high  $TF$  and/or a high  $DF$  will gain larger *segment-weight*.

## 2.3 Segmenting Text-Segments

Fig. 2 shows the rule for segmenting text-segments. Generally speaking, a term with a longer *text-segment* is more descriptive than a short one. Always applying the shorter *text-segment* to segment the longer one may lead to the loss of meaningful terms. As such, the segmentation rule in Fig. 2 applies a *segment-weight* for each *text-segment*. Only a *text-segment* with a higher weight can segment a *text-segment* with a lower weight. As discussed in Section 2.2, a multi-word term with a longer *text-segment* will usually have a higher *segment-weight*, and as such, will have a lower opportunity to be further segmented by other text-segments.

Given two text-segments  $p$  and  $q$ ,  $p$  can be further segmented by  $q$  when the following conditions hold simultaneously:

1.  $q \subset p$
2.  $W_q > W_p$

where  $W_p$  and  $W_q$  denote the segment-weights of text-segments  $p$  and  $q$  respectively.

**Fig. 2.** Rule for segmenting a text-segment

## 2.4 Selecting Terms

As discussed in Section 2.2, a new multi-word term such as a new terminology may have a high *TF* whereas an existing multi-word term may have a high *DF* and/or a high *TF*. As such, the term selection algorithm can be developed based on both *TF* and *DF*. In this research, a term-selection-weight (*TSW*) has been used to evaluate each term candidate. Equation (2) shows the developed method for evaluating the term-selection-weight for term candidate *i*. In Equation (2),  $TF_i$  and  $DF_i$  represent the values of *TF* and *DF* respectively for the term candidate *i*.

$$TSW_i = TF_i \times DF_i \quad (2)$$

Once each term candidate has been given its term selection weight, the system can select the terms from these candidates based on a specified threshold.

## 3 Experiment

### 3.1 Experimental Resource

The Hong Kong government Website at <http://www.info.gov.hk> is used for the experiment. The experiment has been mainly conducted on the traditional Chinese documents (encoded in BIG-5). An Oracle defined stoplist [3] has been used to segment these documents. A P4, CPU 1.8G, Ram 256MB computer was used for the experiment.

By using a Web spider, called Teleport, a collection of 308 Web documents in traditional Chinese were randomly fetched from the Website. The extracted terms from these documents were identified based on the following methods: (a) Using Tsai's list of Chinese words [4] dictionary to identify the extracted terms; (b) for remaining un-identified terms, manually checked their correctness.

### 3.2 Setting Term Selection Thresholds and Evaluation Base

Based on the Equation (2), three experimental thresholds Ex1, Ex2 and Ex3 have been identified as 2, 3 and 4 respectively. The Ex1 is assigned to extract the minimum multiple-occurrence terms. The Ex2 and Ex3 are assigned for comparing with Ex1.

To provide an evaluation base, the Tsai's list of Chinese words has been applied to extract terms from the selected 308 documents. A total of 11662 terms have been extracted from these documents and the total time spent for this extraction process was 690 mins 24 seconds. Table 1 shows the extraction result for different thresholds.

**Table 1.** Dictionary-based extraction result

	Ex1	Ex2	Ex3
No. Of Terms (NOT)	6939	5216	4920

### 3.3 Experiment Result

Table 2 shows the experiment result based on different thresholds. For each result, a number of extracted terms including in the dictionary (TiD) and a number of extracted terms excluding from the dictionary (TxD) have been recorded.

**Table 2.** Experiment Result

	<b>Ex1</b>	<b>Ex2</b>	<b>Ex3</b>
<b>TiD</b>	5301	4273	4251
<b>TxD</b>	1985	1251	1215

**Recall and Precision:** Table 3 shows the recall rate and the precision rate for each result. The recall rate and the precision rate are calculated based on Equations (3) and (4) respectively. The result in Table 3 shows that the higher threshold is applied, the higher recall rate and the higher precision rate can be achieved.

**Table 3.** Recall and Precision

	<b>Ex1</b>	<b>Ex2</b>	<b>Ex3</b>
<b>Recall (%)</b>	76.39	81.92	86.40
<b>Precision (%)</b>	72.76	77.35	77.77

$$\text{Recall} = \text{TiD}_{\text{Exi}} / \text{NOT}_{\text{Exi}} \quad (3)$$

$$\text{Precision} = \text{TiD}_{\text{Exi}} / (\text{TiD}_{\text{Exi}} + \text{TxD}_{\text{Exi}}) \quad (4)$$

**New Identified Terms:** the experimental result in Table 2 shows that a large number of extracted terms are not included in the dictionary (TxD). By manually examining these terms, most of them are meaningful phrases, person's names, street names, and so on. By including these manually identified new terms, Equation (5) should be used to calculate the precision rate instead of Equation (4). Table 4 shows the total number of these manually identified new terms (NT) and the recalculated precision rates.

$$\text{Precision} = (\text{TiD}_{\text{Exi}} + \text{NT}_{\text{Exi}}) / (\text{TiD}_{\text{Exi}} + \text{TxD}_{\text{Exi}}) \quad (5)$$

**Table 4.** Number of Manually Identified New Terms (NT)

	<b>Ex1</b>	<b>Ex2</b>	<b>Ex3</b>
<b>NT</b>	1333	944	917
<b>Precision (%)</b>	91.05	94.44	94.65

**System Efficiency:** the complexity of the *multi-word term extraction system* is  $O(N^2)$  in worst case, where N is the number of the generated text-segments. In this

experiment, the total time spent was 386 minus and 35 seconds (the dictionary-based approach spent 690 minus and 24 seconds).

## 4 Conclusion

In this paper, we have presented the *multi-word term extraction system*, a new automatic statistical approach for identifying multi-word terms based on co-related text-segments existing in a set of documents. New algorithms have been developed to identify multi-word terms effectively and efficiently. Object-oriented techniques have been applied to develop this extendable system. The experiment conducted on a set of traditional Chinese documents downloaded from the Hong Kong government Web site has shown that this system can achieve a minimum of 76.39% of recall rate and a minimum of 91.05% of precision rate for retrieving multiple occurrences terms, which include 18.30% of new identified terms. The experiment on English documents is ongoing.

## References

1. J. S. Chang, S. D. Chen, S. J. Ker, Y. Chen, and J. Liu, “A multiple-Corpus Approach to Recognition of Proper Names in Chinese Texts”, *Computer Processing of Chinese and Oriental Languages*, Vol. 8, No. 1, 1994, pp. 75–85.
2. Y.-S. Lai and C.-H. Wu, “Unknown Word and Phrase Extraction Using a Phrase-Like-Unit-Based Likelihood Ratio”, *International Journal of Computer Processing of Oriental Languages*, Vol. 13, No. 1 (2000) 83–95.
3. Chinese Stoplist (Traditional). <http://www.lc.leidenuniv.nl/awcourse/oracle/text.920/a96518/astopsup.htm#45728>
4. C.-H. Tsai, “A Review of Chinese Word Lists Accessible on the Internet”, <http://technology.chtsai.org/wordlist/>.

# A Multiscale Self-growing Probabilistic Decision-Based Neural Network for Segmentation of SAR Imagery

Xian-Bin Wen<sup>1</sup>, Hua Zhang<sup>1</sup>, and Zheng Tian<sup>2</sup>

<sup>1</sup> Department of Computer Science and Engineering, Tianjin University of Technology,  
Tianjin 300191, China  
xb\_wen@yahoo.com.cn

<sup>2</sup> Northwestern Polytechnical University, Xi'an, 710072, China

**Abstract.** A new segmentation algorithm for synthetic aperture radar (SAR) image is proposed using multiscale self-growing probabilistic decision-based neural network (MSPDNN). The proposed algorithm is able to find the natural number of category in SAR image based on the Bayesian information criterion (BIC). The learning process starts from a single SAR image at proper scale randomly initialized in the feature space, and grows adaptively during the learning process until most appropriate number of category are found. Experimental results of the proposed algorithm are proposed and compared with that of previous algorithms.

## 1 Introduction

Synthetic Aperture Radar (SAR) image segmentation is usually said to be a complex problem in the pattern recognition area, due to the presence of speckle. To fully exploit the coherent nature and complexity of SAR image formation, we employ a introduced class of mixture multiscale autoregressive (MMAR) model evolving on dyadic trees, and the expectation-maximization (EM) algorithm to the MMAR model is applied [1]. However, the EM algorithm has a high possibility of being trapped in local optima and is also slow to the converge [2]. In this paper, we propose a new MMAR-based neural network, namely multiscale self-growing probabilistic decision-based neural networks (MSPDNN). The leaning process starts from randomly initializing a single SAR image at any scale in the feature space and adaptively growing the category until the most appropriate number of category are reached.

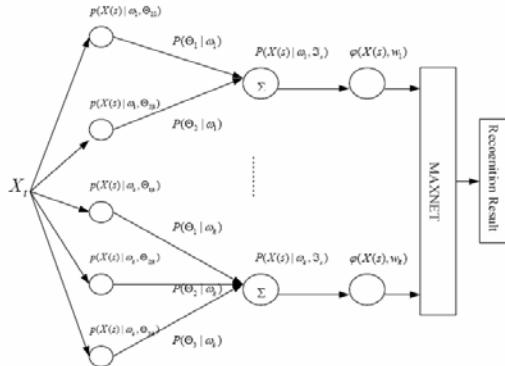
## 2 Multiscale Self-growing Probabilistic Decision-Based Neural Network

Given SAR image associated with class  $\omega_i$ , multiscale sequence  $X_L, X_{L-1}, \dots, X_0$  of SAR images are constructed as in [1], the pixel mapped to node  $s$  is denoted as  $X(s)$ . We assume that the likelihood function  $f(X(s)|\omega_i, \mathcal{S}_s)$  for class  $\omega_i$  is a MMAR model, i.e.

$$\begin{aligned}
f(X(s) | \omega_i, \mathfrak{I}_s) &= \sum_{r=1}^R P(\Theta_{rli} | \omega_i) p(X(s) | \omega_i, \Theta_{rli}). \\
&= \sum_{r=1}^R P(\Theta_{rli} | \omega_i) \Phi\left(\frac{X(s) - a_{rli,0} - a_{rli,1}X(s\bar{\gamma}) - \dots - a_{rli,p_{ri}}(s\bar{\gamma}^{p_{ri}})}{\sigma_{rli}}\right). \quad (1)
\end{aligned}$$

and the parameters of the MMAR model can be estimated iteratively by EM algorithm[1], where  $\Theta_{rli}$  represent the parameters of the  $r$ th mixture component,  $R$  is the total number of mixture components,  $p(X(s) | \omega_i, \Theta_{rli})$  and  $P(\Theta_{rli} | \omega_i)$  is distribution function and the prior probability (also called mixture coefficients) of  $r$ th component, respectively.  $\sum_{r=1}^R P(\Theta_{rli} | \omega_i) = 1$ ,  $f(X(s) | \omega_i, \mathfrak{I}_s)$  is the probability distribution function.  $\mathfrak{I}_s$  is the set of  $X(s\bar{\gamma}), \dots, X(s\bar{\gamma}^{p_i})$  ( $p_i = \max_r p_{ri}$ ),  $s\bar{\gamma}$  is defined to refer- ence the parent of node  $s$ , and  $\Phi(\cdot)$  be the standard normal distribution function.

MSPDNN is a multi-variate of multiscale Gaussian neural network, and employ a modular network structure, as shown in Fig.1. A detailed description of the MSPDNN will be given in the following sections.



**Fig. 1.** The structure of the multiscale self-organizing mixture network

## 2.1 Discriminant Functions of MSPDNN

Based on the likelihood function  $p(X(s) | \omega_i, \mathfrak{I}_s)$  for class  $\omega_i$ , the discriminant function of the muti-class MSPDNN models the log-likelihood function

$$\varphi(X(s), w_i) = \log f(X(s) | \omega_i, \mathfrak{I}_s). \quad (2)$$

where  $w_i = \{\Theta_{rli}, P(\Theta_{rli} | \omega_i), T_i\}$ .  $T_i$  is the output threshold of the subnet  $i$ .

## 2.2 Locally Unsupervised Learning

Given a set of patterns  $X^+ = \{X(t), t = 1, 2, \dots, N\}$  and a set of candidate MMAR models  $\mathbf{M} = \{MM_i \mid i = 1, 2, \dots, L\}$ , each model associated with a parameter set  $w_i$ . In order to select a proper model  $MM_i$  from  $\mathbf{M}$  to represent the distribution of  $X^+$ , the Bayesian information criterion (BIC) for model  $M_i$  and training data  $X^+$  is defined as

$$\text{BIC}(MM_i, X^+) = -2 \log P(X^+ \mid \hat{w}_i, MM_i)) - d(MM_i) \log N. \quad (3)$$

where  $\hat{w}_i$  is a maximum likelihood estimate of  $w_i$ ,  $d(MM_i)$  is the number of free parameters in model  $MM_i$ , and  $N$  is the number of train data. Like analysis in [3], choosing the model with minimum BIC is equivalent to choosing the model with the largest posterior probability. And, BIC can be used to compare model with differing parameterization, differing numbers of cluster components, or both. So, if there are two candidate models  $MM_1$  and  $MM_2$  for modeling a data set  $X^+$ , BIC difference  $\Delta\text{BIC}_{21}$

$$\Delta\text{BIC}_{21}(X^+) = \text{BIC}(MM_2, X^+) - \text{BIC}(MM_1, X^+). \quad (4)$$

can be used to evaluate which model is a preferred one.

The unsupervised training process of LU learning based on BIC can be described below:

- (1) Construct multiscale sequence of SAR image.
- (2) Set the initial number of multiscale mixture Gaussian components:  $G_c = 1$ , and set randomly initial values of parameters in  $\Theta$ , which represents the parameters of a MMAR.
- (3) If  $\Delta\text{BIC}_{21}(X^+) \leq \text{growing-confidence}$ , then, relearn  $\Theta$  by applying EM algorithm on a uni-component multiscale mixture Gaussian on  $X^+$ , and process terminates; or else increment  $G_c$ , and relearn  $\Theta$  by applying EM algorithm on two components multiscale mixture Gaussian on  $X^+$ .
- (4) Clustering: let  $EM\_class_i$  denote the input data  $X(t)$  which belongs to the  $i$ th multiscale Gaussian component after EM learning. For each pattern  $X(t)$  in  $X^+$ , if  $k = \arg \max_i \{P(\lambda_i \mid X(t))\}$ , assign  $X(t)$  to  $EM\_class_i$ .
- (5) Grows one component: let  $\text{growing} = \arg \max_i \{\Delta\text{BIC}_{21}(\text{EM\_class}_i)\}$ , for  $i = 1, \dots, G_c$ , if  $\max_i \{\Delta\text{BIC}_{21}(\text{EM\_class}_i)\} \leq \text{growing-confidence}$ , process terminates; or else we initialize parameters  $\tilde{\Theta}_1$  and  $\tilde{\Theta}_2$  of the newly split two components from  $EM\_class_{growing}$ , and remove the parameter  $\tilde{\Theta}_{growing}$  from  $\Theta$ ; update  $\Theta$  by putting  $\tilde{\Theta}_1$  and  $\tilde{\Theta}_2$  into  $\Theta$ ; increment  $G_c$ .
- (6) Using current  $\Theta$  as the initial values, perform EM learning on all the clusters.
- (7) Repeat (4)-(6) until process terminates.

### 2.3 Global Supervised Learning

In the Globally Supervised (GS) training phase, training data are used to fine-tune the decision boundaries. Specifically, when a training pattern is misclassified to the  $i$ th class, reinforced and /or anti-reinforced learning are applied to update the parameters of subnet  $i$ . Thus we have reinforced learning

$$w_i^{(m+1)} = w_i^{(m)} + \eta \nabla \varphi(X_i(m), w_i). \quad (5)$$

and anti-reinforced learning

$$w_j^{(m+1)} = w_j^{(m)} - \eta \nabla \varphi(X_i(m), w_j). \quad (6)$$

where  $\eta$  is user-assigned (positive) learning rates,  $0 < \eta \leq 1$ . For the false rejection data set  $D_2^i$ , reinforced and anti-reinforced learning will be applied to class  $\omega_i$  and  $\omega_j$ , respectively. As for the false acceptance set  $D_3^i$ , anti-reinforced learning will be applied to class  $\omega_i$ , and reinforced learning will be applied to class  $\omega_j$ . The gradient vector  $\nabla \varphi$  in (5) and (6) can be computed in the similar manner, as proposed in [4]. The false rejection data set  $D_2^i$  and the false acceptance set  $D_3^i$  are defined as follows:

$$D_2^i = \{ X(t); X(t) \in \omega_i, X(t) \text{ is misclassified to another class } \omega_j \}.$$

$$D_3^i = \{ X(t); X(t) \notin \omega_i, X(t) \text{ is classified to } \omega_i \}.$$

The threshold value  $T_i$  of a subnet  $i$  in the MSPDNN recognizer can also be learned by reinforced or anti-reinforced learning rules. Specifically, the threshold  $T_i$  at iteration  $m$  is updated according to the reinforced learning

$$T_i^{(m+1)} = T_i^{(m)} + \eta_i l'(T_i^{(m)} - \varphi(X_i(m), w_i)). \quad (7)$$

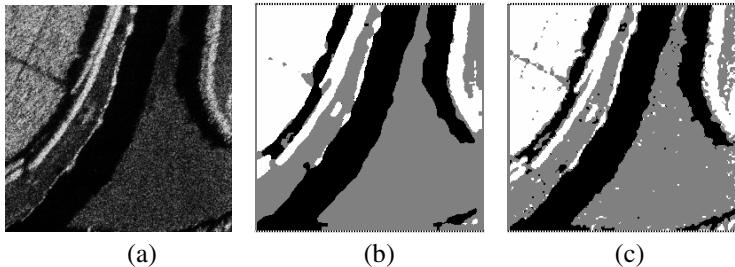
and anti-reinforced learning

$$T_i^{(m+1)} = T_i^{(m)} - \eta_i l'(\varphi(X_i(m), w_i) - T_i^{(m)}). \quad (8)$$

where  $\eta_i$  is a positive learning parameter,  $l(\cdot)$  is a penalty function, and  $l'(\cdot)$  is the derivative of the penalty function.

## 3 Experimental Results for SAR Imagery

To demonstrate the proposed algorithm, we apply it to a complex SAR images in Fig. 2(a), which consists of multiple classes of homogeneous regions, respectively. In the experiments, we generate an above-mentioned quadtree representation and use a two orders regression. Because it is found that by increasing the regression order to  $p = 2$  for both images, we can achieve a lower probability of misclassification and a



**Fig. 2.** (a) Original SAR image composed of woodland and cornfield. (b) Segmented image obtained using EM algorithm. (c) Segmented image obtained using MSPDNN algorithm.

good trade-off between modeling accuracy and computational efficiency. Learning rates  $\eta$  and  $\eta_i$  in the MSPDNN were set to 0.5 and 0.05, respectively. The penalty function  $l(x)$  is chosen to be  $1/(1+\exp(x))$ .

Fig. 2 shows segmentation results from applying MSPDNN to the SAR images, as well as results from EM algorithm in [1] for comparison. From Fig. 2, the MSPDNN algorithm not only performs better than the EM algorithm, especially at the boundaries, but also can automatically select the proper number of cluster in images, and converges much faster than the EM algorithm, and considerably reduces the segmentation time.

## References

1. Wen, X.B., Tian, Z.: Mixture Multiscale Autoregressive Modeling of SAR Imagery for Segmentation. *Electronics Letters* 39 (2003) 1272-1274
2. Redener, R.A., Walker, H.F.: Mixture Densities, Maximum Likelihood and the EM Algorithm. *SIAM Rev.* 26 (1984) 195-239
3. Kass, R.E.: Bayes Factors. *J. Amer. Statist. Assoc.* 90 (1995) 773-795
4. Lin, S.H., Kung, S.Y., Lin, L.J.: Face Recognition/detection by Probabilistic Decision-based Neural Networks. *IEEE Trans. Neural Networks*, special issue on Artifi. Neural Network Pattern Recog. 8(1997) 114-132

# Face Detection Using an Adaptive Skin-Color Filter and FMM Neural Networks\*

Ho-Joon Kim<sup>1</sup>, Tae-Wan Ryu<sup>2</sup>, Juho Lee<sup>3</sup>, and Hyun-Seung Yang<sup>3</sup>

<sup>1</sup> School of Computer Science and Electronic Engineering  
Handong University, Pohang, 791-708, Korea  
hjkim@handong.edu

<sup>2</sup> Department of Computer Science, California State University,  
Fullerton, CA, 92834 , USA  
tryu@ecs.fullerton.edu

<sup>3</sup> Department of Computer Science, KAIST  
Daejeon, 305-701, Korea  
{jhlee, hsyang}@paradise.kaist.ac.kr

**Abstract.** In this paper, we present a real-time face detection method based on hybrid neural networks. We propose a modified version of fuzzy min-max (FMM) neural network for feature analysis and face classification. A relevance factor between features and pattern classes is defined to analyze the saliency of features. The measure can be utilized for the feature selection to construct an adaptive skin-color filter. The feature extraction module employs a convolutional neural network (CNN) with a Gabor transform layer to extract successively larger features in a hierarchical set of layers. In this paper we first describe the behavior of the proposed FMM model, and then introduce the feature analysis technique for skin-color filter and pattern classifier.

## 1 Introduction

Growing interest in computer vision has motivated a recent surge in research on problems such as face recognition, pose estimation, face tracking and gesture recognition. However, most methods assume human faces in their input images have been detected and localized [1-2]. Recently, skin detection has emerged as an active research topic in several practical applications including face detection and tracking [3-4]. In this paper we present an adaptive skin-color filter model which is capable of adjusting the skin-color model by a training process. We also present an improved neuro-fuzzy pattern classification model based on FMM neural networks [5-6]. Since the weight factor can be adjusted by training process, the system can prevent undesirable performance degradation which may be caused by some environmental factors such as

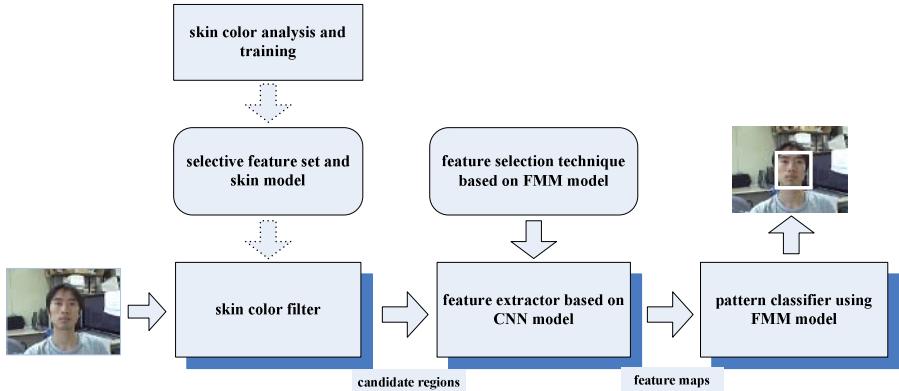
---

\* This research was supported by a 21<sup>st</sup> Century Frontier R&D Program and Brain Neuroinformatics Research Program sponsored by Minister of Information and Communication and Minister of Commerce, Industry and Energy in KOREA.

illumination changes. Through the feature analysis using the proposed model, we can select the most relevant features for the skin-color filter as well as the pattern classifier.

## 2 Underlying System

As shown in Fig. 1, the underlying face detection system consists of three subprocesses: skin-color filter, feature extractor and pattern classifier.



**Fig. 1.** The underlying face detection system

Through the skin color analysis and training process, the system can generate an adaptive skin model and a relevant feature set for the given illumination condition. The feature extractor generates numerous features from the input image. The number of features and the relevance factors of the features affect the computation time and the performance of the system. Therefore we propose a feature analysis technique to reduce the amount of features for the pattern classifier.

## 3 A Weighted FMM Neural Network

We have proposed a modified FMM neural network[7] called weighted fuzzy min-max(WFMM) neural network. In this paper we present an improved structure of the model and a feature analysis method.

As shown in Equation (1) and (2), the model employs an activation function which has the factors of feature value distribution and the weight value for each feature in a hyperbox. The hyperbox membership function has weight factor to consider the relevance of each feature as different value. In the equation,  $w_{ij}$  is the connection weight between  $i$ -th feature and  $j$ -th hyperbox. The weighted FMM neural network is capable of utilizing the feature distribution and frequency in learning process as well as in classification process. Since the weight factor effectively reflects the relationship

between feature range and its distribution, the system can prevent undesirable performance degradation which may be caused by noisy patterns.

$$b_j(A_h) = \frac{1}{\sum_{i=1}^n w_{ji}} \cdot \sum_{i=1}^n w_{ji} [\max(0, 1 - \max(0, \gamma_{jiv} \min(1, a_{hi} - v_{ji}))) + \max(0, 1 - \max(0, \gamma_{jiu} \min(1, u_{ji} - a_{hi}))) - 1.0] \quad (1)$$

$$\begin{cases} \gamma_{jiU} = \frac{\gamma}{R_U} & R_U = \max(s, u_{ji}^{new} - u_{ji}^{old}) \\ \gamma_{jiv} = \frac{\gamma}{R_V} & R_V = \max(s, v_{ji}^{old} - v_{ji}^{new}) \end{cases} \quad (2)$$

Consequently the proposed model can provide more robust performance of pattern classification when the training data set in a given problem includes some noise patterns or unusual patterns.

## 4 A Feature Analysis Technique

The most advantageous feature of convolutional neural network is invariant detection capability for distorted patterns in images [1-2]. The underlying system employs a convolutional neural network in which a Gabor transform layer is added at the first layer. The first layer of the network extracts local feature maps from the input image by using Gabor transform filters. The other layers of the feature extractor include two types of sub-layers called *convolution layer* and *sub-sampling layer*. Each layer of the network extracts successively larger features in a hierarchical set of layers. Finally a feature set is generated for the input of the pattern classifier. The number of the features can be reduced by the feature analysis technique using the FMM model described in the previous section. We define a measure called *relevance factor (RF)* as shown in Equation (3). The measure means the degree of relevance between a feature value and a pattern class.

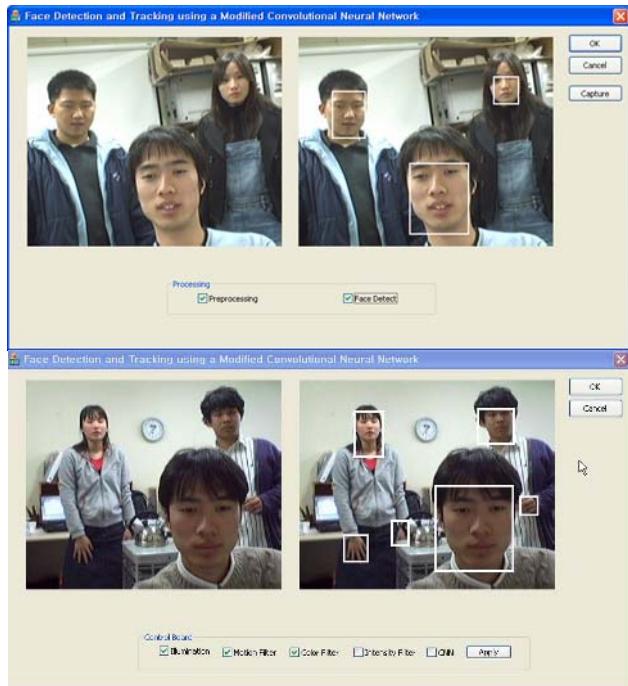
$$RF(x_i, C_k) = (\frac{1}{N_k} \sum_{B_j \in C_k} S(x_i, (u_{ji}, v_{ji})) \cdot w_{ij} - \frac{1}{(N_B - N_k)} \sum_{B_j \notin C_k} S(x_i, (u_{ji}, v_{ji})) \cdot w_{ij}) / \sum_{B_j \in C_k} w_{ij} \quad (3)$$

In the equation, constant  $N_B$  and  $N_k$  are the total number of hyperboxes and the number of hyperboxes that belong to class k, respectively. S is a similarity measure between two fuzzy intervals. If the  $RF(x_i, k)$  has a positive value, it means an

excitatory relationship between the feature  $X_i$  and the class  $k$ . But a negative value of  $RF(x_i, k)$  means an inhibitory relationship between them.

## 5 Experimental Results

For the training of skin-color filter, the system considers eleven color features which are labeled as  $F-1 = Red$ ,  $F-2 = Green$ ,  $F-3 = Blue$ ,  $F-4 = Intensity$ ,  $F-5 = Cb$ ,  $F-6 = Cr$ ,  $F-7 = magenta$ ,  $F-8 = Cyan$ ,  $F-9 = Yelleow$ ,  $F-10 = Hue$ , and  $F-11 = Saturation$ .



**Fig. 2.** Two training data captured under different illumination conditions

**Table 1.** Feature analysis results for the two different images

<i>image - 1</i>			<i>image - 2</i>		
features	feature range	<i>RF</i>	features	feature range	<i>RF</i>
<i>F-5</i>	0.547 ~ 0.737	9.3703	<i>F-11</i>	0.027 ~ 0.128	0.8888
<i>F-3</i>	0.435 ~ 0.627	9.2604	<i>F-10</i>	0.828 ~ 0.983	0.5827
<i>F-9</i>	0.372 ~ 0.564	9.2631	<i>F-6</i>	0.053 ~ 0.233	0.5004
<i>F-11</i>	0.074 ~ 0.279	8.7050	<i>F-5</i>	0.759 ~ 0.958	0.4529

Table 1 shows the skin-color analysis result and the feature range data derived from the training process. As shown in the table, different kinds of features can be adaptively selected for a given condition, and the feature ranges of skin-color filter can be also adjusted by the training process. Table 1 shows four features which have the highest value of the relevance factor RF. A number of hyperboxes for face and non-face patterns have been generated and the relevance factors are also adjusted through the training process. Therefore the system can select more effective feature set adaptively for the given environment.

## 6 Conclusion

A feature analysis method for face detection using a modified FMM model has been introduced. Through the training process, the skin-color filter is adapted for the illumination condition under which the given images are captured. A relevance factor has been defined for the feature selection technique. The measure can be utilized in designing an optimal structure of the classifier. We have applied the proposed model to a real-time face detection system in which the illumination conditions are frequently changed.

## References

1. Garcia, Cristophe and Delakis, Manolis: Convolutional Face Finder: A Neural Architecture for Fast and Robust Face Detection, IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol.26, No.11, (2004) 1408-1423
2. Lawrence, Steve, Giles, C. L., Tsoi, A. C. and Back, Andrew D.: Face Detection: A Convolutional Neural-Network Approach, IEEE Transaction n Neural Networks, Vol.8, No.1, (1997) 98-113
3. Hsu, Rein-Lien, Mohamed Abdel-Mottaleb and Jain, Anil K.: Face Detection in Color Images," IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol.24, No.5, (2002) 696-706
4. Storring, Moritz, Kocka, Tomas, Andersen, Hans J., Granum, Erik: Tracking Regions of Human Skin Through Illumination Changes, Pattern Recognition Letters, Vol.24, (2003) 1715-1723
5. Simpson, P. K.: Fuzzy Min-Max Neural Networks Part 1: Classification. IEEE Transaction on Neural Networks, Vol.3. No.5. (1997) 776-786
6. Gabrys, B. and Bargiela A.: General Fuzzy Min-Max Neural Network for Clustering and Classification. IEEE Transaction on Neural Networks, Vol.11. No.3. (2000) 769-783
7. Kim, H. J., Ryu, T. W., Nguyen, T. T., Lim, J. S. and Gupta, S.: A Weighted Fuzzy Min-Max Neural Network for Pattern Classification and Feature Extraction. Proceeding of International Conference on Computational Science and Its Application, Part.4 (2004) 791-798

# GA Optimized Wavelet Neural Networks

Jinhua Xu

Department of Computer Science, East China Normal University  
jhxu@cs.ecnu.edu.cn

**Abstract.** In this paper, a new GA-based constructive algorithm is proposed for wavelet neural networks. Wavelets will be added to the WNNs from low resolution level to high resolution level. At each resolution, the translation parameters of a new wavelet is trained using GA, and output weights is obtained using least square techniques. The proposed algorithm is suitable to the high dimensional problems.

## 1 Introduction

Wavelet transforms have emerged as a means of representing a function in a manner which readily reveals properties of the function in localized regions of the joint time-frequency space. The idea of combining wavelets with neural networks has led to the development of wavelet neural networks (WNNs) [1].

The determination of a network size and weight parameters is clearly critical when using WNNs. Some research have been done on this problem. In [2], an iterative method which combining genetic algorithms and least squares techniques is proposed for optimizing WNNs. GAs are used for optimal selection of the structure of the WNNs and the parameters of the transfer function of its neurons. Least squares techniques are used to update the weights of the network. In [3], a new class of wavelet networks is proposed, where the model structure for a high dimensional system is chosen to be a superimposition of a number of functions with fewer variables. A forward orthogonal least squares algorithm and the error reduction ratio is applied to select the model terms. In [4], wavelet network is constructed by some selected wavelets from a wavelet basis by exploring the sparseness of training data and using techniques in regression analysis. In [5], an orthogonalized residual based selection (ORBS) algorithm is proposed for WNNs.

The use of evolutionary algorithms (EA) to aid in the artificial neural network (ANN) learning has been a popular approach to address the local optima and design problem of ANN[6]. The typical approach is to combine the strength of backpropagation (BP) in weight learning and EA's capability of searching the architecture space. Some EA methods were proposed to learn both the network structure and connection weights[2].

Genetic algorithm (GA) is a directed random search technique that is widely applied in optimization problems. This is especially useful for complex optimization problems where the number of parameters is large and the analytical

solutions are difficult to obtain. GA can help to find out the optimal solution globally over a domain. In this paper, a new constructive algorithm is proposed for WNNs. Wavelets will be added to the WNNs from low resolution level to high resolution level. At each resolution, GA is applied to select the wavelet basis. The local optima problem of gradient-based algorithm is avoided. The proposed algorithm is suitable to the high dimensional problems.

## 2 Preliminaries

The wavelet analysis procedure is implemented with dilated and translated versions of a mother wavelet. In theory, the dilation (scale) parameter of a wavelet can be any positive real value and the translation (shift) can be an arbitrary real number. In practice, in order to improve computation efficiency, the values of the shift and scale parameters are often limited to some discrete lattices. This is then referred to as the discrete wavelet transform (DWT). The WNNs stemmed from the DWT have a linear-in-parameter structure[4,5]. In practical applications, it is unnecessary and impossible to represent a signal using an infinite decomposition in terms of wavelet basis functions. The decomposition are therefore often truncated at an appropriate accuracy. An approximation to a function  $f \in L^2(\mathbf{R})$  using the truncated wavelet decomposition with the coarsest resolution  $J_{min}$  and the finest resolution  $J_{max}$  can be expressed in the following:

$$f(x) = \sum_{j=J_{min}}^{J_{max}} \sum_{k \in K_j} c_{j,k} \psi_{j,k}(x) \quad (1)$$

where  $K_j$  are subsets of  $\mathbf{Z}$  and often depend on the resolution level  $j$  for all compactly supported wavelets and for most rapidly vanishing wavelets.

The wavelet network in (1) may involve a great number of candidate wavelet terms. Experience shows that often many of the terms are redundant and only a small number of significant wavelet terms are necessary to describe a given nonlinear system with a given accuracy. Some basis selection algorithms were proposed to select the significant basis from the candidate wavelet library[4,5]. However, when the number of the basis in the wavelet library is very large, the heavy computational cost may make the basis selection algorithms not feasible in practice. In this paper, a genetic algorithm is introduced to find the significant wavelets which should be included to the wavelet networks.

## 3 A GA-Based Constructive Algorithm for Wavelet Networks

In this section, a new constructive algorithm is proposed for WNNs, which starts with no wavelet in the WNN and adds new wavelets trained using GA.

Given  $N$  pairs of training sample,  $\{(x(1), y(1)), \dots, (x(N), y(N))\}$ . Set the desired output  $y = [y(1), y(2), \dots, y(N)]^T$ . Suppose the dilation is in the range  $[J_{min}, J_{max}]$ , where  $J_{min}$  and  $J_{max}$  are integers which represent the coarsest and finest resolution level respectively. For simplicity, assume that the dilations of all dimension for each wavelet are equal, that is,  $d_{i1} = \dots = d_{in} = d_i$ . The algorithm can be summarized as follows:

Step 1: Initialization. Set the output of the WNN  $\hat{y}_0 = 0$ , the residual  $r_0 = y$ ; the dilation  $d = J_{min}$ ; Set the number of wavelet  $i = 1$ .

A WNN with  $i - 1$  wavelets implements the function given by

$$\hat{y}_{i-1}(x) = \sum_{j=1}^{i-1} w_j \psi_j(x) \quad (2)$$

where  $\psi_j(x)$  represents the function implemented by the  $j^{th}$  wavelet. Moreover,  $r_{i-1}(x) = y(x) - \hat{y}_{i-1}(x)$  is the residual error function for the current network with  $i - 1$  wavelets. Addition of a new wavelet proceeds in two steps:

Step 2: Input training.

Use GA to train the translation parameters  $\Theta = [\theta_1, \theta_2, \dots, \theta_n]^T$ , where  $n$  is the dimension of the wavelet. Set

$$\phi = [\psi(1), \psi(2), \dots, \psi(N)]^T$$

where  $\psi(t) = \prod_{j=1}^n \psi(2^d x_j(t) - \theta_j)$ . Let

$$\hat{y}_i = \hat{y}_{i-1} + w\phi \quad r_i = y - \hat{y}_i = r_{i-1} - w\phi$$

with  $w = (\phi^T \phi)^{-1} \phi^T r_{i-1}$ .

The best  $\Theta$  may be selected to minimize the cost function

$$\begin{aligned} V_i(\Theta) &= r_i^T r_i = (r_{i-1} - w\phi)^T (r_{i-1} - w\phi) \\ &= r_{i-1}^T r_{i-1} - (\phi^T \phi)^{-1} (\phi^T r_{i-1})^2 \end{aligned} \quad (3)$$

A genetic algorithm is proposed to solve the optimization problem in (3) to find the  $\Theta^*$ . The translation parameters of a new wavelet are encoded into the chromosome. The fitness function to evaluate a chromosome in the population can be chosen as the residue reduction of a new wavelet, which is written as

$$f(\Theta) = (\phi^T \phi)^{-1} (\phi^T r_{i-1})^2 \quad (4)$$

Spinning the roulette wheel is used as the selection operator. A conventional one-point crossover operator has been employed. Two mutation operators, random mutation and little perturbation mutation, have been used.

Step 3: Output training. If  $V_{i-1} - V_i(\Theta^*) < \lambda V_{i-1}$ , ( $\lambda$  is a chosen constant threshold for the decay rate), the wavelet is rejected, goto step 4 for the next

resolution; otherwise,  $\Theta$  is accepted and set  $T_i := [t_{i1}, \dots, t_{in}]^T = \Theta^*$ ,  $d_i := d$  and  $\phi_i := [\psi_i(1), \dots, \psi_i(N)]$ , where  $\psi_i(t) = \prod_{j=1}^n \psi(2^{d_i}x_j(t) - t_{ij})$ . Then  $\phi_i$  is normalized as  $v_i = \phi_i / \sqrt{\phi_i^T \phi_i}$ . Suppose  $i - 1$  wavelets have been obtained and orthonormalized as  $q_1, q_2, \dots, q_{i-1}$ . The new obtained  $v_i$  is orthogonalized to the previous wavelet as follows:

$$p_i = v_i - ((v_i^T q_1)q_1 + \dots + (v_i^T q_{i-1})q_{i-1}) \quad (5)$$

$$q_i = p_i / \sqrt{p_i^T p_i} \quad (6)$$

$$\bar{w}_i = q_i^T y \quad (7)$$

and set

$$\hat{y}_i = \hat{y}_{i-1} + \bar{w}_i q_i \quad r_i = r_{i-1} - \bar{w}_i q_i$$

If  $r_i^T r_i < \epsilon$ , the approximation accuracy is reached, go to step 5; Otherwise, set  $i := i + 1$  and goto step 2 to train a new wavelet in the current resolution.

Step 4: Change the dilation parameter  $d := d + 1$ , if  $d < J_{max}$ (maximum resolution), go to step 2; otherwise, go to step 5.

Step 5: Set  $M := i$ (the number of wavelets). Stop training.

## 4 Numerical Examples

Chaotic time series identification: The logistic map[2] is a chaotic time series close to being a white noise that satisfies the ergodicity property. This series can be generated as follows:

$$x_{n+1} = 4x_n(1 - x_n), \quad x_0 \in (0, 1) \quad (8)$$

The 1-D wavelet used is the Mexican hat with support  $[-4, 4]$ . The population size used for the GA is 20, the maximum generation is 200.

In order to compare the prediction results of the WNN with other work [2], the training set and testing sets are identical to that in [2]. The training set consists of 2000 points extracted from the series generated from (8) with initial condition  $x_0 = 1/11$ . Ten independent runs are performed. To test the generalization capabilities of the constructed wavelet networks, 8000 points have been generated starting from the following different initial conditions:

$$x_1 = \sqrt{2}/2, \quad x_2 = \sqrt{3}/3, \quad x_3 = 1/11, \quad x_4 = 4/7, \quad x_5 = 8/9 \quad (9)$$

The generalization capabilities are measured in terms of the normalized square error,  $ERR\% = 100 * \sum_{k=1}^N (y(k) - \hat{y}(k))^2 / \sum_{k=1}^N y^2(k)$ , standard deviation (SD),  $\sigma = \sum_{k=1}^N (y(k) - \hat{y}(k))^2 / N$ . Means of the number of wavelets, ERRs and SDs are shown in Table 1 and 2. It can be seen that the wavelet networks constructed using the proposed approach have less number of nodes and better generalization capabilities than the networks in [2].

**Table 1.** Comparison of the prediction result of the proposed WNNs with the result in [2]

Algorithms	Number of nodes	ERR	SD
RBNN <sub>2</sub> [2]	20	0.0090	0.0033
WBNN <sub>3</sub> [2]	14	0.0091	0.0033
WBNN <sub>4</sub> [2]	12	0.0090	0.0033
GA-WNN	4	0.0052	1.92e-3

**Table 2.** Generalization performance of the WNNs using the proposed GA algorithm

Measures	$x_1 = \sqrt{2}/2$	$x_2 = \sqrt{3}/3$	$x_3 = 1/11$	$x_4 = 4/7$	$x_5 = 8/9$
EER, %	0.0049	0.0050	0.0051	0.0051	0.0051
SD( $\sigma$ )	1.86e-3	1.88e-3	1.89e-3	1.89e-3	1.89e-3

## 5 Conclusion

In this paper, a new constructive algorithm is proposed for WNNs, which starts with no wavelet in the WNN and adds new wavelets from low resolution level to high resolution level. At each resolution, the translation parameters of a new wavelet are trained using GA. Since GA is a directed random search technique, local optima trap problem of gradient-type algorithm is avoided. Moreover, the proposed GA Optimized WNNs can be used for high dimensional problems.

## References

1. Zhang, Q., Benveniste, A.: Wavelet Networks. IEEE Trans. on Neural Networks **3** (1992) 889-898
2. Alonge, F., Dippolite, F., Raimondi, F.M.: System Identificaton via Optimised Wavelet-Based Neural Networks. IEE Proc.-Control Theory Appl. **150**(2) (2003) 147-154
3. Billings, S.A., Wei, H.: A new Class of Wavelet Networks for Nonlinear System Identification. IEEE Trans. on Neural Networks **16**(4) (2005) 862-874
4. Zhang, Q.: Using Wavelet Network in Nonparametric Estimation. IEEE Trans. on Neural Networks **8** (1997) 227-236
5. Xu, J., Ho, D.W.C.: A Basis Selection Algorithm for Wavelet Neural Networks. Neurocomputing **48** (2002) 681-689
6. Yao, X.: Evolving Artificial Neural Networks. Proc. IEEE **87**(9) (1999) 1423-1447

# The Optimal Solution of TSP Using the New Mixture Initialization and Sequential Transformation Method in Genetic Algorithm\*

Rae-Goo Kang and Chai-Yeoung Jung\*\*

Dept. Computer Science & Statistic, Chosun University,  
375 Seoseok-dong Dong-gu Gwangju, Korea  
kangrg@hanmail.net, cyjung@chosun.ac.kr

**Abstract.** TSP is a problem finding out the shortest distance out of possible courses where one starts a certain city and turns back to a starting city, visiting every city only once among  $N$  cities. This paper proposes the new method using both population initialization and sequential transformation method at the same time and then proves the improvement of capability by comparing them with existing methods.

**Keywords:** Genetic Algorithm, GA, TSP, Initialization.

## 1 Introduction

TSP is the problem to find out the shortest distance out of possible courses where one starts a certain city and, turns back to a starting city, visiting every city only once as  $N$  cities and the distances between cities are given.

The investigation space of TSP is  $\{T_1, T_2, \dots, T_n\}$ , the set of all traveling, and the size of it is  $N!$ . The solution is the shortest travelling distance. TSP is applied to various kinds of fields such as optimization of network and problems deciding the process order in a factory and home-delivery courses etc.[1][2][3].

This paper draws the new Mixture Initialization using both Random Initialization and induced Initialization Population Initialization, which should be preceded to apply GA to TSP.

Also, it has the nearest solution to the optimal solution found out by sequential transformation method when applying Selection Operator.

## 2 Genetic Algorithm in the Experiment

### 2.1 Selection Operator

The common rule is that the probability for superior solution to be chosen should be high although various selection operators have been presented so far. This paper uses the Roulette Wheel selection and Stochastic universal sampling operators.

---

\* This study was supported by research funds from Chosun University, 2006.

\*\* Correspondent author.

Roulette Wheel selection, the most representative operator, solves the selection probability of each objective  $i$  by summing up  $f(i), i=1,2,3,\dots,N$ ,  $f$  means the fitness of each objective  $i$  (fitness should not be 0.) With Stochastic universal sampling, every objective can be chosen for the same probability.

## 2.2 Crossover Operator

Crossover Operator is the most various and representative operator in GA. PMX, CX, OX, and Edge Recombination(ER) was used in this paper.[4]

Unlike PMX, ER, CX and OX, a kind of Heuristic Crossover Operator by Grefensette, is the way using the edge information around parent generation, not the existing way deciding the gene through the genes in parent generation. And this was introduced by Starkweathe.[5][6]

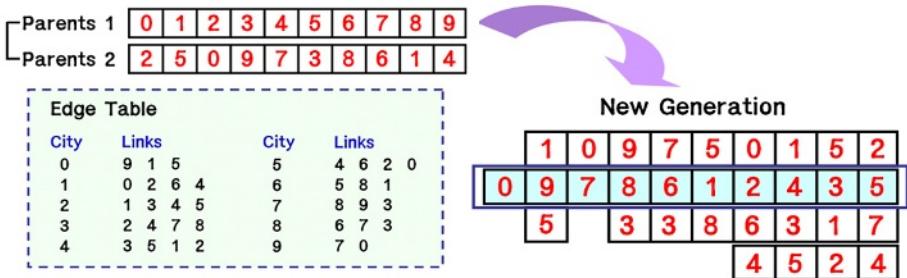


Fig. 1. Edge Recombination

Fig 1 shows the operation process of ER operation, one of the most being-used operators out of Crossover Operator. At first, one chooses one city at random (In Fig1, city 0 is chosen.) and then, finds the city, which has the least number of linkable edges. If the number of edges is the same, a certain city is chosen at random. After finishing all these processes, the new generation can be obtained.

## 2.3 Mutation Operator

Each population becomes stronger and move look-alike by Selection Operators and Crossover Operators. However, the move the generation goes down, the less the variety of genes is. Mutation Operators is used to compensate these faults.

With Mutation Operators, New Population can be made by preventing a specific bit from fixing from the early generation.

In this paper, Swapping mutation and Inversion are used out of Mutation Operators.

## 3 Proposed Methods

This paper proposes Mixture Initialization and Sequential Transformation method to obtain a superior solution of TSP.

### 3.1 Mixture Initialization

The first Population Initialization is more important than any other thing to get the nearest value to the optimal solution. There are two methods in Population Initialization. One is Random Initialization where population is produced by extracting at random without any rules. And the other is Induced Initialization where population is produced consistently by using background knowledge and information relating to given values. Random Initialization has been used mostly among Population Initialization methods for TSP. This paper proposes Mixture Initialization using both Random Initialization and Induced Initialization at the same time. (Random Initialization. Uses a random generator and Induced Initialization. Is based on background knowledge or experience.)

One chooses a starting city through random generator, and lists cities orderly from the city with the shortest distance to the city with the farthest distance, referring already-known information about distance among cities. If  $N$  cities are listed like this order,  $N \times N$  matrix is formed. This matrix was as the first population.

### 3.2 Sequential Transformation Method

Before Selection Operator operates in each generation, this paper proposes Sequential Transformation method, which changes the solution of population produced in the former generation sequentially, and applies the selection operator.

In this methods, the probability for the superior solution to be chosen becomes high when the population to be used in next generation is produces.

That's because populations are rearranged sequentially in the order of population which produces the best solution by the solution of the former generation

## 4 Consequence of Experiment

To analyze the results of Mixture Initialization and Sequential Transformation method proposed in this paper, 2 Selection Operators, 4 Crossover Operators, 2 Mutation Operators were used and 2 superior genes were preserved by using elitism.

We designed the number of total cities 100, the initialization of population size 1000, the number of total generations unlimited and the operation automatically ended if the value of solution is not changed during 100 generations. And  $P_c$  and  $P_m$  was used 0.7 and 0.1. This experiment was realized by using PowerBuilder8.0 based on Windows XP in P-4 2.4GHz and data was saved and analyzed by Oracle 9i.

Fig 2 shows the moment we obtain the best result. This result was obtained by TSP newly developed with PowerBuilder8.0 to prove the capability of two methods propose in this paper.

As shown in Table 1, 2664Km is the shortest distance by mainly used Random Initialization, but the better result is obtained by Mixture Initialization and Sequential Transformation method newly-proposed in this paper: the shortest distance is 2528Km. It is shown that the distances by newly-proposed methods are always shorter in every result.

And also, these two methods have achieved maximum improvement rate 10.1%, minimum 0.2%, and average 6.2%.

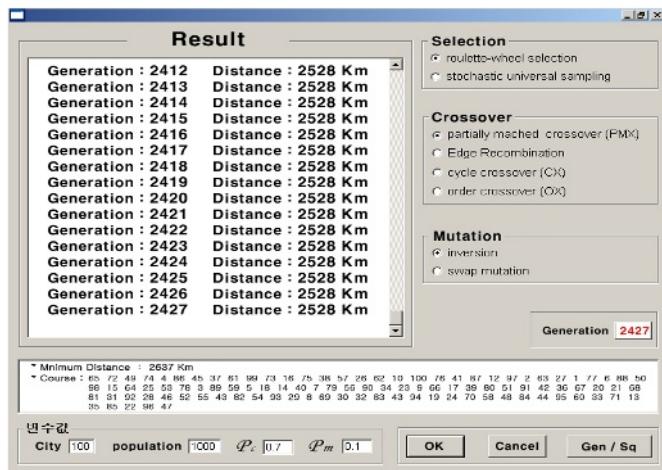


Fig. 2. Application used in experiment

Table 1. comparison of experiment result

Selection	Crossover	Mutation	newly-proposed method					
			Min	Avg.	gen	Min	Avg.	gen
Roulette Wheel	PMX	Inversion	2776	3812.2	2650	<b>2528</b>	<b>3332.1</b>	<b>2427</b>
		Swapping	3328	4215	3520	3185	3965.1	3320
	ER	Inversion	3521	4996.1	3358	3217	4402.3	3256
		Swapping	2998	3512	2658	2698	3168	2756
	CX	Inversion	<b>2664</b>	<b>3081.2</b>	<b>2375</b>	2618	3990.8	3012
		Swapping	2968	3325.2	2451	2812	4065.1	2405
	OX	Inversion	3028	3521.2	3302	2786	3302.1	3598
		Swapping	2908	3865.5	2785	2901	4002	2741
Stochastic universal sampling	PMX	Inversion	3054	3344.5	2556	3015	4812.1	2588
		Swapping	4215	3702	3322	3854	4102.2	3025
	ER	Inversion	3524	3561.1	3158	3222	5021	2930
		Swapping	3698	2968.7	2745	3213	5502.6	3302
	CX	Inversion	2967	3206.2	2566	2851	4892.1	2547
		Swapping	4015	3345	2856	3874	1561.9	2635
	OX	Inversion	3025	3625	3254	3005	5009	3010
		Swapping	3332	3478	3023	2995	5821.1	2998

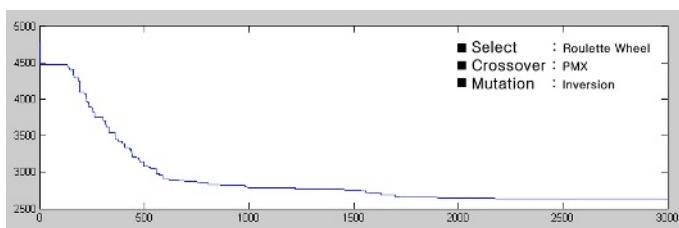


Fig. 3. Optimal value graph using newly-proposed method

## 5 Conclusion

This paper proposes two methods more effectively to solve TSP.

One is Mixture Initialization using Random Initialization and Induced Initialization at the same time, and the other is Sequential Transformation method rearranging objectives in each generation and heightening the probability to only superior genes to be chosen. With these two methods, average improvement rate 6.2% was obtained and superior values produced from 1generation unlike existing methods.

So to speak, improving efficiency is helped by the method using the given information of distances between cities and the method heightening probability for only superior genes to be chosen through rearranging objectives.

The methods proposed in this paper was experimented with existing GA alone, thus if a new algorithm is applied and cities are lager or more complicated, another study should be done continually to obtain the optimal value.

## References

1. Jin gang gou.: Genetic Algorithm and the application, kousa (2000)
2. Goldberg, D.: Genetic Algorithms in search, Optimization, and Machine Learning Addison Wesley, Reading, MA (1989)
3. K.D. Boese, "Cost Versus Distance In the Traveling Salesman Problem", Technical Report CSD-950018, UCLA (1995)
4. Michalewicz, Z.: Genetic Algorithms + Data Structures = Evolution Programs, Springer-Verlag (1992)
5. Grefenstette, J. Gopal,R. Rosmaita, B and Gucht, D.: "Genetic Algorithms for the Traveling Salesman Problem", Proc. the 1st Inter. Conf. on GAs and Their Applications (1985) 160-168
6. Whitley, D. Starkweather, T and Fuquay, D.: "Scheduling problems and traveling salesman: the genetic edge recombination and operator", Proc. Third Int. Conf. G As (1989) 133-140

# Steering Law Design for Single Gimbal Control Moment Gyroscopes Based on RBF Neural Networks

Zhong Wu<sup>1</sup>, Wusheng Chou<sup>2</sup>, and Kongming Wei<sup>1</sup>

<sup>1</sup> School of Instrumentation Science and Optoelectronics Engineering  
Beijing University of Aeronautics and Astronautics, Beijing 100083, China  
wuzhong\_buaa@163.com, weikongming@aspe.buaa.edu.cn

<sup>2</sup> School of Mechanical and Automation Engineering  
Beijing University of Aeronautics and Astronautics, Beijing 100083, China  
wschou@buaa.edu.cn

**Abstract.** Usually, the pseudo-inverse of the Jacobian matrix needs to be calculated in the conventional laws for the Single Gimbal Control Moment Gyroscopes (SGCMGs). However, the steering law can not work when the Jacobian matrix is singular and its pseudo-inverse is indefinite. To avoid the conditions stated above, a new steering law is designed using radial basis function(RBF) neural networks. This algorithm can output the desired gimbal angles directly according to the momentum command. And also, this algorithm can deal with the singular conditions since the pseudo-inverse of the Jacobian matrix is not needed. Simulation results demonstrate the effectiveness of the steering law.

## 1 Introduction

Single Gimbal Control Moment Gyroscopes (SGCMGs) are torque-producing devices which are mounted inside a spacecraft and operate based on principles of momentum exchange. Due to their torque amplification features and simple construction, SGCMGs have wide application perspectives in the field of spacecraft control. Generally, three or more SGCMGs are often used to meet the need of the 3-axis attitude control. In order to control the spacecraft attitude accurately using SGCMGs, high-performance steering laws should be designed.

Usually, the pseudo-inverse of the Jacobian matrix needs to be calculated in the conventional steering laws<sup>[1-7]</sup>. However, the steering law can not work when the Jacobian matrix is singular and its pseudo-inverse is indefinite. Although the steering laws based on singularity-robust inverse can escape from the singular points, they can not produce exact torque for the attitude control and unexpected attitude oscillations will be resulted from the steering errors<sup>[4-7]</sup>. Therefore, Krishnan and Vadali<sup>[8]</sup> presented a steering law based on the transpose of the Jacobian matrix, which can provide viable gimbal rates for SGCMGs even when the gimbals pass through a singular configuration. Paradiso<sup>[9]</sup> adopted a global search technology to produce singularity-avoiding feedforward gimbal trajectories in response to a command history forecast from a momentum management or maneuver scheduler. Nevertheless, both of them have a heavy burden in calculation and can not be implemented easily.

Actually, the problem of the steering law design can be regarded as an inverse to a nonlinear map—SGCMG kinematics. If the inverse map can be obtained, then the steering law design is fulfilled. Therefore, the problem of the steering law design can be transformed into the study on the inverse kinematics. In this paper, radial basis function(RBF) neural networks are utilized to approximate the inverse since it can approximate any nonlinear continuous function with arbitrary degree of accuracy. Thus, the steering law based on RBF neural networks is a global method and can output the desired gimbal angle directly in response to the momentum command. The rest parts of this paper are organized as follows. In section 2, a brief description of kinematics for SGCMG systems is given. Section 3 will describe how to design a steering law based on RBF neural networks. In section 4, simulation results are presented to verify the steering law. Conclusions are made in section 5.

## 2 SGCMG Kinematics and Problem Statement

For conveniences, assume  $n$  identical SGCMGs are used in the attitude control system of the spacecrafts( $n > 3$ ). Let  $\sigma, h$  denote the gimbal angle and the angular momentum vector respectively, then the kinematics of the SGCMG systems can be written as<sup>[2-3]</sup>:

$$h(t) = f(\sigma(t)) \quad (1)$$

and correspondingly,

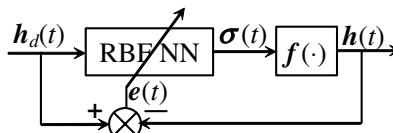
$$\dot{h}(t) = J(\sigma(t))\dot{\sigma}(t) \quad (2)$$

where  $f(\cdot)$  is a certain nonlinear function, Jacobian matrix  $J(\sigma(t)) = \partial f(\sigma(t))/\partial \sigma(t)$ .

## 3 Steering Law Design Based on RBF Neural Networks

In order to avoid the calculation of the pseudo-inverse of the Jacobian matrix, the steering law can be designed through the direct inverse to (1). RBF neural networks can be employed to approximate the inverse since  $f$  is a complex nonlinear function and it is impossible to obtain its inverse in analytical forms. Furthermore, RBF networks should be adjusted on line to guarantee the steering performance, as shown in Fig.1. This network has 3 input nodes and  $n$  output nodes. The weight  $w_{ji}$  can be adjusted on line according to the steering errors. The gimbal angle is given by<sup>[10]</sup>:

$$\sigma_i = w_{0i} + \sum_{j=1}^3 w_{ji} \varphi_j(|h_d - c_j|), \quad i=1, 2, \dots, n \quad (3)$$



**Fig. 1.** Steering law based on RBF neural networks

where  $\mathbf{h}_d(t)$  is the desired momentum trajectory,  $w_{ji}$  is the weight between the  $j$ th hidden node and the  $i$ th output node,  $\varphi_j$  is the radial basis function, and  $\mathbf{c}_j$  is the center vector of the function,  $j=1,2,3$ . Let  $\boldsymbol{\sigma}=[\sigma_1, \dots, \sigma_n]^T$ ,  $\mathbf{W}_0=[w_{01}, \dots, w_{0n}]^T$ ,  $\mathbf{W}=(w_{ij})_{n \times 3}$ ,  $\boldsymbol{\varphi}=[\varphi_1(|\mathbf{h}_d - \mathbf{c}_1|), \dots, \varphi_3(|\mathbf{h}_d - \mathbf{c}_3|)]^T$ , then we can change (3) into the following form:

$$\boldsymbol{\sigma} = \mathbf{W}_0 + \mathbf{W}\boldsymbol{\varphi} \quad (4)$$

Choose a performance index as:

$$E = \mathbf{e}^T \mathbf{e} / 2 \quad (5)$$

where  $\mathbf{e}=\mathbf{h}_d - \mathbf{h}$ . Assume  $\mathbf{h}_d$  varies very slowly, then

$$\partial \mathbf{e} / \partial \boldsymbol{\sigma} \approx -\partial \mathbf{h} / \partial \boldsymbol{\sigma} = -\mathbf{J} \quad (6)$$

Take the time derivative of  $E$  along (2), (4), and (6), then we can get

$$\dot{E} = \mathbf{e}^T \dot{\mathbf{e}} = \mathbf{e}^T \partial \mathbf{e} / \partial \boldsymbol{\sigma} \dot{\boldsymbol{\sigma}} = -\mathbf{e}^T \mathbf{J} \dot{\mathbf{W}} \boldsymbol{\varphi} \quad (7)$$

Let

$$\dot{\mathbf{W}} = \alpha \mathbf{J}^T \mathbf{e} \boldsymbol{\varphi}^T \quad (8)$$

Then, (7) can be changed into

$$\dot{E} = -\alpha \mathbf{e}^T \mathbf{J} \mathbf{J}^T \mathbf{e} \boldsymbol{\varphi}^T \boldsymbol{\varphi} \leq 0 \quad (9)$$

where  $\alpha$  is a positive constant. From (9), it is easy to know that function  $E$  has a lower bound. Therefore, we can use (8) as the update law of the weight matrix  $\mathbf{W}$  to make the steering error  $\mathbf{e}$  converge to zero asymptotically. Additionally, the update law (8) also can be written in the form of  $w_{ji}$ , i.e.

$$\dot{w}_{ji} = \alpha J_i^T \mathbf{e} \varphi_j(|\mathbf{h}_d - \mathbf{c}_j|), \quad i=1,2,\dots,n, j=1,2,3 \quad (10)$$

where  $J_i = \partial h / \partial \sigma_i$  is the  $i$ th column vector of  $J$ .

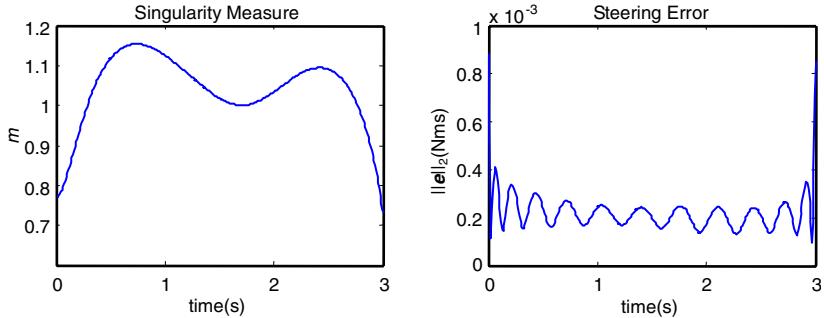
Before the use of the steering law based on RBF networks, it should be trained first using enough sample data. For redundant SGCMGs, it is very difficult to determine the sample space because a fixed momentum  $\mathbf{h}$  corresponds to a manifold in gimbal angle space. Therefore, the sample trajectories can be used in the generation of the sample data. For example, we can choose enough momentum trajectories in momentum space first. Then, the corresponding gimbal angle trajectories can be determined using global optimization techniques<sup>[9]</sup> to guarantee the avoidance of all the internal singularities. Thus, the momentum and the gimbal angle trajectories can compose the sample space to train the network. Obviously, the use of the sample trajectories can eliminate the problems resulted from the nonlinear surjective maps (SGCMG kinematics), and can improve the performance of the singularity avoidance.

## 4 Simulation Results

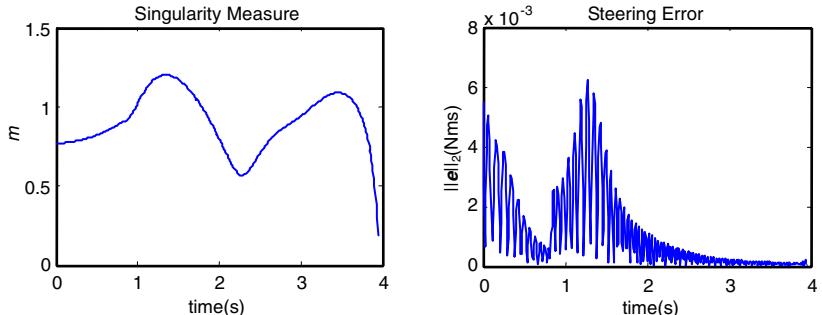
To demonstrate the effectiveness of the steering law based on RBF neural networks, a simulation study was performed concerning a pyramid-type SGCMG system with the following kinematics<sup>[3]</sup>:

$$\begin{cases} h_1 = h_0(-c_\beta s_1 - c_2 + c_\beta s_3 + c_4) \\ h_2 = h_0(c_1 - c_\beta s_2 - c_3 + c_\beta s_4) \\ h_3 = h_0 s_\beta (s_1 + s_2 + s_3 + s_4) \end{cases} \quad (11)$$

where  $s_\beta = \sin\beta$ ,  $c_\beta = \cos\beta$ ,  $s_i = \sin\sigma_i$ ,  $c_i = \cos\sigma_i$ ,  $i=1, \dots, 4$ ,  $\beta=54.74^\circ$ . Assume that each SGCMG has unit momentum ( $h_0=1$ ), the initial gimbal angle  $\sigma_0=[45^\circ, -45^\circ, 45^\circ, -45^\circ]^T$ ,  $\alpha=0.2$ , simulation results can be obtained in Fig.2~Fig.3 under different momentum commands, where the singularity measure  $m=\text{sqrt}(\mathbf{J}\mathbf{J}^T)$ .



**Fig. 2.** Simulation curves when  $\mathbf{h}_d=0.7071[t, t, 0]^T$



**Fig. 3.** Simulation curves when  $\mathbf{h}_d=0.7071[t, t, 0]^T$ ,  $t \leq 0.83$ ;  $\mathbf{h}_d=0.7071[(1.66-t), t, 0]^T$ ,  $t > 0.83$

From Fig.2, it can be seen that no internal singularities are encountered during the whole steering procedure. The maximal steering error is less than 0.04%. When the steering procedure approaches the saturation singular points ( $t>3$ s), the steering error will increase rapidly, and desaturation measure should be used. Fig.3 also shows that there are no internal singularities during the steering procedure. Although the sudden variation ( $t=0.83$ s) of the momentum command can result in the increase of the steering error, the error can converge to zero rapidly and the maximal error is only 0.6%.

Compared with the conventional steering laws<sup>[2-7]</sup>, this one does not need the calculation of the pseudo-inverse of the Jacobian matrix and can deal with the singular conditions easily.

## 5 Conclusion

In order to eliminate the problems resulted from the calculation of the pseudo-inverse of the Jacobian matrix in the conventional steering laws, a new steering law is designed for redundant SGCMG systems using RBF neural networks. Since RBF networks can approximate any nonlinear functions with arbitrary precision and without local minimum problems, the steering law of this paper has good performance in accuracy and singularity avoidance. Furthermore, this algorithm can output the desired gimbal angle directly according to the momentum command and the pseudo-inverse of the Jacobian matrix is not needed.

## References

1. Lappas, V.J., Steyn, W.H., Underwood C.I.: Control moment gyro (CMG) gimbal angle compensation using magnetic control during external disturbances. *Electronics Letters*, 37(9) (2001) 603-604
2. Wu, Z., Wu, H.X.: Survey on steering laws for single gimbal control moment gyroscope systems. *Journal of Astronautics*, 21(4) (2000) 140-145 (in Chinese)
3. Bedrossian, N.S.: Steering law design for redundant single gimbal control moment gyro systems. *M.S.Thesis*, Mechanical Engineering, Massachusetts Inst. of Technology, Cambridge, MA, Aug., (1987)
4. Wie, B., Bailey, D., Heiberg, C.: Singularity robust steering logic for redundant single-gimbal control moment gyros. *Journal of Guidance, Control and Dynamics*, 24(5) (2001) 865-872
5. Ford, K.A., Hall, C.D.: Singular direction avoidance steering for control-moment gyros. *Journal of Guidance, Control, and Dynamics*, 23(4) (2000) 648-656
6. Jung, D., Tsotras, P.: An experimental comparison of CMG steering control laws. In: *Collection of Technical Papers - AIAA/AAS Astrodynamics Specialist Conference*, Providence, RI, (2) (2004) 1128-1144
7. Wie, B.: Singularity escape/avoidance steering logic for control moment gyro systems. *Journal of Guidance, Control and Dynamics*, 28(5) (2001) 948-955
8. Krishnan, S., Vadali, S.R.: An inverse-free technique for attitude control of spacecraft using CMGs. *Acta Astronautica*, 39(6) (1996) 431-438
9. Paradiso, J.A.: Global Steering of Single Gimbaled Control Moment Gyroscopes Using a Directed Search. *Journal of Guidance, Control and Dynamics*, 15(5) (1992) 1236-1244
10. Yang, M., Li, J.G., Lu, G.Z.: The inverse kinematics control algorithm based on RBF neural networks for manipulators. In: *Proceedings of the 5th World Congress on Intelligent Control and Automation*, Hangzhou, China, (2004) 855-859

# Automatic Design of Hierarchical RBF Networks for System Identification

Yuehui Chen<sup>1</sup>, Bo Yang<sup>1,2</sup>, and Jin Zhou<sup>1</sup>

<sup>1</sup> School of Information Science and Engineering  
Jinan University, Jinan 250022, P.R. China  
[yhchen@ujn.edu.cn](mailto:yhchen@ujn.edu.cn)

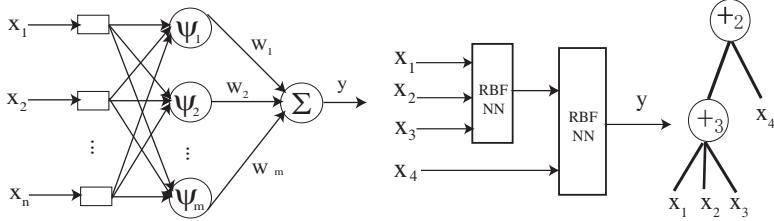
<sup>2</sup> State Key Lab. of Advanced Technology for Materials Synthesis and Processing,  
Wuhan University of Science and Technology, Wuhan, China  
[yangbo@ujn.edu.cn](mailto:yangbo@ujn.edu.cn)

**Abstract.** The purpose of this study is to identify the hierarchical radial basis function neural networks and select important input features for each sub-RBF neural network automatically. Based on the pre-defined instruction/operator sets, a hierarchical RBF neural network is created and evolved by using Extended Compact Genetic Programming (ECGP), and the parameters are optimized by Differential Evolution (DE) algorithm. Empirical results on benchmark system identification problems indicate that the proposed method is efficient.

## 1 Introduction

Hierarchical Neural Network (HNN) is neural network architecture in which the problem is divided and solved in more than step [1]. Ohno-Machado divide hierarchical network into two architectures that are bottom-up and top-down architectures [1]. Many version of HNN have been introduced and applied in various applications [1][3][4][5]. Erenshteyn and Laskov examine the application of hierarchical classifier to recognition of finger spelling [2]. They refer hierarchical NN as multi-stage NN. The approach aimed to minimize the network's learning time without reducing the accuracy of the classifier. Mat Isa et al. used Hierarchical Radial Basis Function (HiRBF) to increase RBF performance in diagnosing cervical cancer [3]. HiRBF cascading together two RBF networks, where both network have different structure but using the same algorithms. The first network classifies all data and performs a filtering process to ensure that only certain attributes to be fed to the second network. The study shows that HiRBF performs better compared to single RBF. HRBF has been proved effective in the reconstruction of smooth surfaces from sparse noisy data points [5].

In this paper, an automatic method for constructing HRBF network is proposed. Based on the pre-defined instruction/operator sets, a HRBF network can be created and evolved. HRBF allows input variables selection. In our previous studies, in order to optimize Flexible Neural Tree (FNT) the hierarchical structure was evolved using Probabilistic Incremental Program Evolution algorithm (PIPE) with specific instructions [6][7] and Ant Programming [8]. In this



**Fig. 1.** A basis function operator (left), and a tree-structural representation of a hierarchical RBF neural network with function instruction set  $F = \{+2, +3, +4, +5, +6\}$ , and terminal instruction set  $T = \{x_1, x_2, x_3\}$  (right)

research, the hierarchical structure is evolved using the Extended Compact Genetic Programming. The fine tuning of the parameters encoded in the structure is accomplished using Differential Evolution (DE). The novelty of this paper is in the usage of HRBF model for selecting the important variables and for improving the accuracy of system identification.

## 2 The Hierarchical RBF Model

The function set  $F$  and terminal instruction set  $T$  used for generating a hierarchical RBF model are described as  $S = F \cup T = \{+2, +3, \dots, +N\} \cup \{x_1, \dots, x_n\}$ , where  $+i (i = 2, 3, \dots, N)$  denote non-leaf nodes' instructions and taking  $i$  arguments.  $x_1, x_2, \dots, x_n$  are leaf nodes' instructions and taking no other arguments. The output of a non-leaf node is calculated as a RBF neural network model (see Fig.1). From this point of view, the instruction  $+i$  is also called a basis function operator with  $i$  inputs.

The basis function operator is shown in Fig.1(left). In general, the basis function networks can be represented as  $y = \sum_{i=1}^m \omega_i \psi_i(x; \theta)$ , where  $x \in R^n$  is input vector,  $\psi_i(x; \theta)$  is  $i$ th basis function, and  $\omega_i$  is the corresponding weights of  $i$ th basis function and  $\theta$  is the parameter vector used in the basis functions. In this research, Gaussian radial basis functions are used,  $\psi_i(x; \theta) = \prod_{j=1}^n \exp(-\frac{\|x_j - b_j\|^2}{a_j^2})$  and the number of basis functions used in hidden layer is same with the number of inputs, that is,  $m = n$ .

**Tree Structure Optimization.** Finding an optimal or near-optimal HRBF is formulated as a product of evolution. In this paper, the Extended Compact Genetic Programming (ECGP) [9] is employed to find an optimal or near-optimal HRBF structure. ECGP is a direct extension of ECGA to the tree representation which is based on the PIPE prototype tree. In ECGA, Marginal Product Models (MPMs) are used to model the interaction among genes, represented as random variables, given a population of Genetic Algorithm individuals. MPMs are represented as measures of marginal distributions on partitions of random variables. ECGP is based on the PIPE prototype tree, and thus each node in the prototype tree is a random variable. ECGP decomposes or partitions the prototype tree into sub-trees, and the MPM factorises the joint probability of all

nodes of the prototype tree, to a product of marginal distributions on a partition of its sub-trees. A greedy search heuristic is used to find an optimal MPM mode under the framework of minimum encoding inference. ECGP can represent the probability distribution for more than one node at a time. Thus, it extends PIPE in that the interactions among multiple nodes are considered.

**Parameter Optimization with DE Algorithm.** The DE algorithm was first introduced by Storn and Price in 1995 [10]. In generation  $k$ , we denote the population members by  $x_1^k, x_2^k, \dots, x_N^k$ . The DE algorithm is given as follows [11]:

- 1) Set  $k = 0$ , and randomly generate  $N$  points  $x_1^0, x_2^0, \dots, x_N^0$  from search space to form an initial population;
- 2) For each point  $x_i^k (1 \leq i \leq N)$ , execute the DE offspring generation scheme to generate an offspring  $x_i^{(k+1)}$ ;
- 3) If the given stop criteria is not met, set  $k = k + 1$ , goto step 2).

The DE Offspring Generation approach used is given as follows,

- 1) Choose one point  $x_d$  randomly such that  $f(x_d) > f(x_i^k)$ , another two points  $x_b, x_c$  randomly from the current population and a subset  $S = \{j_1, \dots, j_m\}$  of the index set  $\{1, \dots, n\}$ , while  $m < n$  and all  $j_i$  mutually different;
- 2) Generate a trial point  $u = (u_1, u_2, \dots, u_n)$  as follows:

**DE Mutation.** Generate a temporary point  $z$  as follows,

$$z = (F + 0.5)x_d + (F - 0.5)x_i + F(x_b - x_c); \quad (1)$$

Where  $F$  is a give control parameter;

**DE Crossover.** for  $j \in S$ ,  $u_j$  is chosen to be  $z_j$ ; otherwise  $u_j$  is chosen

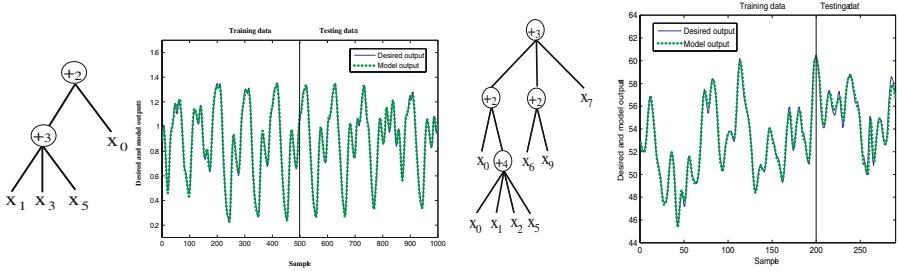
a to be  $(x_i^k)_j$ ;

- 3) If  $f(u) \leq f(x_i^k)$ , set  $x_i^{k+1} = u$ ; otherwise, set  $x_i^{k+1} = x_i^k$ .

### 3 Simulation Studies

In this research, benchmark problems Mackey-Glass and Jenkins-Box time-series are employed to evaluate of the performance of the proposed method. For Mackey-Glass time-series, we predict the  $x(t + 6)$  with using the inputs variables  $x(t), x(t - 6), x(t - 12), x(t - 18), x(t - 24)$  and  $x(t - 30)$ . The evolved HRBF tree and the actual time-series data, the output of HRBF model and the prediction error are shown in Fig.2 (left). A comparison result of different methods for forecasting Mackey-Glass data is shown in Table 1 (left).

For Jenkins-Box time-series, 10 inputs variables are used for constructing a HRBF model. The proper time-lags for constructing a HRBF model are finally determined by an evolutionary procedure. The evolved HRBF tree and the actual time-series, the HRBF model output and the prediction error are shown in Fig.2 (right). From the evolved HRBF tree, it can be seen that the optimal inputs



**Fig. 2.** The evolved architecture of HRBF model for prediction of the Mackey-Glass time-series(left), and the actual time series data, output of the evolved HRBF model and the prediction error(right)

variables for constructing a HRBF model are:  $u(t-2)$ ,  $u(t-3)$ ,  $u(t-4)$ ,  $u(t-6)$ ,  $y(t-1)$ ,  $y(t-2)$  and  $y(t-3)$ . It should be noted that the HRBF model with proper selected input variables has accurate precision and good generalization ability. A comparison result of different methods for forecasting Jenkins-Box data is shown in Table 1 (right). From the above simulation results, it can be seen that the proposed HRBF model works well for generating prediction models of time series.

**Table 1.** Comparison of prediction errors using different methods for the Mackey-Glass and Gas furnace time-series problem

Mackey-Glass		Gas Furnace	
Method	RMSE	Method	MSE
RBF [12]	0.0114	ANFIS [14]	0.0073
GA+Fuzzy [13]	0.049	FuNN [15]	0.0051
FNT1 [7]	0.0069	FNT1 [7]	0.00066
FNT2 [7]	0.0027	FNT2 [7]	0.00029
HRBF	0.0076	HRNF	0.0012

## 4 Conclusions

Based on a novel representation and calculation of the hierarchical RBF models, an approach for evolving the HRBF was proposed in this paper. The hierarchical architecture and inputs selection method of the HRBF were accomplished using ECGP algorithm, and the free parameters embedded in the HRBF model were optimized using DE algorithm. Simulation results shown that the evolved HRBF models are effective for the time-series prediction problems. Our future works will concentrate on applying the proposed approach for more complex problems.

## Acknowledgment

This research was partially supported the Natural Science Foundation of China under grant No. 60573065.

## References

1. Ohno-Machado, L.: Medical Applications of Artificial Neural Networks: Connectionist Model of Survival. Ph.D Dissertation. (1996) Stanford University
2. Erenshteyn, R., Laskov, P.: A multi-stage approach to fingerspelling and gesture recognition. In: Proceedings of the Workshop on the Integration of Gesture in Language and Speech. (1996) 185-194
3. Mat Isa, N. A., Mashor, M. Y. and Othman, N. H.: Diagnosis of Cervical Cancer using Hierarchical Radial Basis Function (HiRBF) Network. In: Sazali Yaacob, R. Nagarajan, Ali Chekima (Eds.), Proceedings of the International Conference on Artificial Intelligence in Engineering and Technology. (2002) 458-463
4. Ferrari, S., Maggioni, M., Alberto Borghese, N.: Multiscale Approximation With Hierarchical Radial Basis Functions Networks. IEEE Trans. on Neural Networks. 15 (2004) 178-188
5. Ferrari, S., Frosio, I., Piuri, V. and Alberto Borghese, N.: Automatic Multiscale Meshing Through HRBF Networks. IEEE Trans. on Instrumentation and Measurement. 54 (2005) 1463-1470
6. Chen, Y., Yang B., and Dong, J.: Nonlinear System Modeling via Optimal Design of Neural Trees. International Journal of Neural Systems. 14 (2004) 125-137
7. Chen, Y., Yang, B., Dong, J. and Abraham, A.: Time-series Forecasting using Flexible Neural Tree Model. Information Science. 174 (2005) 219-235
8. Chen, Y., Yang, B., Dong, J.: Automatic Design of Hierarchical TS-FS Models using Ant Programming and PSO algorithm. Lectur Notes on Computer Science 3192. (2004) 285-294
9. Sastry K. and Goldberg, D. E.: Probabilistic model building and competent genetic programming. In R. L. Riolo and B. Worzel, editors, Genetic Programming Theory and Practise. (2003) 205-220 Kluwer, 2003.
10. Storn, R. and Price, K.: Differential evolution - a simple and efficient adaptive scheme for global optimization over continuous spaces. Technical report, International Computer Science Institute, Berkley, (1995)
11. Price, K.: Differential Evolution vs. the Functions of the 2nd ICEO. In: Proceedings of 1997 IEEE International Conference on Evolutionary Computation (ICEC'97), Indianapolis, USA. (1997) 153-157
12. Cho, K.B., Wang, B.H.: Radial basis function based adaptive fuzzy systems their application to system identification and prediction, Fuzzy Sets and Systems, 83 (1995) 325-339
13. Kim, D., kim, C.: Forecasting time series with genetic fuzzy predictor ensembles, IEEE Trans. Fuzzy Systems 5 (1997) 523-535
14. Jang, J.-S.R., Sun, C.-T. and Mizutani, E.: Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence, Upper saddle River, NJ:prentice-Hall (1997)
15. Kasabov, N., Kim, J.S., Watts, M. and Gray, A.: FuNN/2 - A fuzzy neural network architecture for adaptive learning and knowledge acquisition, Inforation Science. 101 (1996) 155-175

# Dynamically Subsumed-OVA SVMs for Fingerprint Classification

Jin-Hyuk Hong and Sung-Bae Cho

Dept. of Computer Science, Yonsei University

134 Sinchon-dong, Sudaemoon-ku

Seoul 120-749, Korea

hjinh@sclab.yonsei.ac.kr, sbcho@cs.yonsei.ac.kr

**Abstract.** A novel method to fingerprint classification, in which the naïve Bayes classifier (NB) and OVA SVMs are integrated, is presented. In order to solve the tie problem of combining OVA SVMs, we propose a subsumption architecture dynamically organized by the probability of classes. NB calculates the probability using singularities and pseudo codes, while OVA SVMs are trained on FingerCode. The proposed method not only tolerates ambiguous fingerprint images by combining different fingerprint features, but produces a classification accuracy of 90.8% for 5-class classification on the NIST 4 database, that is higher than conventional methods.

## 1 Introduction

Since the Henry system categorizes fingerprints by the relative position and number of core and delta points, many researchers have tried to extract them in the flow of the ridges [1]. Karu and Jain proposed a heuristic algorithm with singularities [2], while Zhang and Yan used singularities together with pseudo ridges to classify fingerprints [3]. In order to obtain a high classification rate, various features have been also actively investigated. Jain, *et al.* proposed FingerCode that uses a Gabor filter to extract the directional ridge flow [4], Park used the orientation filtered by fast Fourier transform [5], where Min, *et al.* proposed localized models of SVMs using FingerCode [6]. There are some other attempts to integrate several features and methods to produce a robust fingerprint classifier [1, 7]. Senior used hidden Markov models and decision trees to recognize the ridge structure of the print [1], while Yao, *et al.* combined flat and structured features using the recursive neural networks and support vector machines (SVMs) [7].

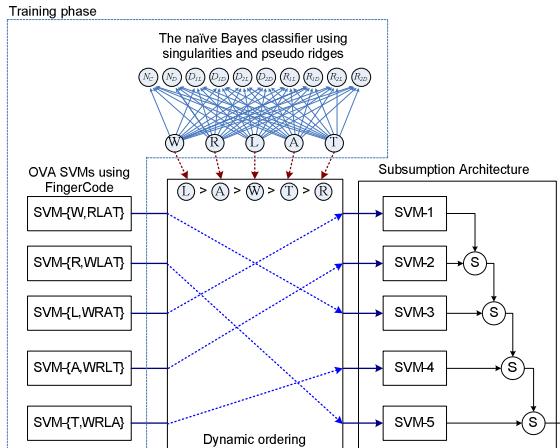
This paper describes a novel fingerprint classification approach integrating the naïve Bayes classifier (NB) and SVMs. In order to accomplish highly accurate classification, SVMs with FingerCode are generated based on the one-vs-all (OVA) scheme while NB with singularities dynamically organizes them.

## 2 A Dynamic Fingerprint Classifier

### 2.1 Overall Architecture

Contrary to conventional methods that have a static classification process, we propose a dynamic fingerprint classifier that not only uses various fingerprint features

(singularity, pseudo ridges and FingerCode) but also solves the ambiguity of OVA SVMs. The proposed method consists of NB and OVA SVMs as shown in Fig. 1. NB estimates the posterior probability for fingerprint classes  $prob = \{p_w, p_l, p_r, p_a, p_t\}$  by using singular points and pseudo ridges, while OVA SVMs classify fingerprints by using the FingerCode, where the margin of a sample  $o\text{-svm} = \{m_w, m_r, m_l, m_a, m_t\}$  is produced. We evaluate them based on the subsumption architecture to manage the ambiguity such as ties and rejects. The subsumption architecture selects an action when there are multiple models by sequentially evaluating each model. When a model is satisfied, it suppresses the other models.



**Fig. 1.** Overview of the proposed method

$prob[5] = \{p_w, p_r, p_l, p_a, p_t\}$ // $prob[]$ is calculated by the naïve Bayes classifier $order[5] = \{0, 1, 2, 3, 4\}$ $o\text{-svm}[5] = \{m_w, m_r, m_l, m_a, m_t\}$ // $o\text{-svm}[]$ is obtained by the OVA SVMs	// classify with OVA SVMs according to the subsumption architecture if( $prob[order[0]] < r_1$ ) // $r_1$ is a rejection threshold return reject;  for( $i=0; i<5; i++$ ) {     if( $o\text{-svm}[order[i]] >= a$ ) // $a$ is a threshold     {         if( $o\text{-svm}[order[i]] < r_2$ )         // $r_2$ is a rejection threshold return reject;         return $order[i]$ ;     } } return $order[0]$ ;
---	--

**Fig. 2.** Pseudo code for probabilistically ordering OVA SVMs

As shown in Fig. 1, the order of OVA SVMs to evaluate is determined by the posterior probability of each class which NB produces. The corresponding OVA SVM of a more probable class takes precedence in the subsumption architecture against the other OVA SVMs. A sample is evaluated sequentially until an OVA SVM is satisfied. When an OVA SVM satisfies, the sample is classified into the corresponding class of the OVA SVM, while it is classified into the class of the highest probability when none of OVA SVMs are satisfied. Fig. 2 shows the pseudo code for the proposed method. Ordering OVA SVMs properly for an input sample provides dynamic classification.

## 2.2 Fingerprint Features: FingerCode, Singularities and Pseudo Codes

In this paper, we use FingerCode to train OVA SVMs, and singularities and pseudo codes as the feature of the naïve Bayes classifier. FingerCode, proposed by Jain in 1999, is a representative fingerprint feature extracted from the NIST database 4 [4]. The algorithm sets a registration point in a fingerprint image and tessellates it into 48 sectors. The Gabor filter is applied with 4 directions ( $0^\circ$ ,  $45^\circ$ ,  $90^\circ$ , and  $135^\circ$ ) so as to accentuate the ridge parallel to each direction. Ridges that are not parallel to the direction will be blurred. Standard deviations are computed on 48 sectors for each of the 4 filtered images to generate the 192-dimensional feature vector called FingerCode.

The number and location of core and delta points are used to classify fingerprints, where the poincare index is the representative algorithm detect singular points [3, 8]. The number of cores and deltas are denoted as  $NC$  and  $ND$ , respectively. The nearest core to the center is denoted as  $C$  (if there is no core,  $C$  represents the center of the image). Pseudo ridges proposed by Zhang, *et al.* consist of a predefined number (100 in this paper) of points [3]. A pseudo ridge is composed of 200 points based on orientation in two opposite directions from the starting point  $C$ . The naïve Bayes classifier is learned with 5 mutually exclusive and exhaustive classes ( $W, R, L, A, T$ ) and 10 features ( $NC, ND, D_{1L}, D_{1D}, D_{2L}, D_{2D}, R_{1L}, R_{1D}, R_{2L}, R_{2D}$ ), where each class is linked with all features as shown in Table 1.

**Table 1.** Features of the naïve Bayes classifier

Feature	Definition	State
$N_C, N_D$	Number of core points and delta points	0, 1, 2
$D_{1L}, D_{2L}$	Location of delta point	0, 1, 2, 3, 4, Absent
$R_{1L}, R_{2L}$	Location of the end point of the pseudo ridge	0, 1, 2, 3, 4, turn
$D_{1D}, D_{2D}$	Distance between $C$ and delta points	1, 2, 3, Absent
$R_{1D}, R_{2D}$	Distance between $C$ and the end point of the pseudo ridge	1, 2, 3, turn

## 3 Experimental Results

### 3.1 Dataset

The NIST Database 4 is used to verify the proposed method [9]. It consists of 4,000 scanned images of  $512 \times 512$  resolution obtained from two impressions (F and S) of 2,000 fingerprints. Fingerprints are equally distributed into 5 classes, whorl (W), right

loop (R), left loop (L), arch (A) and tented arch (T). Due to the ambiguity in fingerprints, 350 fingerprints (17.5%) are cross-referenced with two classes. The first label is only considered in training SVMs and NB while both labels are used in the test. In the experiment, the fingerprints of the first impression are used as the training set (F0001~F2000), and the others construct the test set. FingerCode proposed by Jain is used after normalization from -1 to 1, where some rejected images are included in the training set (1.4%) and the test set (1.85) [4]. The LIBSVM package (available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>) provides SVMs using the Gaussian kernel with  $\sigma^2=0.0625$ .

### 3.2 Results and Analysis

The proposed method, that combines NB and SVMs dynamically, is aimed at classifying fingerprints by handling more subtle discriminations. Although the major classification task is performed by SVMs, NB works as assistant by ordering them. It results an accuracy of 90.8% that is higher than the other methods including previously published ones. As shown in Table 2, the proposed method is better than SVMs with winner-takes-all as well as NB. The confusion matrix is shown in Table 3, in which fingerprints are evenly distributed comparing with NB or SVMs.

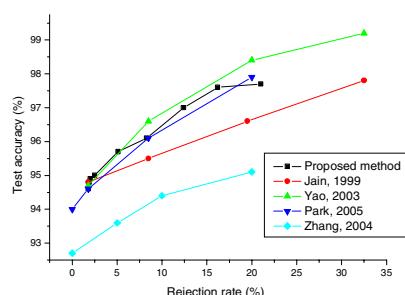
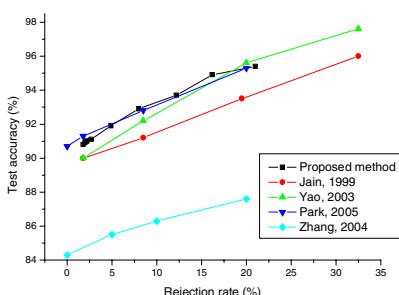
We have compared the proposed method with other methods that have previously been published. Several points are plotted in Fig. 3 along with the curve of possible

**Table 2.** Comparison with individual approaches

Method	Accuracy
NB	85.4%
SVMs (Winner-takes-all)	90.1%
NB+SVMs (Product)	90.2%
Proposed method	90.8%

**Table 3.** Confusion matrix of the proposed method

True class	Assigned class				
	W	R	L	A	T
W	373	10	10	0	0
R	4	374	0	6	15
L	5	0	377	8	9
A	0	6	4	365	40
T	1	8	15	39	295



**Fig. 3.** Five-class and four-class accuracy-rejection plots of the NIST 4 database

points for the proposed method here and the methods of Jain, *et al.* [4], Yao, *et al.* [7], Park [5], and Zhang and Yan [3]. For each method, the accuracy of the classifier is shown with the corresponding rejection rate. As shown in Fig. 3, the proposed method yields higher performance than the others in both classification tasks.

## 4 Conclusion

In this paper we have proposed a novel fingerprint classification method effectively integrating NB and OVA SVMs, which gives better accuracy than previously reported in the literature on the NIST-4 database. Several popular fingerprint features such as singularities, pseudo codes and FingerCode are used in the proposed method, where the combination of methods described here produces better results (90.8% for the five-class classification problem and 94.9% for the four-class classification problem with 1.8% rejection during the feature extraction phase of FingerCode) than any of the component classifiers. The propose method also solves tie problems which often occurs in multi-class classification by ordering OVA SVMs based on the probability of classes.

## Acknowledgement

This work was supported by the Korea Science and Engineering Foundation (KOSEF) through the Biometrics Engineering Research Center (BERC) at Yonsei University. The authors also would like to thank Prof. Anil Jain and Dr. Salil Prabhakar for providing FingerCode data.

## References

- [1] A. Senior, "A combination fingerprint classifier," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 23, no. 10, pp. 1165-1174, 2001.
- [2] K. Karu and A. Jain, "Fingerprint classification," *Pattern Recognition*, vol. 29, no. 3, pp. 389-404, 1996.
- [3] Q. Zhang and H. Yan, "Fingerprint classification based on extraction and analysis of singularities and pseudo ridges," *Pattern Recognition*, vol. 37, no. 11, pp. 2233-2243, 2004.
- [4] A. Jain, S. Prabhakar and L. Hong, "A multichannel approach to fingerprint classification," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 21, no. 4, pp. 348-359, 1999.
- [5] C. Park and H. Park, "Fingerprint classification using fast Fourier transform and nonlinear discriminant analysis," *Pattern Recognition*, vol. 38, no. 4, pp. 495-503, 2005.
- [6] J.-K. Min, J.-H. Hong and S.-B. Cho, "Effective fingerprint classification by localized models of support vector machines," *Lecture Notes in Computer Science*, vol. 3832, pp. 287-293, 2006.
- [7] Y. Yao, G. Marcialis, M. Pontil, P. Frasconi and F. Roli, "Combining flat and structured representations for fingerprint classification with recursive neural networks and support vector machines," *Pattern Recognition*, vol. 36, no. 2, pp. 397-406, 2003.
- [8] N. Yager and A. Amin, "Fingerprint classification: A review," *Pattern Analysis and Applications*, vol. 7, no. 1, pp. 77-93, 2004.
- [9] C. Watson and C. Wilson, Fingerprint Database, National Institute of Standard and Technology, Special Database 4, *FPDB*, April 1992.

# **Design on Supervised / Unsupervised Learning Reconfigurable Digital Neural Network Structure**

In Gab Yu<sup>1</sup>, Yong Min Lee<sup>1</sup>, Seong Won Yeo<sup>2</sup>, and Chong Ho Lee<sup>1</sup>

<sup>1</sup> Department of Information Technology & Telecommunication,  
Inha University, Incheon, Korea

{yuingab, ymlee}@inhaian.net

<sup>2</sup> Siemens Automotive Systems co., Korea  
seongwon.yeo@siemens.com

**Abstract.** We propose a reconfigurable neural network structure which has capability to process supervised or unsupervised learning algorithm computation. The proposed structure is based on modular structure which can configure artificial neural network architecture flexibly. Main processing unit of the proposed structure is designed to obtain flexibility of its internal structure by specific instructions. Therefore it is possible to configure MLP (Multi-Layer Perceptron) with back-propagation for alphabet recognition and parallel SOM for impulse noise detection problem. The performance comparison with the matlab simulation shows its value in the aspects of reliability.

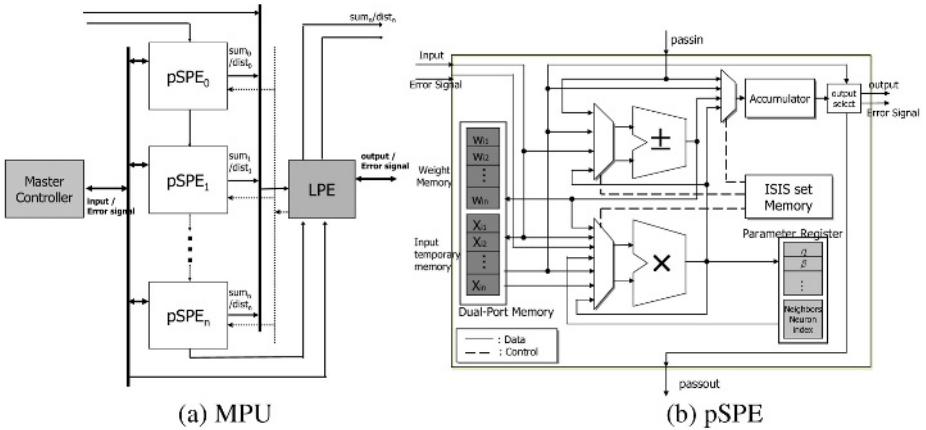
## **1 Introduction**

ANN (Artificial Neural Network) has various sizes, types and learning algorithms depending on its application fields and cases. Therefore if ANN is implemented by hardware, it is essential that ANN hardware structure has flexibility. Recently, ANN hardware has been implemented on FPGA which has reconfigurability. But FPGA has limited logic density, ANN on FPGA have usually performed the off-chip learning phase. To overcome this problem, several methods are proposed to operate learning algorithm on FPGA. First, there is simple logic circuit without multiplier. the modified BP (Back-Propagation algorithm) which has no multiplication is implemented with pulse mode operation [1]. But, the disadvantage of this method is the lack of precision. Second, there is the method which increases effective functionality of FPGA by time-multiplexing a single FPGA through run-time reconfiguration [2]. Using this method, BP is divided into sequential step on the same FPGA resource. However, the overhead of this method depends on the reconfiguration time compare to computation time.

We propose the ANN hardware structure of which PE (Processing Element) can process supervised or unsupervised learning algorithm computation efficiently. This structure, based on ERNIE (Expansible & Reconfigurable Neuro Informatics Engine) [3], does not need reconfiguration time for learning process and can configure ANN architecture intuitively.

## 2 Reconfigurable Neural Networks Structure

MPU is a modular structure of the smallest size which is essential to configure neural network architecture. MPU consists of two types of PE, pSPE and LPE as shown in Fig. 1. (a).



**Fig. 1.** Block diagram of MPU and pSPE

pSPE's major functions are calculation of summing node in MLP, weight update in BP learning and computation of error signal for hidden layers. It is also used in calculating Euclidean distance and weight update in SOM.

In pSPE, the outputs of each arithmetic logic block are connected to the input of bus multiplexer as shown in Fig. 1. (b). pSPE can control internal data path in accordance with bus multiplexer's select signal. ISIS (Instruction for Selecting Internal Structure) is an instruction for integrating commands of select signal in bus multiplexer and control signal in memory and arithmetic logic block. pSPE can operate designated computation by specific ISIS set.

LPE has three functions. First, there is the activation function implemented by LUT (Look-up Table). Second, there is a selection of winner neuron in SOM. Third, there is the auxiliary computation of error signal which is needed in BP learning. Like an activation function, its derivative is separately implemented by LUT.

Master controller generates configuration data which are needed to configure neural network architecture to make the proposed structure possible to operate in on-chip stand-alone. It is used in calculation error signal for output layer in BP learning. In SOM learning, it also inputs the index of winner neuron and input pattern to the proposed structure.

### 2.1 Implementation

The proposed structure is designed with Verilog HDL. Xilinx ISE 7.1i and XST (Xilinx Synthesis Tool) is used to design circuits, synthesize respectively. Virtex-II (XC2V6000-4ff1152) is chosen as target device. Synthesized circuit of the proposed

**Table 1.** Design summary of the proposed structure

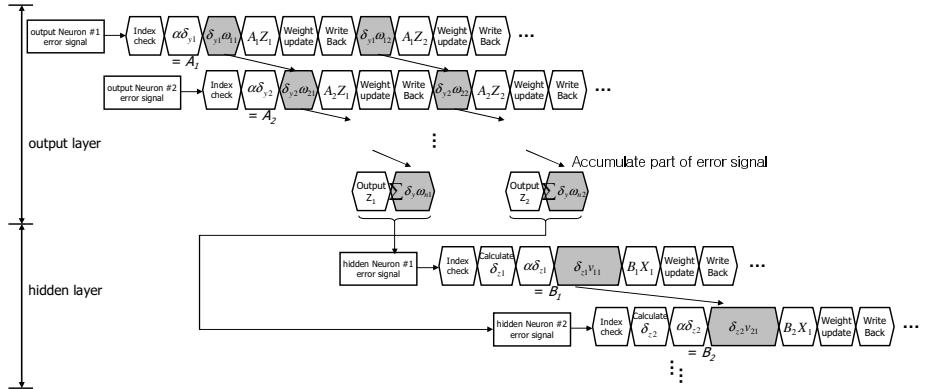
Specification	Description
Number of slices	28073 out of 33792 (83%)
Number of MULT18×18s	64 out of 144
Number of BRAMs	132 out of 144
Maximum Frequency	49.3 MHz

structure is verified by Modelsim 5.8C. Table 1. shows the implementation result of hardware which consists of two MPUs (32 pSPEs and 1 LPE per MPU).

### 3 Implementation Learning Algorithm on the Proposed Design

#### 3.1 Implementation BP Algorithm on the Proposed Structure

In the proposed structure, BP is operated as each module takes particular tasks as shown in Fig. 2. pSPE processes weight update and error signal calculation for hidden layer by specific ISIS set. Auxiliary calculation of error signal for hidden layer is done in LPE while master controller calculate error signal of output layer.

**Fig. 2.** Timing diagram of BP procedure in the proposed structure

(1) represents the computation time from beginning input an input pattern to finish learning in BP.

$$t_{bp} = \sum_{k=1}^l N_{h,k} C_o + 2N_o + N_i C_h \quad (1)$$

where  $l$  is the number of hidden layers.  $N_{h,k}$  is the number of neurons in hidden layer.  $N_i$  is the number of input neurons.  $N_o$  is the number of neurons in output layer.  $C_o$  and  $C_h$  are computation time of output and hidden neurons respectively.

The performance comparison of BP operated by the proposed structure is 1219 MCUPS when total pSPE of the proposed structure are 256PEs and is 1,996 MCUPS

when total pSPE of the proposed structure are 512PEs. It is better performance than CNAPS (512PEs, 1,460MCUPS) [4] and Sandy (256PEs, 567MCUPS) [5].

### 3.2 Implementation Parallel SOM Learning on the Proposed Design

By using pSPE's neighborhood index register, the proposed structure is possible to operate parallel SOM learning. Each pSPE has a register which store indices of pSPE its neighborhood neuron. Each pSPE is regarded as winner neuron or its neighborhood neuron whether the index of pSPE is same as the index of inputted winner neuron or the index of stored neighborhood neurons. All of the pSPEs can figure out whether they are winner or neighborhood neuron by only the index of the winner neuron. The winner neuron and its neighborhood are learned by appropriate learning procedure using specific ISIS set.

(2) represents the clock cycle from beginning input an input pattern to finish learning in parallel SOM learning.

$$t_{SOM} = rN_i C_o Q \quad (2)$$

where  $r$  is a radius of neighborhood,  $Q$  is the number of radius decrease times and  $N_i$  is the number of input neurons.  $C_o$  is the weight update time of a pSPE in output layer. The performance of parallel SOM (8×8 size map) learning operated by the proposed structure is 263MCUPS. It is better performance than NBISOM\_25 (5×5 size, 149MCUPS) [6] and MANTRA-I (6×10 size, 13.9MCUPS) [7].

## 4 Experiments

### 4.1 Alphabet Recognition Experiment

For verifying the precision of the proposed structure, we made MLP (256-32-5) using the proposed structure and matlab modeling respectively, and carried out experiment on alphabet recognition. Learning rate is set from 0.05 to 0.2. If learning times is over 20000 times, learning processing is stopped. We compared final output and weight error after the experiment. The average error of final output (AEFO) is 0.0341, which is too small, but the average error of between weights (AEW) is 0.1347, which is larger. The reasons are that the number of weight is larger than the number of outputs, and also accumulated error caused learning repetition. Even though AEW is large, there is not much effective because AEFO is sufficiently small.

### 4.2 Impulse Noise Detection Experiment

For verifying its precision, we made SOM which has 5×5 size map on the proposed structure and matlab modeling respectively, and carried out experiment on classification normal and noise pixel in the target image under the environment of [8]. A 2-demensional area surrounded central pixel which is interested is set. 4 lines included central pixel are each vertical, horizontal, and both diagonal line. SOM can recognize normal and noise pixel by a difference between central pixel's value and mean value of each line [8]. Target image is 8bit grayscale Lena image with 10% random impulse noise, 254×254 size except boundary area. Training patterns are obtained 64513

patterns from target image. As comparing with original image, 58075 normal patterns and 6441 noise patterns are detected, and distribution of winner neuron for normal pattern and noise pattern were different. We compared matlab simulation result with the result from proposed structure with test patterns (Lena image with 15% impulse noise). There are 26959 times significantly different result. Therefore total error rate is 1.74%. Although total error rate occurred 1.74%, there is no significant effect on the SOM result because in most cases neighborhood neurons would be a winner neuron.

## 5 Conclusion

We propose FPGA based ANN hardware structure which can process supervised or unsupervised learning algorithm computation efficiently. By using ISIS, we can control internal data path in pSPE for desired computation. This contributes two advantages. First, there is an efficient processing of learning algorithm computation. Second, the proposed structure is capable of solving problem when a change of ANN structure in hardware is necessary. There is no need to control from outside when the ANN structure is being reconfigured and also to prevent performance-lowering of the hardware.

## Acknowledgement

This work is supported by Ministry of Commerce, Industry and Energy of Korea under project NO.33553-02 and by Inha University Research Grant.

## References

1. Hikawa, H.: Implementation of Simplified Multilayer Neural Networks with On-chip Learning. Proceedings of IEEE International Conference on Neural Networks, Vol. 4, (1995) 1633-1637
2. Elredge, J. G et al.: RRANN: A hardware Implementation of the Backpropagation Algorithm Using Reconfigurable FPGAs. IEEE World Conference on Computational Intelligence, Vol. 4, (1994) 2097-2102
3. Kim, Y. J. et al.: Expansible & Reconfigurable Neuro Informatics Engine : ERNIE. Journal of the Institute of Electronics Engineers of Korea CI, Vol. 40, NO.6, (2003) 56-68.
4. Heemskerk, J. N. H.: Overview of Neural Hardware: Neurocomputers for Brain-Style Processing. Design, Implementation and Application. Ph.D Thesis, Unit of Experimental and Theoretical Psychology Leiden University, the Netherlands (1995)
5. Torresen, T.: Parallelization of Backpropagation Training for Feed-Forward Neural Networks. Ph.D thesis, The Norwegian institute of Technology The University of Trondheim, (1996) 58
6. Ruping, S et al.: SOM accelerator system. Neurocomputing, Vol. 21, (1998) 31-50
7. Ienne, P. et al.: Modified Self-Organizing Feature Map Algorithms for Efficient Digital Hardware Implementation. IEEE Transactions on Neural Networks, Vol.8, NO. 2, (1997) 315-330
8. Song, S. M. et al.: Impulse Noise Detection Using Self-Organizing Neural Network and Its Application to Selective Median Filtering. Transaction KIEE, Vol. 54, NO. 3 (2005)

# Car Plate Localization Using Pulse Coupled Neural Network in Complicated Environment

Ming Guo<sup>1</sup>, Lei Wang<sup>2</sup>, and Xin Yuan<sup>2</sup>

<sup>1</sup> School of Computing, City College, Zhejiang University,  
Hangzhou, P.R. China, 310015

<sup>2</sup> College of Computer Science, Zhejiang University, Hangzhou, P.R. China, 310027  
[yxxinyuan@zju.edu.cn](mailto:yxxinyuan@zju.edu.cn)

**Abstract.** Car plate recognition is an important problem in many traffic related applications. In this paper, we focus on car plate localization—the first step of car plate recognition. We propose a hybrid method based on Pulse Coupled Neural Network (PCNN) and wavelet analysis. First of all, we use PCNN to enhance the image. Then, regions of interest (ROIs) will be get through wavelet analysis. After that, PCNN enhancement is applied again in ROIs, followed by a training and classification process for final labelling ROIs as car plate regions or not. Experiment results show that the precision can get 96%, which is higher than other localization methods on the same image database.

## 1 Introduction

Car plate recognition is the key in many traffic related applications such as traffic control, Intelligent Transport System (ITS) and general vehicle security systems. Generally, it consists of three procedures: localization of car plate regions, segmentation of characters from the plate regions and recognition of each character [1].

The key of car plate recognition is the first procedure, car plate localization, which is not easy especially in outdoor scenes. A large number of efforts have been made to localize car plate regions from images. They include some classical methods such as gradient information [2], morphological operators [3], and horizontal and vertical projections [4], and some new methods such as Yang's algorithm, which utilize the color collocation of the plate's background and characters combined with the plate's structure and texture to localize the vehicle license plate and reached the recognition rate 95% [5].

Pulse Coupled Neural Network (PCNN) is a special kind of artificial neuron network model, which emulates the behavior of cortical neurons observed in visual cortices of cats [6]. Because PCNN can capture relative complete objects even if the objects do not look so clear, it is better than other methods in car plate localization. Chacon applied PCNN to the whole image to generate candidate regions and can obtain 85% precision [7]. To improve correct localization rate, we develop a hybrid method of PCNN and wavelet analysis for localizing car plate.

## 2 Methodology

### 2.1 PCNN Enhancement

In this step, we use PCNN enhancement to eliminate some unimportant details in the image so that we can get relatively exact ROIs in the following step.

The general model of PCNN consists of five blocks, feeding, linking, dynamic threshold, internal activity, and pulse generator [8]. Fig. 1 shows the neuron model. When neuron's internal activity  $U$  (calculated by  $F$  and  $L$ ) is higher than its threshold  $\theta$ , the neuron *fires* and emits a pulse. The threshold  $\theta$  is charged to  $V_\theta$  immediately, which is much higher than internal activity  $U$ , and then decay exponentially according to the time constant  $\tau_\theta$ . When the threshold  $\theta$  is lower than the internal activity  $U$ , the neuron fires again.

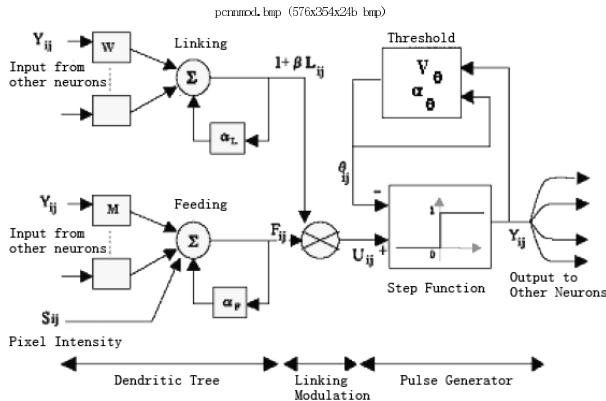


Fig. 1. pulse coupled neuron model

The feeding region of the neural element can be defined as follows,

$$F(t) = G_{feed} e^{-\alpha F^{\Delta t}} F(t-1) + S + Y(t-1) * W , \quad (1)$$

where  $G_{feed}$  is the feed gain,  $S$  is the input image,  $\alpha_F$  is the time constant of the leakage filter of the feeding region,  $Y(t)$  is the neuron output at time  $t$ , and  $W$  is the feeding kernel. The linking activity can be described as

$$L(t) = G_{link} e^{-\alpha L^{\Delta t}} L(t-1) + Y(t-1) * M , \quad (2)$$

where  $G_{link}$  is the link gain,  $\alpha_L$  is the time constant of the leakage filter of the linking region, and  $M$  is the linking kernel. The internal activity of the neuron element depends on the linking and feeding activity denoted as

$$U(t) = F(t)[1 + \beta L(t)] , \quad (3)$$

where  $\beta$  is the linking coefficient defined the amount of modulation of the linking activity. The dynamic threshold is implemented by

$$\theta(t) = e^{-\frac{1}{\alpha_\theta}} \theta(t-1) + VY(t) , \quad (4)$$

where  $\alpha_\theta$  is the time constant of the leakage filter of the threshold and  $V$  is the threshold gain. The final output of the neuron is defined by

$$Y(t) = \begin{cases} 1 & \text{if } U(t) > \theta(t) \\ 0 & \text{otherwise} \end{cases} . \quad (5)$$

In our algorithm, The pulse coupled neural network is a single layer two-dimensional array. Each pixel in the image is associated with a unique neuron and vice versa. The feeding input ( $F$ ) is the intensity of its corresponding pixel. Each neuron receives a linking input ( $L$ ) from its four neighbors. We record the firing time of each neuron and form a firing time sequence image with  $N$  levels ( $N = 50$  in our algorithm). Different levels represent different objects. Object details have been eliminated so that ROIs extraction, which will be carried out in next step, is easier.

## 2.2 ROI Extraction

Because the car plate regions have the obvious features, so we consider the structural characteristics of regions are different from other regions. So wavelet analysis is suitable for ROI extraction.

We split the entire image to a set of  $m \times n$  blocks in which  $m$  and  $n$  depends on the size of image and the size of car plate region. Then wavelet decomposition is applied in each block and generate four subbands (smoothing, horizontal, vertical, and diagonal), denoted by  $W_{LL}$ ,  $W_{HL}$ ,  $W_{LH}$ , and  $W_{HH}$ , which represent responses of different orientations. Now we extract subband's energy and standard deviation features from wavelet coefficients and compose an eight-dimensional vector for each block.

A k-means clustering can be used in those feature vectors of the image blocks. We cluster them to two classes, one contains backgrounds and the other ROIs. The class which has relatively high energy is considered containing ROIs.

## 2.3 PCNN Enhancement on ROIs

To improve precision of classification in the next step, we need threshold the ROIs to a binary image. The normal thresholding methods can not obtain satisfied results because of variant illuminations. So we use PCNN again on ROIs, which has the same model mentioned above with different parameters. The PCNN fires only two times to form a firing time sequence image with two levels — one contains characters and the other contains background.

## 2.4 Recognition of Car Plate Regions

In this step, we extract several features from PCNN processed ROIs in the car plate image for further labelling them as license plate region or not by pattern

classification method. We use support vector machine (SVM) method [9] to implement training and classification process because SVM can train samples with small numbers and achieve a relatively better effectiveness of classification. After all above processes, we obtain the final segmented image which contains license plate.

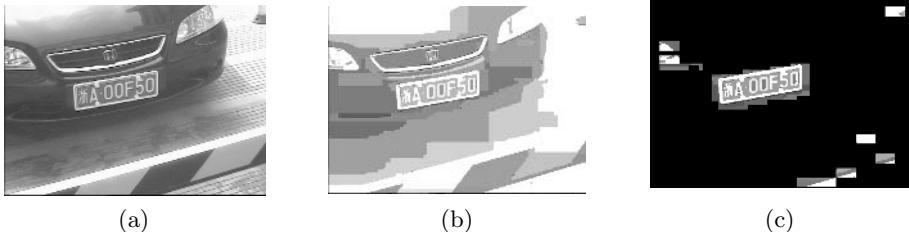
### 3 Experimental Results

Car plate localization algorithm mentioned above was tested with an image database of 250 images acquired with a camera installed in a entrance of parking. All images are obtained in outdoor scenes. We selected 50 images randomly as training set, and others as test set.

After enhancing by PCNN, the car plate image was split into  $60 \times 60$  blocks on which wavelet features were extracted in order to mark ROIs. Then PCNN method was applied again on ROIs two form a binary image. The parameters of two PCNN is listed in Tab. 1. One of the experiment results is illustrated in Fig. 2. It is easy to see that the effect is satisfied. It is in favor of the further process such as segment single character and its recognition.

**Table 1.** Parameters for PCNN

Parameters	$\beta$	$G_{feed}$	$\alpha_F$	$G_{link}$	$\alpha_L$	$\alpha_\theta$	$V$
First	0.2	0.7	-0.1	0.5	-0.1	0.75	10
Second	1.0	0.7	-0.1	0.6	-0.1	0.75	10



**Fig. 2.** One of the experiment results. (a): Original image. (b): After PCNN enhancement. (c): ROIs of the image.

Tab. 2 shows the comparison results of several methods, which contain our algorithm and others mentioned in previous works, on the same image database. Obviously, the precision of our algorithm, which is above 96%, is much higher than others'. The total process of car plate localization needs 3 seconds when running in P4 machine with 2.0G CPU speed and windows platform.

**Table 2.** Comparison of several license plate localization methods on the same image database

Methods	Precision
Our algorithm	96.2%
Gradient information[2]	81.5%
Morphological operator[3]	82.3%
Projection[4]	87.8%
Chacon's PCNN[7]	84.4%

## References

1. Takashi, N., Toshihiko, T., Keichi, Y., et al.: Robust license-plate recognition method for passing vehicles under outside environment. *IEEE Trans. on Vehicular Technology* **49**(6) (2000) 2309–2319
2. Kim, S., Kim, D., Ryu, Y., Kim, G.: A robust license-plate extraction method under complex image conditions. In: Proceedings of 16th International Conference on Pattern Recognition. (2002) 216–219
3. Hsieh, J., Yu, S., Chen, Y.: Morphology-based license plate detection from complex scenes. In: Proceedings of 16th International Conference on Pattern Recognition. Volume 3. (2002) 176–179
4. Wei, W., Huang, X., Wang, M.: An automatic system of vehicle number-plate recognition based on neural networks. *Journal of Systems Engineering and Electronics* **12**(2) (2001) 63–72
5. Yang, Y., Bai, J., Tian, R., Liu, N.: A vehicle license plate recognition system based on fixed color collocation. In: Proceedings of 2005 International Conference on Machine Learning and Cybernetics. Volume 9. (2005) 5394–5397
6. Eckhorn, R., Reitboeck, H., Arndt, M., Dicke, P.: Feature linking via synchronization among distributed assemblies:simulations of results from cat visual cortex. *Neural Comput.* **2** (1990) 293–307
7. Chacon, M., Zimmerman, A.: License plate location based on a dynamic pcnn scheme. In: Proceedings of the International Joint Conference on Neural Networks. Volume 2. (2003) 1195–1200
8. Johnson, J., Padgett, M.: Pcn models and applications. *IEEE Transactions on Neural Networks* **10**(3) (1999) 480–498
9. Vapnik, V.: *The Nature of Statistical Learning Theory*. Springer (2000)

# A Split-Step PSO Algorithm in Predicting Construction Litigation Outcome

Kwok-wing Chau

Department of Civil and Structural Engineering, Hong Kong Polytechnic University,  
Hunghom, Kowloon, Hong Kong  
cekwchau@polyu.edu.hk

**Abstract.** Owing to the highly complicated nature and the escalating cost involved in construction claims, it is highly desirable for the parties to a dispute to know with some certainty how the case would be resolved if it were taken to court. The use of artificial neural networks can be a cost-effective technique to help to predict the outcome of construction claims, on the basis of characteristics of cases and the corresponding past court decisions. This paper presents the application of a split-step particle swarm optimization (PSO) model for training perceptrons to predict the outcome of construction claims in Hong Kong. The advantages of global search capability of PSO algorithm in the first step and local fast convergence of Levenberg-Marquardt algorithm in the second step are combined together. The results demonstrate that, when compared with the benchmark backward propagation algorithm and the conventional PSO algorithm, it attains a higher accuracy in a much shorter time.

## 1 Introduction

It is highly desirable for the parties to a dispute to know with some certainty how the case would be resolved if it were taken to court. This would effectively help to significantly reduce the number of disputes that would need to be settled by the much more expensive litigation process. Artificial neural networks (ANN), in particular the feed forward back-propagation (BP) perceptrons, have been widely applied in different fields [1-2]. ANN can be a cost-effective technique to identify the hidden relationships among various interrelated factors and to mimic decisions that were made by the court. However, slow training convergence speed and easy entrapment in a local minimum are inherent drawbacks of the commonly used BP algorithm [3]. Levenberg-Marquardt (LM) optimization technique [4] is a commonly used ANN that has attained certain improvements such as convergence rates over the BP algorithm. Swarm intelligence is another recent SC technique that is developing quickly [5]. These SC techniques have been applied in construction problem and accomplished satisfactory results [6-7].

In this paper, a split-step PSO algorithm is employed to train multi-layer perceptrons for prediction of the outcome of construction litigation in Hong Kong. It is believed that, by combining the two algorithms, the advantages of global search

capability of PSO algorithm in the first step and local fast convergence of LM algorithm in the second step can be fully utilized to furnish promising results.

## 2 Nature of Construction Disputes

Prior to the actual construction process, the involving parties will attempt to sort out the conditions for claims and disputes through the contract documents. However, since a project usually involves thousands of separate pieces of work items to be integrated together to constitute a complete functioning structure, the potential for honest misunderstanding is extremely high. The legislation now in force requires that any disputes incurred have to be resolve successively by mediation, arbitration, and the courts [8].

## 3 Characteristics of PSO Algorithm

Among other advantages, the more significant one is its relatively simple coding and hence low computational cost. One of the similarities between PSO and a genetic algorithm is the fitness concept and the random population initialization. However, the evolution of generations of a population of these individuals in such a system is by cooperation and competition among the individuals themselves. The capability of stochastic PSO algorithm, in determining the global optimum with high probability and fast convergence rate, has been demonstrated in other cases [9-10]. PSO can be readily adopted to train the multi-layer perceptrons as an optimization technique.

## 4 Training of Perceptrons by PSO

Without loss of generality, a three-layered preceptron is considered in the following.  $W^{[1]}$  and  $W^{[2]}$  represent the connection weight matrix between the input layer and the hidden layer, and that between the hidden layer and the output layer, respectively. During training of the preceptron, the i-th particle is denoted by  $W_i = \{W^{[1]}, W^{[2]}\}$  whilst the velocity of particle i is denoted by  $V_i$ . The position representing the previous best fitness value of any particle is denoted by  $P_i$  whilst the best matrix among all the particles in the population is recorded as  $P_b$ . Let m and n represent the index of matrix row and column, respectively, the following equation represents the computation of the new velocity of the particle based on its previous velocity and the distances of its current position from the best experiences both in its own and as a group.

$$\begin{aligned} V_i^{[j]}(m, n) = & V_i^{[j]}(m, n) + r\alpha[P_i^{[j]}(m, n) - W_i^{[j]}(m, n)] \\ & + s\beta[P_b^{[j]}(m, n) - W_i^{[j]}(m, n)] \end{aligned} \quad (1)$$

where  $j = 1, 2$ ;  $m = 1, \dots, M_j$ ;  $n = 1, \dots, N_j$ ;  $M_j$  and  $N_j$  are the row and column sizes of the matrices W, P, and V;  $r$  and  $s$  are positive constants;  $\alpha$  and  $\beta$  are random numbers in the range from 0 to 1. In the context of social behavior, the cognition part

$r\alpha[P_i^{[j]}(m,n) - W_i^{[j]}(m,n)]$  denotes the private thinking of the particle itself whilst the social part  $s\beta[P_b^{[j]}(m,n) - W_i^{[j]}(m,n)]$  represents the collaboration among the particles as a group. The new position is then determined based on the new velocity as follows:

$$W_i^{[j]} = W_i^{[j]} + V_i^{[j]} \quad (2)$$

The fitness of the i-th particle is determined in term of an output mean squared error of the neural networks as follows:

$$f(W_i) = \frac{1}{S} \sum_{k=1}^S \left[ \sum_{l=1}^O \{t_{kl} - p_{kl}(W_i)\}^2 \right] \quad (3)$$

where  $f$  is the fitness value,  $t_{kl}$  is the target output;  $p_{kl}$  is the predicted output based on  $W_i$ ;  $S$  is the number of training set samples; and,  $O$  is the number of output neurons.

## 5 The Study

The system is applied to study and predict the outcome of construction claims in Hong Kong. The data from 1991 to 2000 are organized case by case and the dispute characteristics and court decisions are correlated. Through a sensitivity analysis, 13 case elements that seem relevant in courts' decisions are identified. They are, namely, type of contract, contract value, parties involved, type of plaintiff, type of defendant, resolution technique involved, legal interpretation of contract documents, misrepresentation of site, radical changes in scope, directed changes, constructive changes, liquidated damages involved, and late payment.

Some of the 13 case elements can be expressed in binary format; for example, the input element 'liquidated damages involved' receives a 1 if the claim involves liquidated damages or a 0 if it does not. However, some elements are defined by several alternatives; for example, 'type of contract' could be remeasurement contract, lump sum contract, or design and build contract. These elements with alternative answers are split into separate input elements, one for each alternative. Each alternative is represented in a binary format, such as 1 for remeasurement contract and 0 for the others if the type of contract is not remeasurement. In that case, only one of these input elements will have a 1 value and all the others will have a 0 value. In this way, the 13 elements are converted into an input layer of 30 neurons, all expressed in binary format. Table 1 shows examples of the input neurons for cases with different types of contract. The court decisions are also organized in an output layer of 6 neurons expressed in binary format corresponding to the 6 elements: client, contractor, engineer, sub-contractor, supplier, and other third parties.

In total, 1105 sets of construction-related cases were available, of which 550 from years 1991 to 1995 were used for training, 275 from years 1996 to 1997 were used for testing, and 280 from years 1998 to 2000 were used to validate the network results with the observations. It is ensured that the data series chosen for training and validation comprised balanced distribution of cases.

**Table 1.** Examples of the input neurons for cases with different types of contract

Input neuron	Cases		
	Remeasurement	Lump sum	Design and build
Type of contract - remeasurement	1	0	0
Type of contract - lump sum	0	1	0
Type of contract – design and build	0	0	1

Sensitivity analysis is performed to determine the best architecture, with variations in the number of hidden layers and number of hidden neurons. The final perceptron has an input layer with thirty neurons, a hidden layer with fifteen neurons, and output layer with six neurons. In the PSO-based perceptron, the number of population is set to be 40 whilst the maximum and minimum velocity values are 0.25 and -0.25 respectively.

**Table 2.** Comparison of prediction results for various perceptrons

Algorithm	Training		Validation	
	Coefficient of correlation	Prediction rate	Coefficient of correlation	Prediction rate
BP-based	0.956	0.69	0.953	0.67
PSO-based	0.987	0.81	0.984	0.80
LM	0.964	0.70	0.957	0.68
Split-step	0.990	0.84	0.987	0.83

## 6 Analysis and Discussions

The performance of the split-step multi-layer ANN is evaluated in comparison with the benchmarking standard BP-based network, a PSO-based network and a LM network. In order to provide a fair and common initial ground for comparison purpose, the training process of the BP-based perceptron or LM network commences from the best initial population of the corresponding PSO-based perceptron or split-step network. Table 2 shows comparisons of the results of network for various perceptrons. It can be observed that the split-step algorithm performs the best in terms of prediction accuracy. It is noted that testing cases of the split-step PSO-based network are able to give a successful prediction rate higher than 80%, which is much higher than by pure chance. It is believed that, if the involving parties to a construction dispute become aware with some certainty how the case would be resolved if it were taken to court, the number of disputes could be reduced significantly.

## 7 Conclusions

In this paper, a perceptron based on a split-step PSO algorithm is employed for prediction of outcomes of construction litigation on the basis of the characteristics of the individual dispute and the corresponding past court decisions. The results show that the split-step PSO-based perceptron outperforms the other commonly used optimization techniques in prediction of outcomes of construction litigation, in terms of both convergence and accuracy. The final network presented in this study is recommended as an approximate prediction tool for the parties in dispute, since the rate of prediction is higher than 80%, which is much higher than chance. It is, of course, recognized that there are limitations in the assumptions used in this study. Other factors that may have certain bearing such as cultural, psychological, social, environmental, and political factors have not been considered here.

## References

1. Chau, K.W., Cheng, C.T.: Real-Time Prediction of Water Stage with Artificial Neural Network Approach. Lecture Notes in Artificial Intelligence, **2557** (2002) 715-715
2. Cheng, C.T., Chau, K.W., Sun, Y.G., Lin, J.Y.: Long-term Prediction of Discharges in Manwan Reservoir using Artificial Neural Network Models. Lecture Notes in Computer Science **3498** (2005) 1040-1045
3. Rumelhart, D.E., Widrow, B., Lehr, M.A.: The Basic Ideas in Neural Networks. Communications of the ACM **37(3)** (1994) 87-92
4. Hagan, M.T., Menhaj, M.B.: Training Feedforward Networks with the Marquardt Algorithm. IEEE Transactions on Neural Networks **5(6)** (1994) 989-993
5. Kennedy, J., Eberhart, R.: Particle Swarm Optimization. Proceedings of the 1995 IEEE International Conference on Neural Networks. Perth (1995) 1942-1948
6. Ardit, D., Oksay, F.E., Tokdemir, O.B.: Predicting the Outcome of Construction Litigation Using Neural Networks. Computer-Aided Civil and Infrastructure Engineering **13(2)** (1998) 75-81
7. Chau, K.W.: Predicting Construction Litigation Outcome using Particle Swarm Optimization. Lecture Notes in Artificial Intelligence **3533** (2005) 571-578
8. Chau, K.W.: Resolving Construction Disputes by Mediation: Hong Kong Experience. Journal of Management in Engineering, ASCE **8(4)** (1992) 384-393
9. Kennedy, J.: The Particle Swarm: Social Adaptation of Knowledge. Proceedings of the 1997 International Conference on Evolutionary Computation. Indianapolis (1997) 303-308
10. Clerc, M., Kennedy, J.: The Particle Swarm—Explosion, Stability, and Convergence in a Multidimensional Complex Space. IEEE Transactions on Evolutionary Computation **6(1)** (2002) 58-73

# An Efficient Unsupervised MRF Image Clustering Method

Yimin Hou<sup>1</sup>, Lei Guo<sup>1</sup>, and Xiangmin Lun<sup>2</sup>

<sup>1</sup> Department of Automation, Northwestern Polytechnical University, Xi'an 710072

<sup>2</sup> Xi'an Institute of Optics and Precision Mechanics of Cas, Xi'an 710068

**Abstract.** In this paper, a robust image segmentation method is proposed. The relationship between pixel intensities and distance between pixels are introduced to the traditional neighbourhood potential function To perform an unsupervised segmentation, the Bayes Information Criterion (BIC) is used to determine the class number, the K-means is employed to initialise the classification and calculate the mean values and variances of the classes. The segmentation is transformed to maximize a posteriori(MAP) procedure. Then, the Iterative Conditional Model (ICM) is employed to solve the MAP problem. In the experiments, the proposed method is compared with other segmentation techniques, for noisy image segmentation applying on synthetic and real images. The experiment results shows that the proposed algorithm is the better choice.

## 1 Introduction

Many segmentation methods have been presented such as edge-based segmentation[1], region-based segmentation[2] and pixel labeling. In the early works of image segmentation, image pixels were classified independently. These approaches could not produce satisfactory classifications. The image spatial information was ignored in the traditional methods. To solve this problem, a statistical method, the MRF image models[3], is adopted in this paper. The relationships a the pixel intensities and distance values are introduced to the clique potential functions, which define the MRF probability distribution discussed[4]. For two pixels, the probability of being classified into a same class changes with the variation of intensity difference and distance between their positions. BIC is used to get the optimization class number and the K-means method is used to obtain the mean values and variances of every class. The algorithm proposed in this paper is absolutely an unsupervised segmentation method. The new potential is used in MAP procedure to get the segmentation result.

In Section 2, the MRF image model is described and, based on Beyes theorem, the transformation from segmentation to MAP is discussed. The iterative conditional model is used in this section to solve the MAP problem. In Section 3, new potential function, which involves the pixel intensity and distance values, is introduced. The model fitting is done. In Section 4, some experiments are illustrated to show the segmentation results.

## 2 MRF-MAP Segmentation

Let  $\mathcal{Q} = \{q(i, j) | 1 \leq i \leq W, 1 \leq j \leq H\}$  be a finite set of sites of a  $W \times H$  rectangular lattice. Let the label field  $X = \{X_1, \dots, X_m\}$  be a Markov Random Field defined on  $\mathcal{Q}$ ,  $m$  is the total number of the classes. In an MRF, the sites in  $\mathcal{Q}$  are related to each other via a neighborhood system,  $\psi = \{N_q, q \in \mathcal{Q}\}$ , where  $N_q$  is the set of neighbors of  $q(i, j)$ . A clique is a subset in  $N_q$ ,  $c \in N_q$  is a clique of distinct sites being neighbors in.

A MRF is completely described by a Gibbs distribution using the Hammersley-Clifford theorem[5][6]. The MRF has the form  $P(X = x) = \frac{1}{Z} e^{-U(X)/T}$ , where  $U(X) = \sum_Q V_c(x)$ .  $c$  is a clique in  $N_q$ . Image  $Y$  can be considered as a rectangular lattice  $\mathcal{Q}$ . Let  $y_q$  denotes the intensity of the pixel at  $q$  and it correspond to the label  $x_q$  in  $X$ .  $P(Y|X)$  and  $P(X)$  denote the conditional probability density of  $Y$  and the prior of the labeling  $X$ . It could be assumed that the image data are obtained by adding an identical independently distributed Gaussian noise[7]. The conditional density  $P(Y|X)$  takes the form of

$$P(Y|X) = \prod_Q P(y_q|x_q) = \prod_Q \left[ \frac{1}{\sqrt{2\pi\sigma_q^2}} \exp\left(-\frac{(y_q - \mu_q)^2}{2\sigma_q^2}\right) \right] \propto \exp\left(-\sum_Q \frac{(y_q - \mu_q)^2}{2\sigma_q^2}\right) \quad (1)$$

where  $\mu_q$  is the mean value and  $\sigma_q^2$  is the variance of the class that site  $q$  belongs to.

$P(X)$  takes the form of  $P(X) = \frac{1}{Z} \exp\left(-\sum_Q \sum_{c \in N_q} \frac{V_c(q(i, j), q(i_1, j_1))}{T}\right)$ , where  $q(i, j)$  and  $q(i_1, j_1)$

are sites belong to the same neighborhood,  $V_c(q(i, j), q(i_1, j_1))$  is potential function. The optimal  $X$  can be got by minimization the potential summation of conditional possibility and the prior according to MAP.

$$X_{opt} = \arg \min_{x \in \Omega} U(X|Y) = \arg \min \left\{ \sum_Q \left[ \frac{(y_q - \mu_q)^2}{2\sigma_q^2} + \frac{1}{T} \sum_{c \in N_q} V_c(q(i, j), q(i_1, j_1)) \right] \right\} \quad (2)$$

The Iterated Conditional Models(ICM) algorithm is likely to reach only local minima, but it provides a much faster convergence than stochastic relaxation-based method. For each pixel  $q$ , given the observed image and current labels of all the pixels in the neighborhood, the label of  $X_q$  is replaced with one that can maximize the probability

$$X_q^{(k+1)} = \arg \max P(X_q^{(k)} | Y, X_r^{(k)}, r \neq q) \quad (3)$$

Starting from the initial state, the algorithm will keep on running base on the procedure above until either the predefined number of iterations is reached or the label of  $X$  does not change.

### 3 New Potential Function and Model Fitting

Classical energy function can be defined as  $V_c(q(i, j), q(i_1, j_1)) = \begin{cases} -\beta & \text{if } q(i, j) = q(i_1, j_1) \\ 0 & \text{otherwise} \end{cases}$ .

Intuitively, the equation above has not moreover involved spatial information of the image. A more effective function is used in this work as proposed:

$$V'_c(q(i, j), q(i_1, j_1)) = \begin{cases} -\frac{\beta\sigma_{ij}^2}{(\sigma_q^2 + (y_{q(i, j)} - y_{q(i_1, j_1)})^2 \times d_{q(i, j)q(i_1, j_1)})} & \text{if } q(i, j) \neq q(i_1, j_1) \\ -\beta & \text{if } q(i, j) = q(i_1, j_1) \end{cases} \quad (4)$$

where  $y_{q(i, j)}$  and  $y_{q(i_1, j_1)}$  are the pixel intensities of  $q(i, j)$  and  $q(i_1, j_1)$ ;  $d_{q(i, j)q(i_1, j_1)}$  represents the distance between the two pixels;  $\beta$  is a constant that controls the classification. We can draw a conclusion from the equation that with the decreasing of  $d_{q(i, j)q(i_1, j_1)}$  and  $(y_{q(i, j)} - y_{q(i_1, j_1)})^2$ ,  $V'_c(q(i, j), q(i_1, j_1))$  decreased to  $-\beta$ , that is  $q(i, j)$  and  $q(i_1, j_1)$  are right one pixel or their intensities are same.

For an MRF model,  $\theta = \{\mu_1, \dots, \mu_m, \sigma_1, \dots, \sigma_m\}$  is one of the parameter set[9]. K-means is used to initialize the classification. After this step, we can get the initial parameter set  $\theta = \{\mu_1, \dots, \mu_m, \sigma_1, \dots, \sigma_m\}$ . Signed a maximal value  $m_{\max}$  and a minimum value  $m_{\min}$  of the total class number  $m$ , K-means was used for every possible value between  $m_{\min}$  to  $m_{\max}$ .

The following formulae described the Bayes Information Criterion.

$$BIC_{m_k} = 2 \log P(Y | \theta_{m_k}, m_k) - v_{m_k} \log(n) \quad (5)$$

where  $v_{m_k}$  is the number of independent parameters in the model.  $n$  is the pixel number,  $Y$  is the observation.

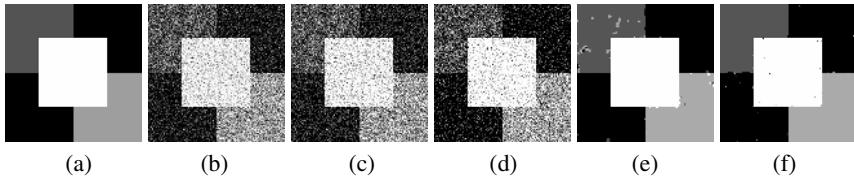
### 4 Experiments

A synthetic image shown in the Fig.1(a) was applied in the first experiment. Fig.1(b) was obtained by adding noise to (a) and the signal to noise ratio(SNR) of it is 4.55. There is great improvement over the K-means, classical EM and classical MRF whose results are shown as Fig.1 (c), (d) and (e).

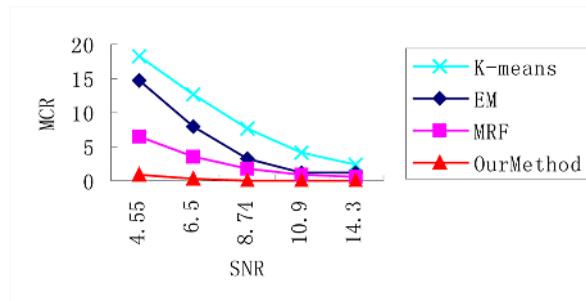
Fig.2 shows the SNR-MCR curves of segmentation of Fig.2(a) with different method. We applied the classification using the four method to five noise images with different SNR. It is obvious that the method proposed in this paper is better than the other three in either high noise or low noise conditions.

In the second example, we use the house image. Fig.3 (a) is the original image. Fig.3 (b) is the image added Gaussian noise and its SNR=8.07. We can observe an improvement when applying our algorithm. No doubt, this model is better enough to capture some finer features of the house image than classical EM and MRF methods.

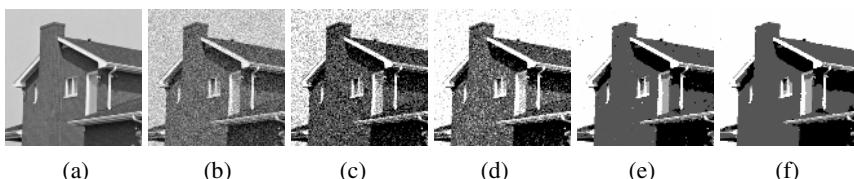
The third example, shown as Fig.4, employed a blood corpuscle image. The segmentation result of our method shows more clear main body region of the corpuscle than the results of K-means, traditional EM and traditional MRF method.



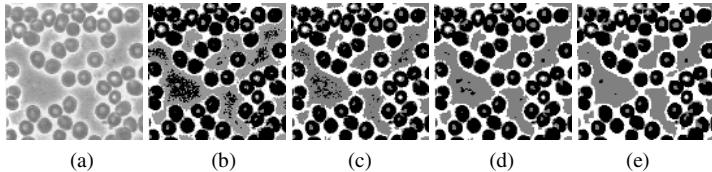
**Fig. 1.** Segmentation experiments on a synthesis image with 4 classes (a) The original image (b) Noisy synthetic image ( $SNR=4.55$ ) (c) K-means segmentation result (d) Traditional EM segmentation result (e) Traditional MRF segmentation result (f) Proposed method segmentation result



**Fig. 2.** The percentages of Miss Classified Ratio for different SNR images



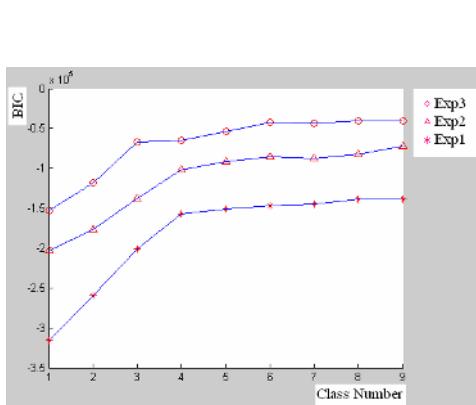
**Fig. 3.** The 4 classes segmentation experiments on house image (a) The original image (b) Noisy synthetic image ( $SNR=8.07$ ) (c) K-means segmentation result (d) Traditional EM segmentation result (e) Traditional MRF segmentation result (f) Proposed method segmentation result



**Fig. 4.** Segmentation experiments on blood corpuscle image (a) The original image (b) The K-means segmentation result (c) Traditional EM segmentation result (d) Traditional MRF segmentation result (e) Proposed method segmentation result

The Fig.5 shows the Class Number-BIC curves of the three experiments above. In the first experiment image BIC values, the point at class 4 is the last highest one before the smooth phase of the curve. So, we regard 4 as the optimal class number of the experiment image. With this criterion, we can also find the optimal class numbers, 4 and 3, of the following two experiments segmentations. Table.(1) shows the means and variances of every class in every experiment.

**Table 1.** Means and variance of every class



**Fig. 5.** Class number-BIC curves of experiments

Experiments		Means	Variances
Exp 1	Class 1	31.09	1974.15
	Class 2	87.15	330.89
	Class 3	154.02	4544.25
	Class 4	224.42	1992.05
Exp 2	Class 1	52.84	668.65
	Class 2	107.64	672
	Class 3	200.3	619.31
	Class 4	232.12	428.04
Exp 3	Class 1	136.63	195.05
	Class 2	164.57	73.19
	Class 3	251.68	14.98

## 5 Conclusions

In this paper, a novel potential function, which involved the pixel intensity and distance values, is introduced to the traditional MRF image model. Then, the segmentation problem is transformed to MAP procedure and the ICM method is employed to obtain the MAP solution. On the other side, to complete an unsupervised segmentation, the model fitting is performed using K-means and BIC. The experiment results prove that the algorithm proposed in this paper is an efficient method for images unsupervised segmentation.

## References

- [1] J. Basak, B. Chanda, D.D. Manjumder, On edge and line linking with connectionist models, *IEEE Trans. Systems, Man Cybernet.* 24 (3) 413–428, 1994.
- [2] S.A. Hojjatoleslami, J. Kittler, Region growing: a new approach, *IEEE Trans. Image Process.* 7 (7) 1079–1084, 1998.
- [3] Z. Tu and S.-C. Zho, Image Segmentation By Data-Driven Markov Chain Monte Carlo, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24,no. 5, pp. 657–673, 2002.
- [4] C.S. Won, H. Derin, Unsupervised segmentation of noisy and textured images using Markov random fields, *CVGIP: Graphical Models Image Process.* 54 (4) 308–328, 1992.
- [5] Geman, S., Geman, D, Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol.6, pp. 721–741,1984.
- [6] T.Lei, Gibbs ringing artifact spatial correlation and spatial correlation in MRI , *SPIE Proceedings*, 5368, pp.837-847,2004.
- [7] J. Besag, Towards Bayesian image analysis, *Journal of Applied Statistics*, vol.16, pp. 395–407, 1989.
- [8] H urn, M.A., Mardia, K.V. et al., Bayesian fused classification of medical images. *IEEE Trans. Med. Imag.* 15 (6), 850–858, 1996.
- [9] Gath, I., Geva, A.B., Fuzzy clustering for the estimation of the parameters of the components of mixtures of normal distributions, *Pattern Recognition Letters*, vol.9 (3), Elsevier, pp. 77-78,1989.

# Robust Gaze Estimation for Human Computer Interaction

Kang Ryoung Park

Division of Media Technology, Sangmyung University,  
7 Hongji-Dong, Jongro-ku, Seoul, Republic of Korea  
parkgr@smu.ac.kr

**Abstract.** To achieve natural human computer interface, a new gaze detection method is proposed, which allows user's natural head and eye movement with one camera system and four IR-LED illuminators. This paper has following 4 advancements compared to previous works. First, all procedures for detecting gaze position are operated automatically. Second, although we use the see-through glasses attached with eye detecting camera, the change of facial position cannot affect the gaze detection accuracy. Third, we use elliptical hough transform and geometric transform in order to detect accurate pupil region. Fourth, to solve the problem of ambiguous coin face-on of pupil shape, we use the EKF (Extended Kalman Filter) and can track continuous eye movement.

**Keywords:** Gaze Detection, Elliptical Hough Transform, Geometric Transform, EKF.

## 1 Introduction

Gaze point estimation is to find the point on a monitor screen where a user is looking at. Wang et al.[1]'s method provides the approaches that combines head pose and eye gaze estimation by a wide view camera and a panning & tilting narrow camera. However, in order to compute the gaze position, their method supposes that they know the 3D distances between two eyes, eye corners, both lip corners and the 3D diameter of eye ball. Also, it is reported that their gaze error was increased in case that user gazes at monitor center position.

More advanced method using narrow and wide view stereo cameras were shown [15]. However, in that method, user should gaze at 5 known (pre-determined) positions on a monitor in order to obtain the 3D position information of facial and eye features in calibration stage. The method [18] used one panning & tilting & zooming & focusing narrow view camera to track user's gaze position. However, to track user eye with one camera, the image resolution is not good enough and it can degrade the gaze accuracy. In addition, they use one-sided illuminators and in case of user with glasses, it cannot escape the large SR (Specular Reflection) which hides whole pupil region. The method of [19] used HMD (Head-mounted Display) and two attached cameras to track both eyes and they did not track user's head motion. The previous gaze detection system [20] shows good gaze detection performance of below 3 degrees, but it also uses one-side illuminators and cannot overcome the SR problem in case of user with glasses.

By summary, we can classify conventional gaze detection researches into two categories. One is to detect the gaze position on desktop monitor. For that, the system should track user's facial and eye position and more than two cameras with panning & tilting & zooming & focusing functionalities are required. So, system complexity is increased, consequently. Whereas, the other is to detect the gaze position on HMD monitor. For that, user wears heavy-weight HMD and only eye movement is tracked by camera. In such case, only one camera can be used, but it can give much inconvenience to user. Also, in case that HMD is slipped down after user dependent calibration, the gaze error is increased.

To overcome such problems, we propose the new method for detecting gaze position with one camera attached to see-through glasses. By using four IR-LED illuminators on desktop monitor, we can track the gaze position on a desktop monitor irrespective of user's facial movement.

## 2 Locating Eye Positions by Elliptical Hough Transform

We first locate eye features in camera image in order to detect gaze position on a desktop monitor. To detect eye features robustly, we attached a USB camera to see-through glasses. To remove the environmental lighting effect, we attached IR pass filter in front of camera lens and one IR-LED illuminator which could illuminate eye region. Due to IR-LED illuminator, the specular reflection points on pupil region can be easily detected by simple binarization. Around the detected corneal specular reflection points, we determine the eye candidate region and locate the accurate pupil center by the elliptical hough transform [8]. To compare the detection performance, we also implemented the elliptical edge detection based on circular edge detection [12] and compare the performance.

Experimental results showed that RMS error between the detected pupil center positions and the actual ones was 0.81 pixels in 640×480 pixels image by elliptical hough transform. Also, results showed that RMS error was 0.93 pixels in 640×480 pixels image by elliptical edge detection. Also, because the pupil center detection was performed only in the restricted searching area based on corneal specular reflection, it takes little time as 12ms in Pentium-IV 2 GHz PC (in case of using elliptical hough transform). And in case of using elliptical edge detection, it takes 18ms.

In addition, we can classify the error as FAR (False Acceptance Rate which accepts the non-eye region as eye) and FRR (False Rejection Rate which rejects the eye region as non-eye). Results showed that the average FAR and FRR was below 0.1%.

## 3 Tracking Continuous Gaze Position on a Desktop Monitor by EKF

In our gaze detection environment, user wears the see-through glasses attached with a USB camera and user can see the desktop monitor by it. Because the camera is attached on the bottom of glasses frame, it can detect the eye movement. But in case of user's facial movement, we should compensate the facial movement to detect gaze

position on a monitor. That is because the view which can be seen by eye is also changed according to user's facial movement.

So, we attached four IR-LED illuminators on four corners of monitor (left top, right top, left bottom and right bottom positions on a 19 inches desktop monitor). Then, four SRs(Specular Reflections) are made by four IR-LED illuminators and they mean the desktop monitor shape. So, we detect the pupil center and four SR positions in input image. Based on them, we can obtain the gaze position on a desktop monitor by the relation among the pupil center and four SR positions.

However, in case that the camera is attached to glasses frame on the skew, four SRs cannot take the rectangular shape. So, we apply a geometric transform to make non-rectangular shape into rectangular one. From the transformed rectangular shape and pupil center position, we can estimate user's gaze position on a desktop monitor.

In case of using one camera, since the proposed approach would be analogous to seeing a coin, more or less face-on, and by the shape of its silhouette working out what its orientation is. If one starts with the coin face on and rotates it by a small angle, the circular silhouette will become an oval - with a particular length and width. Rotate it by one degree and the oval will have a length : width ratio of 1:0.99985 [1:cos(1 degree)], or one part in six thousand. So, the starting with the coin-face on is the worst possible condition, but the accuracy needed is still substantial.

To solve the problem of ambiguous coin face-on, we use the EKF (Extended Kalman Filter) and we can track continuous 3D motion of eye [2]. EKF uses the constant acceleration model and estimates the successive translation and rotation angle based on 2D measurement. In detail, based on previous estimated gaze vector and EKF, we estimate the current gaze vector and solve the problem of ambiguous coin face-on. The EKF converts the measurements of 2D feature positions detected while a user moves his eye into 3D estimates of the translation and orientation of the eye using constant acceleration model. For that, we define a state vector( $\hat{x}(t)$ ) composed of 3D translation, rotation, velocity of translation and rotation, acceleration of translation and rotation. With these, the state and observation equation are obtained and the 3D motions are obtained by Eq. (1) [2]. Here,  $y(t)$  is the measurement vector and  $K(t)$  is Kalman gain.

$$\begin{aligned} \hat{x}(t) &= \hat{x}(t)^- + K(t)(y(t) - h(\hat{x}(t)^-)) \\ P(t) &= (\Phi(\Delta t)P(t-1)\Phi(\Delta t)^T + U) \times \\ &\quad \frac{\partial h}{\partial x(t)}|_{\hat{x}(t)^-} (I - K(t)) \end{aligned} \tag{1}$$

## 4 Experimental Results

The gaze detection error of our method is compared to those of previous methods [6][7][15]. The test data were acquired when 120 users gaze at 23 gaze positions on a 19" monitor. Here, the gaze error is the RMS error between the reference gaze positions and the computed ones. At the 1st experiment, the gaze errors are calculated in two cases as shown. The case I shows the gaze error about test data including only

head movements and the case II does the gaze error about test data including head and eye movements.

Experimental results showed that gaze errors of the method [6] in case I were 5.1cm (by linear interpolation), 4.23cm (by single neural net) and 4.48cm (by combined neural net). Those in case II were 11.8cm (by linear interpolation), 11.32cm (by single neural net) and 8.87cm (by combined neural net), respectively [6]. The gaze error of the method [7] in case I was 3.4cm and that in case II was 4.8cm. The gaze error of the method [15] in case I was 2.24cm and that in case II was 2.89cm. In the meanwhile, the gaze errors of proposed method in case I were 0.48cm and that in case II was 0.51cm. From that, the gaze error of the proposed method is the smallest in any case. In the previous work [17], the gaze accuracy (3.07 and 5.36 cm error in case I and II, respectively) was inferior to our proposed method. That is because in that research, two cameras such as a wide and a narrow view cameras were used and the eye gaze position was calculated only by simple 2D neural network. To overcome such problem, the research [4] used three camera systems, but its gaze detection accuracy (0.64cm in case I and 0.67cm in case II) was inferior to our method. In the previous work [21], they used the cross-ratio method to calculate gaze position on a monitor, Experimental results the RMS error of [21] was 0.55(minimum) ~ 2.4(maximum) cm (on X axis) and 0.5(minimum) ~ 2.34(maximum) cm (on Y axis), respectively. The minimum error of [21] is similar to ours, but their variance of gaze error is great. Consequently, the error was greater than our method of using geometric transformation instead of cross-ratio. At the 2nd experiment, the points of radius 5 pixels are spaced vertically and horizontally at 1.5" intervals on a 19" monitor with monitor resolution of 1280×1024 pixels as such Rikert's research. The difference between the 1st and 2nd experiment is the number of gaze points. At the 1st test, we used 23 gaze positions and at the 2nd test, we used 76 points on a monitor in order to make similar testing condition to Rikert's method. The RMS error between the real and calculated gaze position is 0.61 cm (0.44cm in X axis, 0.42cm in Y axis) and it is superior to Rikert's method (almost 5.08 cm and the error variance is 1.24 cm). At the third test, we measure the gaze accuracy according to the velocity of user's movement. The velocity of user's movement was measured by Polhemus position tracker sensor [11]. Results showed that the RMS error was 0.52cm at slow movement (translation of below 2cm and 2 degrees), 0.58cm at medium movement (translation of below 10cm and 10 degrees) and 0.55cm at high movement (translation of over 10cm and 10 degrees). From that, we can know the performance of our method is not affected by the movement of user's face.

## 5 Conclusion

This paper describes a new method for detecting gaze position on a desktop monitor with one camera attached to see-through glasses. By using four IR-LED illuminators on desktop monitor, we can track the gaze position on a desktop monitor irrespective of user's facial movement. In future works, we plan to reduce the number of IR-LED illuminators on monitor and estimate the monitor shape by using less than four IR-LED illuminators. Also, it is required to have more field test in various kind of environment.

## References

- [1] J. Wang et al., "Study on Eye Gaze Estimation", IEEE Trans. on SMC, Vol. 32, No. 3, pp. 332~350, 2002
- [2] A. Azarbayejani., "Visually Controlled Graphics". IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 15, No. 6, pp. 602~605, 1993
- [3] A. Gullstrand: The optical system of the eye. Appendices to part 1. In. Von Helmholtz H. Physiological Optics. 3<sup>rd</sup> edition.
- [4] Kang Ryoung Park, "A Study on Human Gaze Detection based on 3D Eye Model", AMDO, Andratx, Mallorca, Spain. 11-14, July, 2006
- [5] K. OHMURA et al., "Pointing Operation Using Detection of Face Direction from a Single View", IEICE Trans. Information and Systems, Vol. J72-D-II, No. 9, pp. 1441~1447, 1989
- [6] K. R. Park, Jeong Jun Lee, Jaihie Kim, "Gaze Position Detection by Computing the 3 Dimensional Facial Positions and Motions", Pattern Recognition, Vol. 35, No. 11, pp. 2559~2569, 2002
- [7] K. R. Park, "Facial and Eye Gaze detection", Lecture Notes in Computer Science, Vol. 2525, pp. 368~376, 2002
- [8] R. C. Gonzalez et al., Digital Image Processing, Addison-Wesley, 1995
- [9] Steven C. Chapra et al., Numerical Methods for Engineers, McGraw-Hill, 1989
- [10] Sheng-Wen Shih and J.Liu, "A Novel Approach to 3-D Gaze Tracking Using Stereo Cameras", IEEE Transactions on SMC, Vol. 34, No. 1, Feb. 2004
- [11] <http://www.polhemus.com> (accessed on July 11, 2005)
- [12] John G. Daugman, "How Iris Recognition Works", IEEE Transactions on Circuits and Systems for Video Technology, Vol. 14, No. 1, pp. 21~30, 2004
- [13] Jeong Jun Lee, "Eye Gaze Estimation in Wearable Monitor", Ph.D Thesis, the Graduate School of Yonsei University, 2004
- [14] Y. L. Grand, "Light, Color and Vision", New York: Wiley, 1957
- [15] K. R. Park, "Gaze Detection by Wide and Narrow View Stereo Camera", Lecture Notes in Computer Science (CIARP'2004), Vol. 3287, pp.140~147, 2004
- [16] A. Azarbayejani, et al., "Visually controlled graphics", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 15, No. 6, pp. 602-605, June 1993.
- [17] K. R. Park et al., "A Study on Non-intrusive Facial and Eye Gaze Detection", ACIVS, Vol. 3708, pp.52~59, Sept 20-23, 2005
- [18] T. Ohno et al., "FreeGaze: a gaze tracking system for everyday gaze interaction", in Proc. Symposium on Eye Tracking Research and Applications, pp. 125-132, 2002
- [19] Andrew T. Duchowski et al., "Binocular Eye Tracking in Virtual Reality for Inspection Training,"ACM Press, NY, USA 2002
- [20] [http://www.nacinc.de/english/emr-at\\_voxer.html](http://www.nacinc.de/english/emr-at_voxer.html) (accessed on 2006. 5. 8)
- [21] Dong Hyun Yoo and Myung Jin Chung, "Non-intrusive Eye Gaze Estimation without Knowledge of Eye Pose", Proc. of the Sixth IEEE International Conference on Automatic Face and Gesture Recognition, pp.785~790, 2004

# A New Iris Control Mechanism for Traffic Monitoring System

Young Sung Soh, Youngtak Kwon, and Yakun Wang

Dept. of Information Engineering, Myong Ji Univ., San 38-2,  
Namdong, Yongin, Kyungido, Korea  
{soh, ytkwon, wykluck}@mj.u.ac.kr

**Abstract.** Most of the image-based traffic monitoring system (ITMS) adopts auto iris lens to control the amount of incoming light to camera. Auto iris mechanism measures total light energy in camera's field of view (FOV) and controls iris opening mechanically and inversely proportional to the light energy perceived. Thus, under counterlight, it causes the reduction of incoming light to produce dark scene where brighter one is desirable. To overcome this difficulty, some camera provides a function to define a region of interest (ROI) in FOV and measures light energy only in ROI. Thus, if we leave out counterlight area from ROI, the iris may properly be controlled. However, in ITMS, it frequently happens that large vehicle with white or black roof passes under camera, covers most of the FOV, and results in undesirable iris change. In this paper, we suggest a new iris control mechanism, called user-controlled iris (UCI), in which iris control depends only on background brightness. Since UCI is not sensitive to counterlight or foreground object's brightness, it can maintain the optimal environment for vehicle detection for ITMS.

## 1 Introduction

Image-based traffic monitoring system (ITMS) is widely used to measure traffic parameters such as volume, speed, and occupancy. The first step required in ITMS is vehicle detection and the accuracy of detection determines the quality of ITMS. To detect vehicles robustly, we should be able to obtain a good video input with high contrast between vehicles and road surface irrespective of varying environmental changes including counterlight, sudden illumination changes, and unexpected coverage of camera FOV by large vehicles. Certainly the fixed iris lens cannot be used. Thus most of ITMSs in commercial market use auto iris lens and a few others use ROI-based iris mechanism. The problems with these mechanisms are:

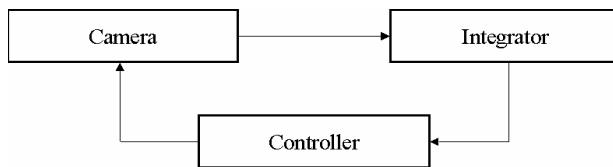
- (1) They cannot cope with environmental changes which happen frequently in real life
- (2) The right of control is on sensor's side, not on user's side

We developed a new iris control scheme UCI where iris is adjusted based only on background brightness, thus overcoming previous difficulties and having the right of control on user's side.

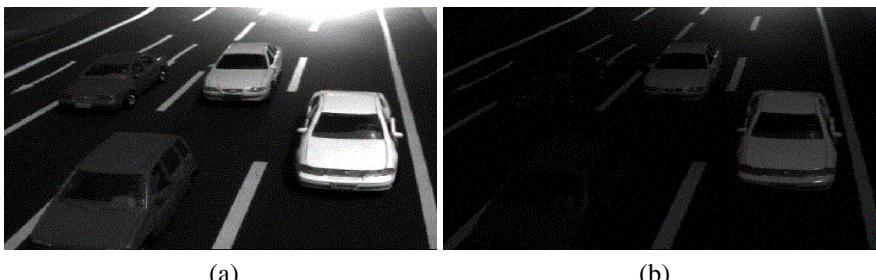
## 2 Related Works

### 2.1 Auto Iris

Auto iris mechanism is depicted in Fig. 1. Integrator integrates output of full camera FOV and Controller controls mechanical iris based on integrated light energy. Since counterlighted area of FOV has an effect of increasing the overall light energy perceived, iris opening becomes narrower, yielding dark scene where brighter one is desirable. In Fig. 2, we show the effect of counterlight in auto iris environment. Fig. 2(a) is before responding to counterlight and Fig. 2(b) is after, respectively. Scene becomes darker and vehicle detection gets more difficult.



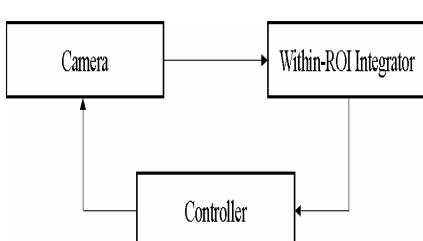
**Fig. 1.** Auto iris mechanism



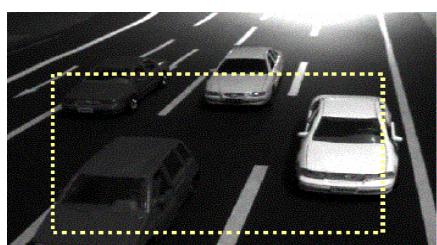
**Fig. 2.** Performance of auto iris under counterlight: (a) before responding, (b) after responding

### 2.2 ROI-Based Iris

Fig. 3 shows the mechanism of ROI-based iris. Basic functions are the same as auto iris except the integrator module. Integration is done only within user-defined ROI.



**Fig. 3.** ROI-based iris mechanism



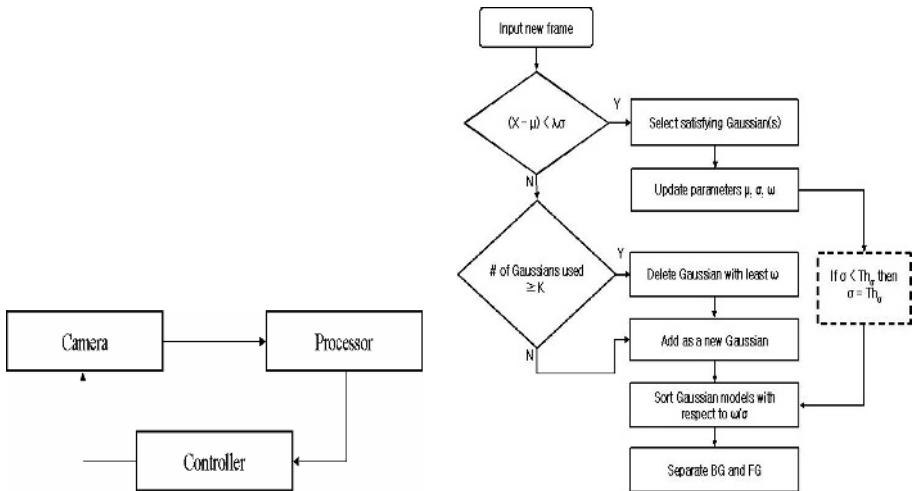
**Fig. 4.** Performance of ROI-based iris

With this ability, we can exclude the counterlighted part from full camera FOV, thus making iris mechanism irrelevant to counterlight effect. Fig. 4 depicts the performance of ROI-based iris. User-defined ROI is drawn as white dotted rectangle. Even in the presence of counterlight, we can clearly see dark vehicles. However, this method has a difficulty if most of the ROI is covered by very bright or dark object surfaces that frequently happens in traffic scene.

### 3 The Proposed Method

#### 3.1 User-Controlled Iris

We propose user-controlled iris (UCI) where the amount of iris opening depends only on light energy from background. UCI mechanism is shown in Fig. 5. Processor gets the input from camera and generates and updates background. Depending on background intensities, Controller generates control signal which is then transmitted to iris control mechanism inside the camera. Lookup table approach is used in determining the amplitude of control signal. Lookup table can be built offline by try and error.



**Fig. 5.** User-controlled iris mechanism

**Fig. 6.** Modified MOG

#### 3.2 Background Generation and Update

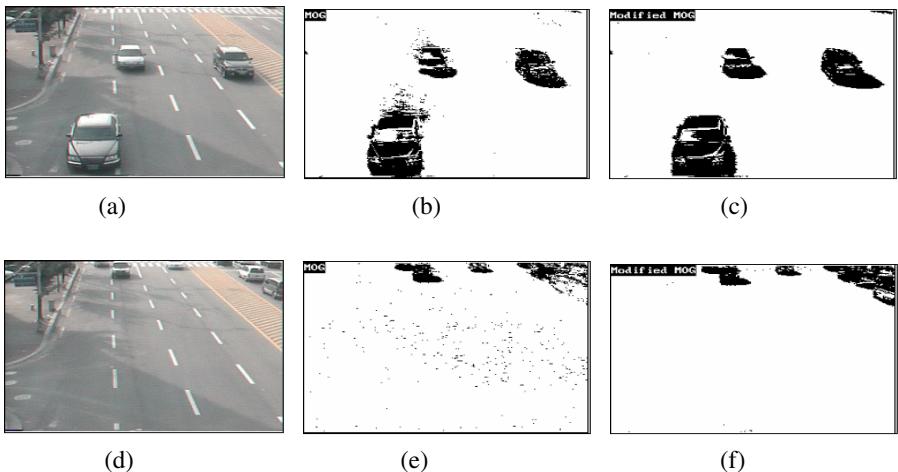
Since UCI depends only on background, we need a robust mechanism to generate and update the background. There have been many approaches to background generation and update. Most of the works assume some probability distribution at each pixel position. Most frequently used probability models include single Gaussian[1~3], double Gaussians[4], and mixture of Gaussians(MOG)[5]. Among them, MOG is proven to be superior to others in that it is able to model multiple backgrounds. However, it suffers from the problem that it is very sensitive to minor variations. That

is, as some distribution keeps getting values close to its mean, it becomes narrower and narrower, thus a value which is only a little bit off the mean is considered to lie in the tail of the distribution resulting in false classification between background and foreground. So we modify MOG update process such that its standard deviation does not fall under some predefined threshold. Modified MOG is depicted in Fig. 6.

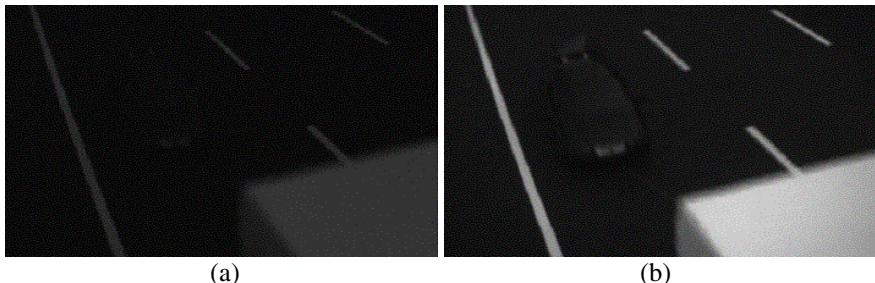
Fig. 6 illustrates MOG generation and update process proposed by Stauffer et al[5]. The dotted box in the right is added for modification. Here  $X$  is input pixel value,  $\mu$  the mean of distribution,  $\sigma$  the standard deviation, and  $\omega$  the weight of the distribution.

## 4 Experimental Results

Fig. 7 shows the result of modified MOG. Fig. 7(a) and (d) are two original images. Fig. 7 (b) and (e) are the corresponding foregrounds obtained by conventional MOG, and Fig. 7 (c) and (f) are the foregrounds extracted by modified MOG. In Fig. 7(b) and (e), many background pixels are classified as foreground since distributions at those pixels were too narrow. They were well eliminated in Fig. 7(c) and (f). In Fig. 8, we demonstrate the effectiveness of UCI. The scenario is that large vehicle with white roof passes under camera and a dark vehicle follows right after it. Fig. 8(a) shows the result when auto iris is used. When the white roof covers most of FOV, iris opening shrinks and scene becomes dark. At that moment a dark vehicle passes, iris opening is not broadened yet, and we have a dark vehicle in a dark scene, resulting in extreme difficulties in vehicle detection. ROI-based iris mechanism cannot solve this problem either. Fig. 8(b) shows the result of our approach. Since iris opening is sensitive only to background intensities, the above scenario does not cause any undesirable phenomenon. Dark vehicle following the white-roof vehicle is visible.



**Fig. 7.** Modified MOG results



**Fig. 8.** Performance comparison between auto iris and UCI: (a) auto iris, (b) UCI

## 5 Conclusion

In this paper, we proposed a new iris control mechanism called UCI for ITMS. We compared UCI with conventional iris control systems including auto iris and ROI-based iris. UCI overcame most of the difficulties possessed by conventional control mechanisms. We demonstrated the result for the situation where conventional approaches fail. Along with a new iris control mechanism, we proposed an MOG-based background generation and update method and demonstrated the performance.

## References

1. Wren, C., Azarbayejani, A., Darrell, T., and Pentland, A. : Pfnder : Real-Time Tracking of the Human Body, *IEEE Trans. Pattern Analysis and Machine Intelligence* vol. 19, no. 7, (1997)780-785
2. Lipton, A., Fujiyoshi, H., and Patil, R. : Moving Target Detection and Classification from Real-Time Video, Proc. IEEE Workshop Application of Computer Vision, (1998)8-14
3. Boult, T. : Frame-Rate Multibody Tracking for Surveillance, Proc. DARPA Image Understanding Workshop, (1998)305-308
4. Haritaoglu, I., Harwood, D., Davis, L.S. : W4: Real-Time Surveillance of People and Their Activities, *IEEE Trans. Pattern Analysis and Machine Intelligence* vol. 22, no. 8, (2000)809-830
5. Stauffer, C., Grimson, W.: Adaptive Background Mixture Models for Real Time Tracking, Proc. Computer Vision and Pattern Recognition, vol. 2, (1999)246-252

# Invariant Object Recognition Using Circular Pairwise Convolutional Networks

Choon Hui Teo<sup>1</sup> and Yong Haur Tay<sup>2</sup>

<sup>1</sup> National ICT Australia Ltd. and  
Research School of Information Sciences and Engineering  
Australian National University  
[choonhui.teo@rsise.anu.edu.au](mailto:choonhui.teo@rsise.anu.edu.au)

<sup>2</sup> Faculty of Information and Communication Technology  
Universiti Tunku Abdul Rahman  
[tayyh@mail.utar.edu.my](mailto:tayyh@mail.utar.edu.my)

**Abstract.** Invariant object recognition (IOR) has been one of the most active research areas in computer vision. However, there is no technique able to achieve the best performance in all possible domains. Out of many techniques, convolutional network (CN) is proven to be a good candidate in this area. Given large numbers of training samples of objects under various variation aspects such as lighting, pose, background, etc., convolutional network can learn to extract invariant features by itself. This comes with the price of lengthy training time. Hence, we propose a circular pairwise classification technique to shorten the training time. We compared the recognition accuracy and training time complexity between our approach and a benchmark generic object recognizer LeNet7 which is a monolithic convolutional network.

## 1 Introduction

*Invariant Object Recognition* (IOR) is a computer vision method which deals with recognizing an interested object in an image under various variations such as affine transformations, distortion, illuminations, pose, occlusion, background, changes in object colors and texture, etc.

Basically, there are two issues associated with IOR on features level. First, the system must be able to detect the object features under various variations and robust to the noise. Second, features defined for objects in one domain (e.g. digit) cannot be used directly for other objects in other domains (e.g. human face).

With convolutional networks (CN) [1], the abovementioned issues can be tackled because CN is a neural network approach which is known to be robust to noise and it facilitates its own feature extraction mechanism without the need of human intervention. However, the shortcoming of CN is computation-intensive which, in turn, causes a lengthy training time.

In this paper, we follow [2] to investigate on the capability of CN in shape-based object recognition invariance to lighting, pose, and background using the proposed method, Circular Pairwise Classification (CPC), to reduce the training time for this problem. Using NORB (NYU Object Recognition Benchmark) [2]

as benchmark dataset, we compared the network training time and recognition accuracy between proposed Circular Pairwise Convolutional Network (CPCN) and LeNet7 developed by LeCun et al. at New York University [2].

## 2 Convolutional Networks (CN)

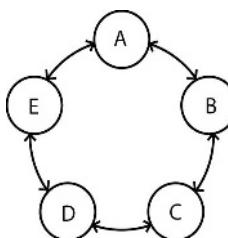
CN (see figure 3(a)) is a kind of multi-layered neural network which facilitates the feature extraction and input-output mapping together with a global learning algorithm. The implicit trainable feature extractor of CN makes it a good candidate for end-to-end object recognition problem. In addition, the trainable feature extractor is adaptable to different problem domains.

In a recognition problem from raw input (e.g. image pixels), CN outperforms Multi-Layered Perceptron (MLP) because the former takes the topology of inputs into consideration while the latter does not [1]. Furthermore, CN combine three architectural ideas, namely *local receptive fields*, *shared weights*, and *spatial subsampling*, to ensure some degree of invariance to shift, scale and distortion [1,3].

## 3 Circular Pairwise Classification

We proposed a pairwise classification framework, *Circular Pairwise Classification*, in which a  $k$ -class classification problem is decomposed into  $k$  two-class classification sub-problems. Each of the classes lies in two different sub-problems, with each sub-problem handled by a pairwise classifier. One can imagine that the  $k$  classes are arranged in a circle whereby each class is only paired with its adjacent left and right neighboring classes. This framework is illustrated in figure 1.

Since each pairwise classifier is trained on an adjacent pair of classes, there is no direct competition between two non-adjacent classes. Consequently, if an unknown input,  $\mathbf{x}$ , is given, we cannot justify that  $\mathbf{x}$  belongs to a class but not the other. Moreover, due to the fact that a pairwise classifier can give erroneous output if  $\mathbf{x}$  does not belong to the pair of classes the pairwise classifier is trained on, chances are high that more than one class will get the maximum votes (i.e., two), especially when the number of classes is more than three (i.e.  $k > 3$ ).



**Fig. 1.** Class pairing in Circular Pairwise Classification. Each small circle here represents a class. This 5-class classification example is decomposed into 5 sub-problems represented by 5 ordered pairs, namely (A, B), (B, C), (C, D), (D, E), and (E, A).

Obviously, this conflict can not be resolved because there is no direct competition between the two conflicting classes (i.e. two adjacent classes will never get maximum votes at the same time). From the above mentioned difficulties, we believe that a naïve configuration of CPC would not work well as it lacks of ‘knowledge’ between the non-adjacent classes. Hence, we propose that an estimation technique should be used to compute these missing ‘knowledge’ from the  $k$ -classifiers’ probabilistic output values.

Given a pairwise classifier (say, MLP),  $\mathbf{C}_{ij}$ , which is trained on the class pair  $(i, j)$  to produce two probabilistic output,  $\mathbf{r}_{ij}$  and  $\mathbf{r}_{ji}$  ( $= 1 - \mathbf{r}_{ij}$ ) where  $\mathbf{r}_{ij}$  and  $\mathbf{r}_{ji}$  are the ratios of probability densities [4], the probability of an unknown input  $\mathbf{x}$ , to belong to class  $i$  and class  $j$  may be computed as such, viz.

$$\mathbf{r}_{ij} = \mathbf{P}(i|\mathbf{x}) / (\mathbf{P}(i|\mathbf{x}) + \mathbf{P}(j|\mathbf{x})) \quad (1)$$

$$\mathbf{r}_{ji} = \mathbf{P}(j|\mathbf{x}) / (\mathbf{P}(i|\mathbf{x}) + \mathbf{P}(j|\mathbf{x})) \quad (2)$$

This is based on Cutzu’s *vote-against* scheme [4]. With another pairwise classifier,  $\mathbf{C}_{jk}$ , and its probabilistic output,  $\mathbf{r}_{jk}$  and  $\mathbf{r}_{kj}$ , we can also estimate  $\mathbf{r}_{ik}$  and  $\mathbf{r}_{ki}$  even though none of the pairwise classifiers are trained with a class pair  $(i, k)$ . The calculation of  $\mathbf{r}_{ik}$  and  $\mathbf{r}_{ki}$  are shown follow:

$$\mathbf{r}_{ik} \approx \mathbf{r}_{ij} \cdot \mathbf{r}_{jk} / (\mathbf{r}_{ij} \cdot \mathbf{r}_{jk} + \mathbf{r}_{kj} \cdot \mathbf{r}_{ji}) \quad (3)$$

$$\mathbf{r}_{ki} \approx \mathbf{r}_{kj} \cdot \mathbf{r}_{ji} / (\mathbf{r}_{ij} \cdot \mathbf{r}_{jk} + \mathbf{r}_{kj} \cdot \mathbf{r}_{ji}) \quad (4)$$

Similar estimation steps apply to all possible pair of classes. In the estimations shown above, we normalized those pair of ratios so that they sum to one; otherwise, the ratios will be very small if the estimation step is lengthy. As the classes are arranged in a circle, we may estimate the ratios in a clockwise or anti-clockwise direction and produce a full set of pairwise ratios. To combine these pairwise ratios into a final decision for the multi-class problem, a meta-classifier such as MLP can be trained to map them into their corresponding desired outputs.

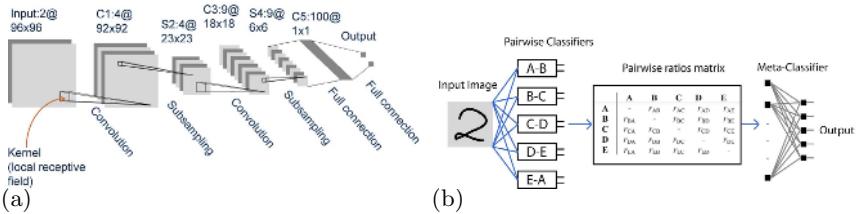
## 4 Experimental Setup

We adopt NORB jittered-cluttered dataset [2,5] as the benchmark dataset. The dataset consists of 5 classes, namely, Animal, Human, Plane, Truck, Car, and Junk (see figure 2). To suit CPC, the original dataset is subdivided into six sub-datasets, in pair: (Animal,Human), (Human, Plane), (Plane, Truck), (Truck, Car), (Car, Junk), and (Junk, Animal).

All pairwise CNs are trained, in parallel, on sub-datasets assigned a priori to them for a maximum of 35 epochs using stochastic backpropagation algorithm with cross-entropy loss. The parameters such as learning rate, momentum coefficient and network complexity are kept fixed for each network. Figure 3(a) illustrates the configuration of a pairwise CN which follows the one in [3]. The connections between layers Input-C1 and S2-C3 are designed to be incomplete to break symmetry in the network and reduce network complexity [1]. The trained



**Fig. 2.** Samples (left camera image) of NORB jittered-clutter dataset. From the left-most column to the rightmost are samples of classes **Animal**, **Human**, **Plane**, **Truck**, **Car**, and **Junk**.



**Fig. 3.** 3(a): Convolutional network for invariant object recognition. 'C' indicates a convolutional layer while 'S' a sub-sampling layer. The Input layer holds two stereo (left and right camera) object images of dimension 96x96 pixels .C1 has four feature maps which convolve inputs from previous layer using six 5x5 convolution kernels. S2 is a 4x4 sub-sampling layer which takes its inputs from C1. C3 has nine feature maps that use 6x6 convolution kernels. S4 is a 3x3 sub-sampling layer. C5 consists of one hundred feature maps that combine all the inputs in S4 through 6x6 kernels. The output layer has two units and is fully connected to C5. 3(b): Example of a circular pairwise convolutional network for a 5-class classification.

pairwise CNs are then combined by stacking them on a meta-classified (e.g. MLP) which maps them to a desired final output (see figure 3(b)).

## 5 Results and Discussion

The recognition accuracy of pairwise CNs, combined pairwise CN and LeNet7 are shown in table 1. The differences in the pairwise CNs test error rates indicated the difficulty in separating objects with different levels of similarity (in terms of object shape). The overall test error rate of combined pairwise CNs is higher than the monolithic CN, LeNet7. This poorer recognition accuracy may be due to insufficient training applied to those pairwise CNs and the estimation errors mentioned in section 3. The reasoning is supported by the fact that LeNet7 actually achieved a lower test error rate of 7.8% [5] (from 16.7% [2]) after it was trained for more iteration and with a set of slightly different learning parameters.

In spite of the poorer performance, pairwise CNs do provide a vehicle for faster training of convolutional network. This can be justified by the fact that

**Table 1.** Test error rates of individual CPCN, meta-classifier, and LeNet7

Pairs of classes	Test error rate (%)	Ref.
(Car, Junk)	5.56	-
(Plane, Truck)	6.96	-
(Junk, Animal)	10.61	-
(Human, Plane)	10.85	-
(Animal, Human)	13.77	-
(Truck, Car)	15.92	-
Combined pairwise CN (meta classifier)	<b>36.06</b>	-
LeNet7 (250k online updates)	<b>16.70</b>	[2]
LeNet7 (> 250k online updates)	<b>7.80</b>	[5]

LeNet7 takes 4.66 million multiply-add operations for full propagation while each pairwise CN takes 1.68 million.

## 6 Conclusion and Future Works

We presented how CN is used as an invariant object recognizer and how the lengthy training process for a monolithic CN can be shortened by using circular pairwise CNs. Although there is a drop in the recognition accuracy, however the accuracy may not be the most important success factor for a recognizer in most cases. In some real world applications it is common that the main priority for the recognizer is to learn quickly and in a distributed manner.

As pairwise classifiers are modular in nature, prior information or desired behavior such as low false positives could be explicitly build into a detection/recognition system by putting more emphasis on the background class. In the invariant object recognition task described above, we can train the pairwise CNs in such a way that each of them learn to recognise a junk class along with another two classes of different objects. The feasibility of this idea is left for future work.

## References

1. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. In: Proceedings of the IEEE. Volume 86. (1998) 2278–2324
2. LeCun, Y., Bottou, L., Huang, F.J.: Learning methods for generic object recognition with invariance to pose and lighting. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition. (2004)
3. Teo, C.H., Tay, Y.H., Lai, W.K.: A novel approach to improve the training time of convolutional networks for object recognition. In: Proceedings of the Twelfth International Conference on Neural Information Processing (ICONIP 2005), Taipei, Taiwan ROC (2005) 17–22
4. Cutzu, F.: Polychotomous classification with pairwise classifiers: a new voting principle. Technical Report TR573, Indiana University (2003)
5. Computational, Biological Learning Lab, N.Y.U.: (Norb: Generic object recognition in images) [Accessed Feb 2005].

# Face Detection Using Binary Template Matching and SVM

Qiong Wang, Wankou Yang, Huan Wang, Jingyu Yang, and Yujie Zheng

School of Computer Science, Nanjing University of Science and Technology,  
Nanjing, China  
nustdaisy@gmail.com

**Abstract.** This paper presents an efficient approach to achieve fast and accurate face detection in still gray level images. The structure of eye region is used as a robust cue to find possible eye pairs. Candidates of eye pair at different scales are discovered by finding regions which roughly match with the binary eye pair template. To obtain real ones, all the eye pair candidates are then verified by using SVM. Faces are finally located according to the eyes position. The proposed method is robust to deal with illumination changes, moderate rotations, glasses wearing and partial face occlusions. The proposed method is evaluated on the BioID face database. Comparative experimental results demonstrate its effectiveness.

## 1 Introduction

Face analysis has become a hot topic in the field of computer vision and pattern recognition in the past decades because of its potential applications, such as biometrics, human computer interaction, and so on. Face detection is a key step of automatic face analysis. The purpose of face detection is to look for and locate faces in given images, and its performance has great effect on face tracking, face recognition, and human gesture analysis.

This paper proposes an efficient approach for face detection in still gray level images. Eye pair candidates are discovered by finding regions which roughly match with the binary eye pair template. The candidates are searched in successively scaled images to find faces in different sizes. Since the algorithm for eye pair candidates selection only uses the structure of eye region as a robust cue, it is expected to be robust to illumination changes, partial face occlusions, facial expressions and moderate rotations.

## 2 Proposed Face Detection Method

Eyes are the most important facial features in face detection and recognition systems. And the structure of eye region is a stable and robust feature which can distinguish eye pair from other patterns. The proposed detection method locates possible eye pairs by binary template matching. Firstly, the facial image is preprocessed to obtain

the structure image. We define the structure image as the binary image which contains the structure feature of human face. Then a binary eye pair template is used to look for eye pair candidates in the image. All the eye pair candidates are then rescaled to a fixed size and sent to the SVM classifier, obtaining the real eye pair. Finally, the face is located in terms of the eye pair position.

## 2.1 Preprocessing

The facial images which will be processed in template matching step are binary images which contain the facial structure information. To obtain good-segmented binary images, four preprocessing algorithms are applied to the input images.

### 2.1.1 Homomorphic Filtering

Homomorphic filtering is a generalized technique for nonlinear image enhancement and correction. It simultaneously normalizes the brightness across an image and increases contrast.

An image can be expressed as the product of illumination and reflectance. The illumination component tends to vary slowly and its frequency fastens on low part in the frequency domain; the reflectance tends to vary rapidly and its frequency is in high part. If two components can be operated separately, the illumination problem will be solved and the image will be enhanced. Hence, the illumination and the reflectance turn to additive through log transform. Then 2-D Fourier transform is used and the coordinate variables are  $u$  and  $v$ .  $H(u, v)$  is the homomorphic filter function applied to the illumination and reflectance, respectively. After taking the inverse Fourier transform and exponent transform, we get enhanced image.  $H(u, v)$  used in this paper has the following form:

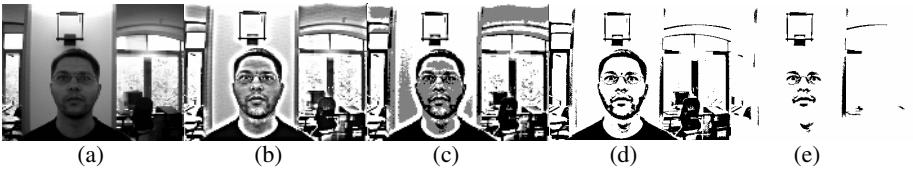
$$H(u, v) = (H_H - H_L) \cdot (1 - \exp(-C \cdot \frac{D}{D_0})) + H_L \quad (1)$$

As shown in Fig. 1, Fig. 1(a) is the input image with low contrast due to the illumination. Fig. 1(b) demonstrates the image enhanced by homomorphic filtering, the contrast is enhanced and more details are available in face region.

### 2.1.2 Clustering and Thresholding

We divide the features of interest from the skin and the background by clustering the gray level image into three clusters through the K-Mean Clustering algorithm. The lightest gray level representing the background or other light pixels is set to 255, the intermediate representing the skin is set to 128, and the darkest representing both the features and other dark pixels of the image is set to 0. Fig. 1(c) shows the result.

After clustering, a threshold is set to 128. Then a binary image is obtained, as shown in Fig. 1(d), which obviously reflects the face structure. Considering the non-face area can influence the speed and the results of template matching, the oversize black area which is useless in the binary image is eliminated by the conventional connected components labeling process. Then the final structure image is obtained, as shown in Fig. 1(e).



**Fig. 1.** An example of preprocessing. (a) Original image. (b) Enhanced image. (c) Clustered image. (d) Binarized image. (e) Structure image.

## 2.2 Binary Template Matching

In order to determine the set of rows which contains the eyes, we apply the template matching to the structure image in order to search possible eye pairs. The problem is that we are not looking for a fixed-shape object. For this reason we adopt a binary eye pair template which models the two eyes in a very rough way but clearly embody the structure of eye region [1]. A single template has been used for all the facial images.

Among the positions with the high cross correlation, all the eye pair candidates are extracted. In the proposed method, some in depth rotation of the face depth or rotation on-the-plane of the image are permitted as long as both eyes of the face are visible. Fig. 2 shows some results of eye pair candidates selection.



**Fig. 2.** Examples of eye pair candidates selection

In order to detect faces in different scales, the facial image is repeatedly scaled by a factor of 1.2. In each scale, all eye pair candidates are extracted and verified by the eyes verifier which will be described in the next section. Consequently, all the faces in one image can be detected.

## 2.3 Eyes Verifier

For the purpose of getting the successful face detection rate, the proposed method is followed by a simple eyes verifier. We use eyes verifier instead of face verifier proposed in some literatures [2], [3], because the eye pattern is considered reliable enough to confirm the existence of the face. We also consider that occlusions on the region below eyes sometimes influence the results if using face verifier.

### 2.3.1 Support Vector Machine

In this paper, we choose the SVM as the classifying function. One distinctive advantage this type of classifier has over traditional neural networks is that SVMs achieve better generalization performance.

Support vector machine is a pattern classification algorithm developed by V. Vapnik and his team [4]. It is a binary classification method that finds the optimal linear decision surface based on the concept of structural risk minimization. SVMs are originally designed to solve problems where data can be separated by a linear decision boundary. By using kernel functions, SVMs can be used effectively to deal with problems that are not linearly separable in the original space. In this paper, we choose Gaussian radial basis function as the kernel function.

### 2.3.2 Eye Pair Candidates Verification

All the eye pair candidates (gray level images) are extracted according to the results of binary template matching. Then they are normalized into the size of  $25 \times 5$  pixels and verified by using SVM to obtain real eye pairs.

The training data used for generating eye verification SVM consists of 400 images of each class (eye-pair and non-eye-pair). Selection of proper non-eye-pair images is very important to train SVM because performance of SVM is influenced by what kind of non-eye-pair images are used. In the initial stage of training SVM, we use non-eye-pair images similar to eye pair such as eyebrows, nostrils and other eye-pair-like patches as eye pair. And we generate non-eye-pair images using bootstrapping method proposed by Sung and Poggio [5].

## 3 Experimental Results

The proposed method is tested on the BioID face database. The BioID face database consists of 1521 images ( $384 \times 286$  pixels, gray level) of 23 different test persons and has been recorded during several sessions at different places. This set features a larger variety of illumination, background and face size. It stresses real world conditions. So it is believed to be more difficult than other dataset containing images with uniform illumination and background.

The eye pair candidates can be selected successfully in most cases, no matter whether face patterns are in different scale, expression, and illumination conditions. Only one robust cue is used for finding eye pair candidates, which is the structure of the eye region. Its robustness can lead to the robustness of the algorithm for eye pair candidates selection. Some successful face detection results are shown in Fig. 3. The



**Fig. 3.** Some detection results

input images vary greatly in background, scale, expression and illumination, the images also including partial face occlusions and glasses wearing. The located eye region and face region are marked with white rectangles, respectively.

The proposed method successfully detected 95.6% faces while the method in Ref. [6] is 91.8%.

## 4 Conclusions and Future Research

In this paper we present an efficient method for detecting faces in still gray level images. Only one robust cue is used for finding eye pair candidates, i.e. the structure of eye region. Eye pair candidates are then sent to SVM classifier to get real eye pair. Finally, faces can be located according to the position of the eyes.

Experimental results show that the proposed eye pair candidates selection may fail in the following conditions:

- glisten of glasses is too strong;
- the facial features is too vague due to the overbright illumination on the face;
- the face slant angle is too large.

Future research will concentrate on making a better use of the facial structure feature to find eyes or face candidates more efficiently.

## Acknowledgment

This work was supported by National Natural Science Foundation of China (Grant No. 60473039, 60503026), and the Project of Science and Technology Plan of Jiangsu Province (Grant No. BG2005008).

## References

1. P. Campadelli, R. Lanzarotti: Localization of Facial Features and Fiducial Points. In Proceedings of the International Conference Visualization, Imaging and Image Processing, Malaga, (2002) 491 – 495
2. J.X Wu, Z.H Zhou: Efficient Face Candidates Selector for Face Detection. Pattern Recognition, Vol. 36. (2003) 1175 – 1186
3. C.C. Han, H.Y.M. Liao, G.J. Yu, L.H. Chen: Fast Face Detection via Morphology-based Pre-processing. Pattern Recognition, Vol. 33. (2000) 1701 – 1712
4. V. Vapnik: The Nature of Statistical Learning Theory. New York: Springer-Verlag, 1995
5. K.K. Sung, T. Poggio: Example-based Learning for View-based Human Face Detection. IEEE Trans. Pattern Anal. Mach. Intell., Vol. 20. No. 1 (1998) 39 – 51
6. O. Jesorsky, K. Kirchberg, R. Frischholz: Robust Face Detection Using the Hausdorff Distance. Lecture Notes in Computer Science, Vol. 2091. (2001) 90 – 95

# **Gain Field Correction Fast Fuzzy c-Means Algorithm for Segmenting Magnetic Resonance Images**

Jingjing Song<sup>1</sup>, Qingjie Zhao<sup>1</sup>, Yuanquan Wang<sup>1</sup>, and Jie Tian<sup>2</sup>

<sup>1</sup> Dept. of Computer Science & Engineering, Beijing Institute of Technology,  
Beijing 100081, P.R. China

{songjj, zhaoqj, yqwang}@bit.edu.cn

<sup>2</sup> Institute of Automation, Chinese Academy of Sciences, Beijing 100080, P.R. China  
jie.tian@mail.ia.ac.cn

**Abstract.** In this paper, we present a new and fast algorithm of fuzzy segmentation for MR image, which is corrupted by the intensity inhomogeneity. The algorithm is formulated by modifying the FFCM algorithm to incorporate a gain field, which compensate for such inhomogeneities. In each iteration, we allow the gain field transforming to a gain field image and filter it using an iterative low-pass filter, and then revert the gain field image to gain field term again for the next iteration. We also use c-means algorithm initializing the centroids to further accelerate our algorithm. Our method reduces lots of executive time and will obtain a high-quality result. The efficiency of the algorithm is demonstrated on different magnetic resonance images.

## **1 Introduction**

Magnetic resonance (MR) images are often corrupted by intensity inhomogeneities in MR imaging[1][10]. It is prone to producing errors by using traditional intensity based segmentation method. Many methods have been proposed to solve this problem[1][2][6][7][10]. In Pham and Prince's AFCM algorithm [6][10], a multiplier field term is incorporated into the FCM objective function to model the brightness variation caused by the inhomogeneity. Besides, the authors add first and second order regularization term into the objective function to ensure that the estimated field is smooth and varies slowly. Without these terms, a multiplier field could always be found to set the objective function to zero[10]. However, AFCM algorithm is sensitive to noise and converge slowly. In Ahmed et al's BCFCM[1], the authors improve the algorithm by including immediate neighborhood as a regularization term in the objective function of FCM. BCFCM algorithm is insensitive to noise, but the computational load is heavy. He et al propose an adaptive FCM algorithm[2]. This algorithm is used to segment three-dimensional multi-spectral MR images. For medical image segmentation, the existing algorithms often take much time in computing, and may be inconvenient for clinical applications.

In this paper, we propose a gain field correction fast fuzzy c-means (GCFFCM) algorithm. A gain filed term is appended to the objective function of fast fuzzy c-means(FFCM) to cut down the influence of intensity inhomogeneity. In each

iteration, we transform the gain field term to gain field image and filter it using an iterative low-pass filter, and then revert the gain field image to gain field term again for the next iteration. We use c-means algorithm initializing the centroids to accelerate our algorithm. The efficacy of our algorithm is demonstrated in section 3.

## 2 Discussion About Algorithms

### 2.1 Standard FCM Algorithm

Suppose a voxel at position  $i$  is modeled as a product of the “true” signal intensity multiplied by a slowly varying factor  $g$  called gain field [7], namely,

$$y_i = g_i x_i + \text{noise}(i) \quad \forall i \in \{1, 2, \dots, N\} \quad (1)$$

where  $y_i$  and  $x_i$  are the observed and the true intensity values at the  $i$ th voxel, respectively,  $\text{noise}(i)$  is the independent white Gaussian distributed noise at voxel  $i$ .  $N$  is the total number of voxels. The objective function of conventional FCM to classify  $x_i$  ( $i=1, 2, \dots, N$ ) into  $c$  clusters can be expressed as[11]:

$$J_{FCM} = \sum_{k=1}^c \sum_{i=1}^N u_{ik}^p \|x_i - v_k\|^2, \quad \sum_{k=1}^c u_{ik} = 1, \quad \forall i, k \quad 0 \leq u_{ik} \leq 1, \quad \forall i, k \quad (2)$$

$u_{ik}$  is the grade of the  $i$ th voxel belonging to class  $k$ , and  $p$  is a weighting exponent which determines the amount of “fuzziness” of a classified result. The norm operator  $\|\cdot\|$  represents the standard Euclidean distance.  $v_k$  is the centroid of class  $k$ .

### 2.2 GCFFCM Algorithm

We could find that if two voxels have same intensity value, they will belong to the same class. Suppose there are  $q$  intensity levels, MR data can be transformed to  $X=\{x_1, x_2, \dots, x_l, \dots, x_q\}$ , where  $h_l$  ( $l=1, 2, \dots, q$ ) denotes the number of voxels with value  $x_l$ . It is similar to the histogram. So the objective function can be written as:

$$J_{FFCM} = \sum_{k=1}^c \sum_{l=1}^q h_l u_{lk}^p \|x_l - v_k\|^2 \quad (\sum_{l=1}^q h_l = N). \quad (3)$$

Since  $q$  is much smaller than  $N$ , FFCM algorithm is much faster than FCM.

Proper initial centroids will improve the accuracy and reduce the number of iterations. If the initial centroids are very far from the real centroids, the segmentation may fail. Thus selecting good initial centroids is a very important step. Considering the fast convergence of the c-means algorithm, we can use this algorithm first, and then treat the results as the initial centroids of our GCFFCM algorithm.

In order to reduce the influence of non-homogeneity, and to incorporate the gain field into the FFCM mechanism, we combine (1) and (3) to yield:

$$J_M = \sum_{k=1}^c \sum_{l=1}^q h_l u_{lk}^p \|y_l - g_l v_k\|^2. \quad (4)$$

To minimize  $J_M$ , we take the first derivatives of  $J_M$  with respect to  $u_{lk}$ ,  $v_k$  and  $g_l$ , then make them to be zero. We gain three conditions below:

$$u_{lk} = \sum_{j=1}^c \left( \frac{y_j - g_j v_k}{y_l - g_l v_j} \right)^{\frac{-2}{p-1}}. \quad (5)$$

$$v_k = \sum_{l=1}^q h_l u_{lk}^p g_l y_l \Big/ \sum_{l=1}^q h_l u_{lk}^p g_l^2. \quad (6)$$

$$g_l = \sum_{k=1}^c h_l u_{lk}^p v_k y_l \Big/ \sum_{k=1}^c h_l u_{lk}^p v_k^2. \quad (7)$$

It is seemed that  $J_M$  can be minimized directly using formula (5), (6) and (7). However, there are two aspects we should consider. Firstly, estimated field should vary slowly and smoothly. AFCM[10] contains first and second order regularization term in the objective function. BCFCM[1] includes a term that considers immediate neighborhood. It is to ensure that the estimated field is smooth and varies slowly. We use another method to solve this problem. An iterative low-pass filter is used to filter the estimated gain field by using (7). The strategy is based on that the gain field is of lower frequency and other parts are of higher frequency. However,  $g_l$  can not be directly filtered, because  $g_l$  loses two-dimensional space information. We should transform  $g_l$  to a two-dimensional gain field image firstly. We can notice that if a voxel has the intensity  $l$ , the value in the two-dimensional gain field image should be  $g_l$ . According to this relation, we can easily get the gain field image, then use the low-pass filter on it. Finally, we should transform the gain field image to  $g_l$  again for the next iteration. Our algorithm can be described as the following:

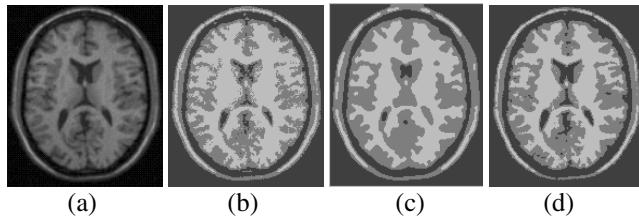
- (1) Use the results of c-means algorithm as the initial centroids, and initialize  $g_l$  with 1.
- (2) Update  $u_{lk}$  using (5).
- (3) Update centroids  $v_k$  using (6).
- (4) Update gain field  $g_l$  using (7).
- (5) Transform  $g_l$  to gain field image, filter the gain field image using an iterative low-pass filter, and revert gain field image to  $g_l$ .
- (6) Return step (2) until  $\|V_{new} - V_{old}\| < \epsilon$ , where  $\epsilon$  is an error threshold .

### 3 Experiments

In this section, we describe the application of our GCFFCM algorithm on MR images. We set fuzzy index  $p=2$ , the termination criterion  $\epsilon=0.01$ , and use an iterative mean filter to smooth the gain field image.

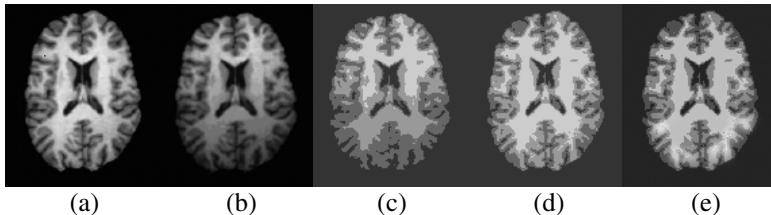
Fig.1 shows the results from FCM, BCFCM and GCFFCM on a T1-weighted MR phantom corrupted with 5% Gaussian noise and 20% intensity non-homogeneity. We can see that FCM algorithm provides an inaccurate segmentation. There is lots of noise

and the contour of white matter is fuzzy. Since considering neighborhood, the BCFCM result is smooth and loses some details in Fig.1(c). Fig.1(d), nevertheless, the GCFFCM provides a better result than the other two algorithm.



**Fig. 1.** FCM, BCFCM and GCFFCM segmentation on a noisy MR image. (a) Original image. (b) FCM segmentation. (c) BCFCM segmentation. (d)GCFFCM segmentation.

Fig.2 shows the results of using the FCM, BCFCM and our algorithm to segment a simulated MR image into 4 classes. The MR image is corrupted by 5% noise and 20% intensity inhomogeneities. Both results using BCFCM and GCFFCM should be acceptable. We use accuracy ratio (ACR:  $100\% \times$ number of correctly segmented voxels / total number of voxels) to measure the segmentation. The ACR of BCFCM and GCFFCM are 90.4% and 92.1%, so our algorithm is better than BCFCM.



**Fig. 2.** Comparison of segmentation results on a noisy simulated MR image. (a) Original image. (b) FCM segmentation. (c) BCFCM segmentation. (d) GCFFCM segmentation.

Besides, we consider the computational complexities of our algorithm. From table 1, we can easily see that if the number of iteration is fixed, the execution time of GCFFCM is much shorter than AFCM and BCFCM. Firstly, AFCM and BCFCM algorithm act on all voxels in a MR image data. However, the number of intensity levels is definite. For 8 bit resolution, there are only 256 intensity levels for each voxel. Hence, the GCFFCM cluster is on a very smaller data space than AFCM and BCFCM algorithm. Secondly, the results of BCFCM may be misclassified if the initial centroids are not appropriate. GCFFCM uses c-means clustering algorithm initializing good centroids. Thirdly, AFCM uses a multigrid algorithm to estimate the multiplier field, and BCFCM uses a regularization term to optimize the bias field. Both of these algorithms are with heavier computational load than GCFFCM.

**Table 1.** Performance time of different algorithm segmenting a 256×256 image into 4classas

Performance time	Number of iteration		
	50	100	200
BCFCM	207sed	450sed	818sed
GCFFCM	6sed	13sed	26sed

## 4 Conclusions

In this paper, we propose an effective GCFFCM algorithm for MR image corrupted by intensity inhomogeneity. The experimental results show that our algorithm is much better than FCM, and the computing speed of GCFFCM is much faster than AFCM and BCFCM. In the next, we plan to integrate a neighborhood regularization term into our algorithm to improve the immunity to noise.

## Acknowledgments

This work was supported by the National Natural Science Foundation of China under grants No.60543007 and No. 60473049.

## References

1. Mohamed N. Ahmed, Sameh M. Yamany et al: A Modified Fuzzy C-Means Algorithm For Bias Field Estimation and Segmentation of MRI Data. IEEE Transactions on Medical Imaging. 21 (3) (2002) 193–199.
2. Renjie He, Sushmita Datta et al: Adaptive FCM with contextual constrains for segmentation of multi-spectral MRI. Proceeding of the 26th Annual International Conference of the IEEE EMBS. 1 (2004) 1660–1663.
3. L. Szilagy, Z. Beny, S. M. Szilagy: Brain Image Segmentation for Virtual Endoscopy. Proceedings of the 26th Annual International Conference of the IEEE EMBS. 1 (2004) 1730–1732.
4. Weijie Chen, Maryellen L. Giger: A fuzzy c-means (FCM) based algorithm for intenity inhomogeneity correction and segmentation of MR images. IEEE International Symposium on Biomedical Imaging. (2004) 1307–1310.
5. Yonggiang Zhao, Minglu Li: A modified fuzzy c-means algorithm for segmentation of MRI. Fifth International Conference on Computational Intelligence and Multimedia Application. (2003) 391 - 395.
6. Dzung L. Pham, Jerry L. Prince: Adaptive Fuzzy Segmentation of Magnetic Resonance Images. IEEE Transaction on Medical Imaging. 18 (3) (1999) 737–752.
7. Lei Jiang, Wenhui Yang: A Modified Fuzzy C-Means Algorithm for Segmentation of Magnetic Resonance Images. Proc. VIIth Digital Image Computing: Techniques and Applications, Sun C., Talbot H., Ourselin S. and Adriaansen T. (Eds.). (2003) 225–231.
8. K. Van Leemput, F. Maes et al: Automated Model-Based Bias Field Correction of MR Images of the Brain. IEEE Transactions on Medical Imaging. 18(10) (1999) 885–896.

9. John G. Sled, Alex P. Zijdenbos et al; A Nonparametric Method for Automatic Correction of Intensity Nonuniformity in MRI Data. *IEEE Transactions on Medical Imaging*. 17 (1) (1998) 87–97.
10. Dzung L. Pham, Jerry L. Prince: An adaptive fuzzy C-means algorithm for image segmentation in the presence of intensity inhomogeneities. *Pattern Recognition Letters*. 20 , (1999) 57–68.
11. J. C. Bezdek, L. O. Hall, L. P. Clarke: Review of MR Image Segmentation Techniques Using Pattern Recognition. *Med. Phys.* 20 (1993) 1033–1048.

# LVQ Based Distributed Video Coding with LDPC in Pixel Domain\*

Anhong Wang<sup>1,2</sup>, Yao Zhao<sup>1</sup>, and Hao Wang<sup>1</sup>

<sup>1</sup> Institute of Information Science, Beijing Jiaotong University, Beijing 100044, China

<sup>2</sup> Taiyuan University of Science and Technology, Taiyuan, China

wah\_ty@yahoo.com.cn, yzhao@center.njtu.edu.cn

**Abstract.** Presently, distributed source coding (DSC) and distributed video coding (DVC) are given high attention in sensor network and other multimedia transmission. That is due to their contribution to easy encoding and robust multimedia communication. In this paper, we propose a new DVC framework where two LVQ (lattice vector quantization) and a rate-variable LDPC (low-density parity-check) are exploited. In both LVQ and LDPC, we use the interpolated from decoded frames as side information to help coding. Because of LVQ's exploitation to dependence of source and the better error-correcting capacity of LDPC, our system achieves more than 1 dB improvement in PSNR than the referenced. Meanwhile, the property of low-complexity encoding is still preserved.

## 1 Introduction

DSC refers to the dependent sources coding with dependence only accessible at decoder. 1973's Slepian-Wolf [1] and 1976's Wyner-Ziv [2] theories state that we can get almost the same compression performance using dependences only at decoder as at both encoder and decoder. These give theoretical base for DSC and DSC gives us support for simplify the encoder.

DVC is born with DSC's application in video coding. Generally, DVC consists of a quantizer following a lossless Slepian-Wolf codec. But in present DVC schemes, such as [3], [4], only SQ is used though paper [5] points that LVQ be optimal. And the Slepian-Wolf codec of [3] [4] are based on a trellis and Turbo coding respectively.

In this paper, we want to set up our DVC based on LVQ and LDPC.

LVQ, as a special vector quantization, can offer better coding performance compared with SQ. And LVQ's superiority over general VQ is that its index of codebook and its decoding can be computed according to the regular algebraic structure of lattices. Also, because fast encoding and decoding algorithm such as in [6], LVQ can be easily used without heavy computation. But the problem from pure LVQ is its large index. So, in our scheme, a so-called LVQ with side information is exploited. In this method only a difference-vector is used to represent a set of lattice points, so the

\* This work was supported in part by National Natural Science Foundation of China (60373028, 90604032), Specialized Research Fund for the Doctoral Program of Higher Education and Program for New Century Excellent Talents in University.

index of LVQ can be compressed again besides the compression in pure LVQ itself. [7] [8] first propose this idea in theory, but they do not use it to real signal processing.

And also, we use side information to help the reconstruction of pure LVQ. Because of the dependence between main and side information, this reconstruction is reasonable. And the regular structure of lattice makes this reconstruction convenient.

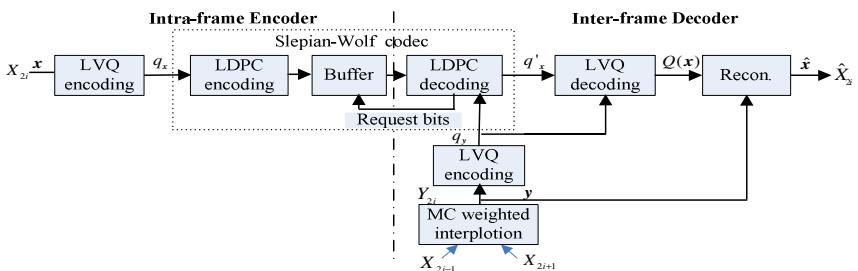
Additionally, a rate-variable LDPC (RV-LDPC) is used to act as Slepian-Wolf codec. LDPC is the most suitable candidate for DSC and now some rate-fixed LDPC has been exploited in DSC [9] [10]. But in practice, the dependence of main information and side information is not known at encoder. So, the rate-fixed LDPC is not enough. Then, we exploit a rate-variable LDPC to meet the varying dependence in video frames. We use feedback and CRC (cyclic redundancy check) to a mother LDPC and send its parity bits in stages on demand from decoder. The rate-variable LDPC in [11] requires sending partial information, which is different from ours.

This paper is organized: in Section 2, we give the proposed DVC framework and in Section 3, the coding detail is presented. Then, some comparison experiments are presented in Section 4. Finally, Section 5 is the conclusion including future work.

## 2 The Proposed Distributed Video Coding Framework

The proposed framework is shown in Fig. 1. It is mainly based on the coding architecture of [12]. But there are differences, such as we replace Turbo coding with LDPC and SQ with LVQ, the other contribution is that we use side information to the reconstruction of pure LVQ.

The system consists of intra-frame encoder and inter-frame decoder. To odd frame,  $X_{2i-1}$  and  $X_{2i+1}$ , conventional intra-frame coding and decoding methods can be used. As in [12], for simplicity, we view  $X_{2i-1}$  and  $X_{2i+1}$  are decoded perfectly. While to even  $X_{2i}$ , its coding process is: at encoder, the adjacent pixels are grouped into vector  $x$ , and  $x$  is quantified by a pair of lattice vector quantizer. Then, the difference symbol of two lattice quantization,  $q_x$ , is encoded by Slepian-Wolf encoder. All parity bits are stored in buffer and ready for being sent on decoder's demands.



**Fig. 1.** The proposed DVC framework in pixel domain

At decoder, firstly, an interpolation (or motion-compensation interpolation) is implemented from  $X_{2i-1}$  and  $X_{2i+1}$ , and we get side information  $Y_{2i}$ ; Next,  $Y_{2i}$  is quantified by two LVQ same as encoder's and get  $q_y$ . Using  $q_y$  and the received parity bits, LDPC decoding is implemented to recovery  $q_x$  and LVQ decoding recovery  $Q(x)$ ; Finally,  $y$  is used to reconstruct  $x$  and the final reconstruction,  $\hat{x}$ , is achieved.

With a pair of lattice quantizer and LDPC we can get better efficiencies than that with SQ and Turbo coding of [12], on the other hand, LVQ based on fast algorithm will not bring much complex encoder than SQ, and the complex motion estimation is omitted from encoder, so, our system has also low-complexity encoder.

Next, we will describe the coding details in our scheme.

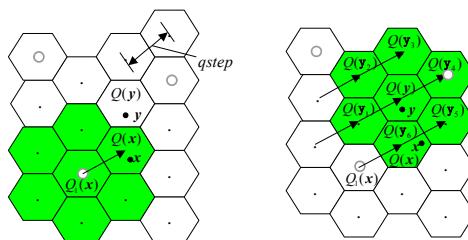
### 3 Implementation Details

#### 3.1 LVQ Coding

The pure LVQ means that an input vector  $x$  is coded by an index of the lattice point  $Q(x)$ , nearest to  $x$ . We use the fast algorithm to get it. While, LVQ coding with side information is shown in Fig. 2, where,  $Q(x)$  and  $Q_i(x)$  are called the lattice point and sublattice point for  $x$  respectively. The coding processing is:

- (1) Getting  $Q(x)$  and  $Q_i(x)$  respectively, then, difference vector  $T(x)$  ,  $T(x) = Q(x) - Q_i(x)$ . And sending  $T(x)$  to decoder.
- (2) Getting  $Q(y)$  and finding all lattice points around  $Q(y)$  , i.e.,  $\{Q(y_1), Q(y_2), \dots, Q(y_{N-1})\}$  , as shown in Fig. 2, where,  $N$  is the number of difference vector.
- (3) Computing  $S_i = T(x) + Q(y_i)$  ( $i = 1,..N$ ) , where,  $Q(y_N) = Q(y)$  .
- (4) Determining the sublattice points from  $S_i$  ( $i = 1,..N$ ) , apparently, only  $Q_i(x)$  is a sublattice point, so we get  $Q_i(x)$  .
- (5) Finding  $Q(x)$  , that is,  $Q(x) = T(x) + Q_i(x)$  .

Apparently,  $T(x)$  represents a set of lattice points. So, if only  $T(x)$  is sent to decoder, compression can be achieved. The compression rate is determined by  $N$  .



**Fig. 2.** Encoding (left) and decoding (right) of LVQ with side information

### 3.2 Reconstruction

In reconstruction, if  $y$  is in the region indicated by  $Q(x)$ ,  $y$  is taken as the reconstruction, i.e.  $\hat{x} = y$ , otherwise, the nearest to  $y$  in the lattice region as the restored.

### 3.3 Slepian-Wolf Codec

Here, for a rate-fixed LDPC, storing all of its parity bit in a buffer, then, based on the demand from decoder, sending partial parity bits stage by stage until MPA (message-passing algorithm) converges. Here, MPA is for the received parity bits and  $q_y$ , which similar to a noisy version of  $q_x$  through a binary symmetric channel.

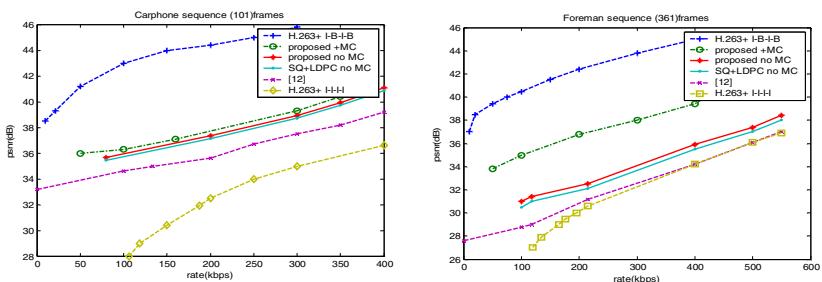
## 4 Experiment Results

The experiment condition is similar to [12] and only luminance of even frames is counted. Lattice  $A_2$  is used and we try  $N=7, 13, 31, 61, 217$  (the bits per pixel is 1.5, 2, 2.5, 3 and 4 respectively) respectively. The experiments include:

- (1) Comparing LVQ+LDPC with SQ+LDPC to check the function of LVQ;
- (2) Comparing the proposed system with the video standard coding H.263+, (I-B-I-B and I-I-I-I)

The results are shown in Fig.3, from the results, we can see that LVQ performs better than SQ in our DVC and the whole system have about 1 dB improvement in PSNR compared with [12], and the proposed system has 4-8 dB PSNR higher than H.263+ intra-frame encoding. Besides, for Foreman sequences with high motion, the MC technology makes the performance better than that has no MC.

In Fig. 4, we show the comparison for interpolated and reconstructed frames. The rate for a pixel is 2.5 bits and average sequence bit-rate is 240kbps, from Fig. 4, we can see that our coding scheme can produce acceptable reconstructed image even in the case of simple average interpolation and it can improve the reconstructed quality even interpolation fails (such as in the case of Fig.4(c)).



**Fig. 3.** Rate-distortion performance of Wyner-Ziv video



**Fig. 4.** 2.5-bit reconstructed frames from Foreman (a) original (b) interpolated-average (c) reconstructed-average(d)interpolated-MC(e)reconstructed-MC

## 5 Conclusion

In this paper, we use lattice vector quantization and channel coding rate variable LDPC to distributed video coding. Experiments results show our system has higher performance than that based on Turbo and SQ and its PSNR is higher than H.263+ intra-frame encoding and decoding. But the expected results have not yet achieved and the future work will be developed at the high-dimensional lattice vector quantization and transform-domain distributed video coding.

## References

1. D.Slepian and J.K. Wolf, "Noiseless coding of correlated information sources ",IEEE Transaction on information theory, vol.19 (1973)471-480.
2. A. Wyner and J. Ziv, "the Rate-distortion Function for source coding with side information at the decoder",IEEE Transaction on information theory,vol.22,no.1 (1976) 1-10.
3. R. Puri and K. Ramchandran, "PRISM: an uplink-friendly multimedia coding paradigm," presented at the Int. Conf. Acoustics, Speech, and Signal Processing (2003) 856-859,.
4. B. Griod, A. Aaron, S. Rane , "Distirbuted video coding", Proceedings of the IEEE, Vol.93, No. 1 (2005) 71-83.
5. D.Rebollo-Monedero, A. Aaron and B. Griod, "Transform for high-rate distributed source coding",in Asilomar Conf. Signals, Systems and Computers, Pacific Grove,CA (2003).
6. J.H. Conway and N.J.A. Sloane. "Fast quantizing and decoding algorithms for lattice quantizers and codes". IEEE Trans. on Information Theory. IT-28(2) (1982) 227-232.
7. Sergio. D. Servetto, "lattice quantization with side information", Date Compression Conference (2000) 510 – 519.
8. Zhixin Liu and Zixiang Xiong,"Slepian-Wolf Coded Nested Quantization (SWC-NQ) for Wyner-Ziv Coding: Performance Analysis and Code Design," Date Compression Conference (2004).
9. D.Schonberg, S.S. Pradhan, and K.Ramchandran, "LDPC codes can approach the Slepian-wolf bound for general binary sources", Allerton Conf. Communication, Control, and Computing ( 2002).
10. A.D. Liverris, Zixiang Xiong, C.N.Georgiades, "compression of binary source with side information at the decoder using LDPC codes", IEEE communications letters,vol.6, NO.10 (2002) 440-442.
11. Mina Sartipi, Faramarz Fekri, "distributed source coding in wireless sensor networks using LDPC coding: the entire Slepian-wolf rate region", IEEE communication society, (2005)1939-1944,.
12. A.Aaron, Rui Zhang and B. Griod, "Wyner-Ziv coding of motion video," Asilomar Conf. Signals and Systems, Pacific Grove,CA (2002).

# Object Matching Using Generalized Hough Transform and Chamfer Matching

Tai-Hoon Cho

School of Information Technology, Korea University of Education and Technology,  
307 Gajun-ri, Chonan, Choongnam, Korea  
thcho@kut.ac.kr

**Abstract.** In this paper, an edge-based matching algorithm is proposed that combines the generalized Hough transform (GHT) and the Chamfer matching to complement weakness of either method. First, the GHT is used to find approximate object positions and orientations, and then these parameters are refined using the Chamfer matching with distance interpolation. Matching accuracy is further enhanced by using a subpixel algorithm. The algorithm was successfully tested on many images containing various electronic components.

## 1 Introduction

Object matching, a very important problem in image analysis, is particularly useful for industrial applications, where a model of an object must be aligned with an image of the object. The transformation obtained by this process can be used for various tasks, e.g., pose adjustment in pick and place operations. Since in many industrial applications the appearance of the object has limited degrees of freedom, rigid transformation (translation and rotation) is often sufficient.

A notable approach among edge-based object matching approaches is Chamfer matching, which was first proposed by Barrow et al. [1], and further improved in [2]. The matching process becomes minimization of the distance between the model edges and the image edges. One critical disadvantage of this approach is that starting positions for the distance minimization should be sufficiently near the global minimum; otherwise, it can be easily trapped on local minima. Instead of Chamfer distance, Hausdorff distance is used in [3] for a distance measure. In [4] it is shown that the additional edge orientation information is beneficial to mainly reduce the number of false positive instances of the model in the image. One drawback of this approach is the computationally too expensive distance transformation for industrial applications.

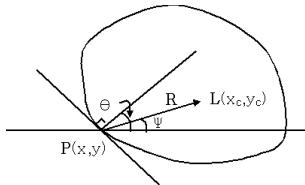
Another class of edge based matching algorithms is based on the generalized Hough transform (GHT) [5]. It is robust to occlusion as well as clutter. Unfortunately, the GHT in the conventional form requires lots of memory and the accuracy obtainable is usually limited due to errors in the estimation of edge directions. A modified GHT approach to remedy some disadvantages of the GHT is suggested in [6]. In this paper, we propose an edge-based matching method that combines the GHT and the Chamfer matching to complement weakness of either method.

## 2 The Proposed Algorithm

In training phase, a model's R-table is constructed from the model edges. In recognition phase, the approximate matching positions found by the GHT using the R-table are refined by the Chamfer matching and subpixel interpolation. For edge extraction, among many edge detectors including the Canny's [7], the Sobel operator [8] is used here for computational efficiency and good performance. To reduce the number of edges, a non-maxima suppression technique [9] is used for thinning edges.

### 2.1 Generalized Hough Transform

The generalized Hough transform (GHT) [5] is designed for detecting arbitrary shapes (Fig. 1). First, a reference point  $L(x_c, y_c)$  is selected inside a given model template consisting of model edges. Let  $\mathbf{R}$  be the vector from an edge point  $P(x, y)$  in the model template to the reference point. The length  $R$  and direction  $\psi$  of each vector  $\mathbf{R}$  is found and stored in a lookup table, called "R-table".  $\theta$  is the edge normal direction of edge point  $P$ , used as an index of the R-table. The R-table is used to find the model in the image. For detecting models that may appear in arbitrary orientation in the search image, a GHT algorithm was devised and is given below in detail.



**Fig. 1.** The generalized Hough transform

- Step 1. From edge points of the model template, an R-table of the model is made.  
 Step 2. If the maximum allowable rotation angle for the model is  $+/-\theta_T$

do the following steps for each integer  $k \in [-\theta_T, +\theta_T]$ .

- 2.1 Construct an accumulator array  $A$  for possible reference coordinates, and initialize to zero.
  - 2.2 For each edge point with edge normal  $\theta$  in the search image,
    - 2.2.1 For the entry of R-table corresponding to  $\theta' = \theta + k$ ,  
 $x_c := x + R(\theta')\cos\psi(\theta');$   $y_c := y + R(\theta')\sin\psi(\theta')$ .
    - 2.2.2  $A(x_c, y_c) := A(x_c, y_c) + 1$
  - 2.3 Record the position  $(x_{m,k}, y_{m,k})$  and  $A_{max,k}$ , where  $A_{max,k}$  is the maximum value of accumulator  $A$  occurred at  $(x_{m,k}, y_{m,k})$ .
- Step 3. Among values of  $A_{max,k}$ ,  $-\theta_T <= k <= \theta_T$ , find maximum  $A_{max,k'}$  greater than a threshold. Then  $k'$  is the orientation and  $(x_{m,k'}, y_{m,k'})$  is the best match position.

The  $\theta$ -th entry of the R-table contains  $R$  and  $\psi$  in the list. Actually, in Step 2, angle index  $k$  may be incremented by the value greater than one, and Step 2.2 is performed only for edge points sub-sampled from the original edge points to reduce the processing time. The accumulator space is divided into quantization cells with uniform size

( $m \times m$  pixels), and the accumulator value is incremented only at the position of the cell that  $(x_c, y_c)$  belongs to, thus allowing a degree of tolerance for positions and orientations of model edges. In Step 3, several candidates can be chosen rather than one maximum to reduce the probability of missing true match position.

As an example, Fig. 2 shows a model image and a search image with edges extracted. Every 16th edge point is used from the original edges in the GHT, the accumulator cell size is  $16 \times 16$ , and angle parameter  $k$  is varied with step size of  $4^\circ$ . We have  $(288, 128, -10)$ ,  $(288, 128, -18)$ , and  $(288, 128, -14)$  for  $(x, y, \text{angle})$  as three most probable parameter sets; the true matching position is near to  $(282, 124, -12)$ .

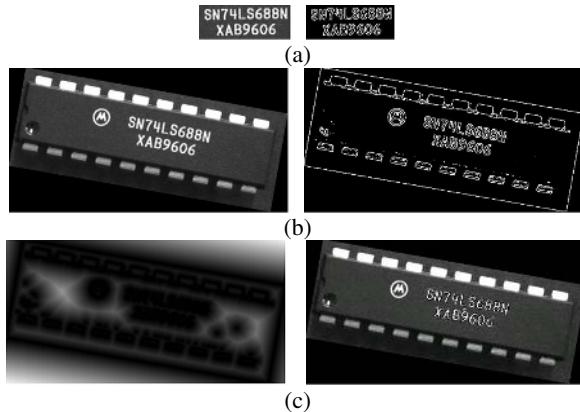
## 2.2 Chamfer Matching

In the Chamfer matching [2], the edge contour of a known object is called “prepolygon image”, and the edge contour of the image, within which the object is searched, is called “predistance image”. Each non-edge pixel in the predistance image is given a distance value from nearest edge pixel using a distance transformation (DT). Since computing true Euclidean distance requires excessive time and memory resources, the 3-4 DT is used as a good approximation of Euclidean distance [2]. The left of Fig. 2(c) shows a distance image obtained by applying the 3-4 DT to the edge image of Fig. 2(b). From the edge list of the prepolygon image, the edge points that are actually used for matching, “polygon”, are chosen. When the polygon is superimposed on the distance image, the root mean square average of the pixels values of the distance image at the polygon coordinates, called the “edge distance”, is computed. A perfect fit will yield the edge distance zero. The position with minimal edge distance is defined as the best fit.

The edge distance is computed for each polygon position determined by the transformation equations. Let  $(x, y)$  be the polygon coordinates and  $(X, Y)$  the position in the distance image. The transformation equations for translation and rotation become  $X = c_x + \cos(r)x - \sin(r)y$  and  $Y = c_y + \sin(r)x + \cos(r)y$ , where  $r$  is the rotation angle, and  $c_x$  and  $c_y$  are the translation parameters in the X- and Y-directions, respectively. The  $(X, Y)$  coordinates are not usually integers, and are rounded to the nearest integer values. However, for more accurate matching, we propose that the distance value at  $(X, Y)$  should be interpolated using its four adjacent pixels in the distance image. A simple bilinear interpolation can be used for this purpose. Finding the optimal polygon position is to search for the global minimum of a 3-dimensional distance function. For efficient search, a hierarchical algorithm is used.

The right of Fig. 2(c) shows matching result using the model of Fig. 2(a); model edges in black are accurately overlaid on the test image. (+ denotes the reference point of the model, center of the model image.) Table 1 shows the results of applying Chamfer matching without/with bilinear interpolation in the distance image. Notice that the distance value is considerably lowered by using the distance interpolation.

To find more accurate matching parameters, we use a subpixel algorithm based on a simple 1-dimensional parabolic interpolation method using three consecutive points. The 1-d interpolation is applied independently for each of three parameters  $x$ ,  $y$ , and angle. Table 1 shows the results of applying the subpixel algorithm to Fig. 2(c) without/with distance interpolation. Note that the distance value is further reduced by the subpixel algorithm.



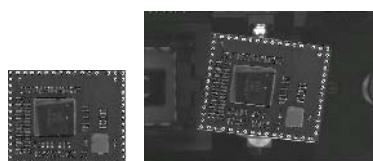
**Fig. 2.** (a) A model image (size:164x63) and its edges. (b) A search image (size:500x259), in which the model is searched, and its edges. (c) A distance image of the edge image in Fig. 2(b) and the result of matching using the model of Fig. 2(a).

**Table 1.** The results of applying Chamfer matching without/with distance interpolation followed by the subpixel algorithm

	Position[pixel]	Angle[degree]	Distance
Before subpixel algorithm	(282, 124)	-12	0.564/0.468
After subpixel algorithm	(282.32, 123.83)	-11.95	0.466/0.377

### 3 Experimental Results

The proposed algorithm was tested on a number of images containing various electronic components with excellent performance. An example is shown in Fig. 3. The accuracy of our algorithm was also measured. To measure x, y accuracy, an image containing Fig. 2(a) as a subimage, is translated in x and y direction in the range of -0.8 ~ 0.8 pixel by step size of 0.2 pixel. Using these 80 images, the algorithm was run to obtain standard deviation and maximum error. Also, the test image was rotated within the range of +/-30° by step size of 1° to yield 60 images. These images were used to produce estimation error of angle. The results are shown in Table 2, which demonstrates outstanding accuracy of our algorithm. In the measurement, the model image has size 164x63 with 1742 edges, while the search image has size 640x480 with about 5000 edges; our algorithm takes approximately 150 msec on the PIII 800MHz.



**Fig. 3.** (a) A model image. (b) A search image.

**Table 2.** Accuracy of position and angle estimated

	x [pixel]	y [pixel]	angle [degree]
Standard dev.	0.093	0.087	0.07
Max. error	0.14	0.12	0.15

## 4 Conclusions

An edge-based matching method is proposed that combines the GHT and the Chamfer matching to complement weakness of either method. First, the GHT is used to find approximate object positions and orientations, and then these parameters are refined using the Chamfer matching with distance interpolation. Matching accuracy is further enhanced by using a subpixel algorithm. The algorithm was successfully tested on a number of images containing various electronic components.

**Acknowledgement.** This work was supported by grant No. RTI04-01-02 from the Regional Technology Innovation Program of the Ministry of Commerce, Industry and Energy(MOCIE).

## References

1. H.G. Barrow, J.M. Tenenbaum, R.C. Bolles, and H.C. Wolf, "Parametric correspondence and chamfer matching: Two new techniques for image matching," *Proc. 5th Int. Joint Conf. Artificial Intelligence*, Cambridge, MA, pp.659-663, 1977.
2. G. Borgefors, "Hierarchical chamfer matching: a parametric edge matching algorithm," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol.10, no.6, pp. 849-865, 1988.
3. W.J. Ruckridge, "Efficiently locating objects using the Hausdorff distance," *International Journal of Computer Vision*, vol.24, no.3, pp.251-270, 1997.
4. C.F. Olson and D.P. Huttenlocher, "Automatic target recognition by matching oriented edge pixels," *IEEE Trans. on Image Processing*, vol.6, no.1, pp.103-113, 1997.
5. D.H. Ballard, "Generalizing Hough transform to detect arbitrary shapes," *Pattern Recognition*, vol.13, no.2, pp.111-122, 1981.
6. M. Ulrich, C. Steger, A. Baumgartner, and H. Ebner, "Real-time object recognition in digital images for industrial applications," *5<sup>th</sup> Conf. on Optical 3-D Measurement Techniques*, Vienna, pp.308-318, 2001.
7. J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Analy. and Mach. Intelli.*, vol.8, no.6, pp.679-698, 1986.
8. E.R. Davies, *Machine Vision*, 3rd Ed., Morgan Kaufmann, 2005.
9. R. Jain, R. Kasturi, and B.G. Schunck, *Machine Vision*, McGraw-Hill, 1995.

## ERRATUM

# An Adaptive Inventory Control Model for a Supply Chain with Nonstationary Customer Demands

Jun-Geol Baek<sup>1</sup>, Chang Ouk Kim<sup>2</sup>, and Ick-Hyun Kwon<sup>3</sup>

<sup>1</sup> Department of Industrial Systems Engineering, Induk Institute of Technology,  
Wolgye-dong, Nowon-gu, Seoul, 139-749, Republic of Korea

<sup>2</sup> Department of Information and Industrial Engineering, Yonsei University,  
Sinchon-dong, Seodaemun-gu, Seoul, 120-749, Republic of Korea

<sup>3</sup> Department of Industrial Systems and Information Engineering, Korea University,  
Anam-dong, Sungbuk-gu, Seoul, 136-701, Republic of Korea

Q. Yang and G. Webb (Eds.): PRICAI 2006, LNAI 4099, pp. 895 – 900, 2006.  
© Springer-Verlag Berlin Heidelberg 2006

---

**DOI 10.1007/11801603\_102**

In the original version of the paper one author was missing. The correct author list of the paper is as follows:

Jun-Geol Baek<sup>1</sup>, Chang Ouk Kim<sup>2</sup>, Jin Jun<sup>3</sup>, and Ick-Hyun Kwon<sup>4</sup>

<sup>1</sup> Department of Industrial Systems Engineering, Induk Institute of Technology,  
Wolgye-dong, Nowon-gu, Seoul, 139-749, Republic of Korea

<sup>2</sup> Department of Information and Industrial Engineering, Yonsei University,  
Sinchon-dong, Seodaemun-gu, Seoul, 120-749, Republic of Korea

<sup>3</sup> Department of Computer Science and Engineering, University of Connecticut,  
Storrs, CT 06269-2155, USA

<sup>4</sup> Department of Industrial Systems and Information Engineering, Korea University,  
Anam-dong, Sungbuk-gu, Seoul, 136-701, Republic of Korea

---

The original online version for this chapter can be found at  
[http://dx.doi.org/10.1007/11801603\\_102](http://dx.doi.org/10.1007/11801603_102)

# Author Index

- Abiyev, Rahib H. 701  
Ahn, Byung-Ha 747  
Akhetmetov, Dauren F. 288  
An, Jaebum 875  
Anandha Mala, G.S. 1155  
Ashoori, Maryam 101  
Baek, Jun-Geol 895  
Bao, Yukun 1073  
Bi, Yaxin 1068  
Bin, Hu 358  
Bo, Liefeng 365  
Bo, Yang 1191  
Bonzon, Elise 41  
Cai, Guoyong 1052  
Cao, Chunhong 940  
Cao, Longbing 711  
Cao, Minh Duc 413  
Chan, Yung-Kuan 1094  
Chao, Ya-Ting 1037  
Chau, Kwok-wing 1211  
Chau, Rowena 1160  
Che, Chan Hou 918  
Chen, Aixiang 180  
Chen, Guan-Hau 1017  
Chen, Jianyun 524  
Chen, Jisong 1160  
Chen, Li-an 934  
Chen, Min 787  
Chen, Qingfeng 222  
Chen, Shu-Yuan 1037  
Chen, Songcan 404  
Chen, Ta-Cheng 777  
Chen, Yi-Ping Phoebe 222  
Chen, Ying-Wu 767  
Chen, Yiqiang 563  
Chen, Yung-Fu 1094  
Cheng, Chung-Chuan 1094  
Chiao, Chin-Pu 1119  
Cho, Sung-Bae 807, 950, 1196  
Cho, Tai-Hoon 1253  
Choi, Jae Ho 573  
Choi, Jeoung-Nae 829  
Choi, Jung-Wook 870  
Chou, Chien-Chang 161  
Chou, Wusheng 1186  
Chu, Shu-Chuan 854  
Dai, Honghua 434, 553  
Domingos, Pedro 1  
Duan, Miyi 524  
Fan, Yanfeng 261  
Fang, Wei 737  
Fu, Pan 1078  
Fukuda, Kensuke 818  
Gao, Xiaoying 413, 923  
Geetha, T.V. 839  
Geng, Xin 553  
Gero, John S. 21  
Ghose, Aditya 907  
Goh, Angela Eck Soong 101  
Gong, Maoguo 661  
Governatori, Guido 31, 71, 150  
Gu, Tianlong 1052  
Gu, Xingsheng 757  
Guha, Prithwijit 593  
Guo, Gongde 1068  
Guo, Lei 1216  
Guo, Ming 1206  
Han, Dingyi 81  
Han, Dongsoo 51  
Han, Min 1047  
Han, Seung Soo 350, 995  
Harvey, Peter 907  
Hashimoto, Yuya 890  
He, Wei 1058  
He, Yanxiang 1135, 1140  
Hirotsu, Toshio 818  
Hoffmann, Achim 464, 533, 1150  
Hong, Jin-Hyuk 807, 1196  
Hong, Kuo-Lung 1094  
Hong, Kwangjin 632  
Hope, A.D. 1078  
Hou, Yimin 1216

- Hsieh, Y.-C. 929  
Hsu, Sheng-Cheng 1119  
Hu, Kunyuan 960  
Hu, Qinghua 423  
Hu, Xiaohua 1145  
Huang, Hong 1058  
Huang, Jinjie 721  
Huang, Tianyu 603  
Huang, Wei 180, 980, 1073  
Huang, Xuming 1068  
Hui, Liu 358  
  
Ishizuka, Mitsuru 651  
  
Jang, Bong Won 583  
Jang, Seong-Whan 829  
Jeon, In Ja 730  
Jeon, Moon-Gu 747  
Ji, Donghong 1135, 1140  
Jiang, Liangxiao 375, 970  
Jiang, Lijun 444, 681, 985  
Jiang, Yongguang 251  
Jiang, Yunfei 180, 844  
Jiang, Zhiwei 1027  
Jiao, Licheng 365, 661, 1032  
Jin, Peng 960  
Jin, TaeSeok 1063, 1083  
Jin, Zhi 130  
Joo, Hyun Jea 583  
Jung, Chai-Yeoung 1181  
Jung, Keechul 632, 955  
Jung, Kyung-Yong 112  
Jung, Yuchul 1099  
  
K.C., Santosh 990  
Kang, Rae-Goo 1181  
Kang, Sanggil 51  
Karim, Samin 200  
Kayasith, Prakasith 885  
Khemani, Deepak 474  
Khorasani, K. 308  
Kijjsirikul, Boonserm 975  
Kim, Byungwhan 350, 995  
Kim, Chang Ouk 895  
Kim, Deok-Hwan 641  
Kim, Dong Hwan 995  
Kim, Ho-Joon 1171  
Kim, Hyun-Ki 834  
Kim, Jong-Hwan 747  
Kim, Jong-kwon 1083  
  
Kim, Kyoung-Min 807  
Kim, Kyung-Joong 950  
Kim, Seong-Whan 61, 623  
Kim, Seung-Woo 870  
Kim, Wan-Su 834  
Kim, Wookhyun 1114  
Ko, Ki-Hong 61, 623  
Ko, Nak Yong 190  
Ko, Wooram 350  
Komatani, Kazunori 484  
Kotaki, Minoru 288  
Kumar, Sathish 839  
Kurihara, Satoshi 818  
Kwak, Hoon Sung 573  
Kwon, Ick-Hyun 895  
Kwon, Youngtak 1227  
  
Lagasquie-Schiex, Marie-Christine 41  
Lai, Kin Keung 980  
Lam, Yuk-Hei 849, 945  
Lang, Jérôme 41  
Lao, Songyang 524  
Lee, Cheng-Hwa 671  
Lee, Chong Ho 1201  
Lee, Hwanju 544  
Lee, Hyun-Ju 505  
Lee, HyunJung 901  
Lee, Jian-Hau 1119  
Lee, Juho 1171  
Lee, Sang-Jo 505  
Lee, Sangkyung 955  
Lee, Sangyoon 875  
Lee, Seok-Lyong 641  
Lee, Sungdoke 51  
Lee, Wei-Po 823  
Lee, Wooboom 1114  
Lee, Yong Min 1201  
Lee, Yunsik 298, 329  
Leenen, Louise 907  
Legaspi, Roberto 890  
Li, BiCheng 515  
Li, Chaoqun 375  
Li, Chuan 251  
Li, Fengxia 603  
Li, Gang 1005  
Li, Guo-Zheng 271  
Li, Jianwei 1058  
Li, Jianzhong 965  
Li, Jinyan 241  
Li, Lei 1125

- Li, Lianggang 222  
 Li, Qing 1032  
 Li, Qingyong 691  
 Li, Sufen 960  
 Li, Wenlong 721  
 Li, Wenhui 940  
 Li, Xiaodong 423  
 Li, Z.Q. 308  
 Lim, Andrew 918  
 Lin, Tsun-Chen 1037  
 Lin, Zuoquan 130  
 Liu, Dexi 1135, 1140  
 Liu, Dong 1005  
 Liu, Hongbo 1109  
 Liu, Jin 787  
 Liu, Junfa 563  
 Liu, Qihong 251  
 Liu, Ru-Sheng 1037  
 Liu, Tian-Yu 271  
 Liu, Yan 767  
 Liu, Yi 278  
 Liu, Yin 515  
 Liu, Yintian 251  
 Liu, Yuechang 844  
 Liu, Yunxia 1042  
 Liu, Zhitao 1073  
 Lou, Weijun 923  
 Lu, Ruqian 130  
 Lun, Xiangmin 1216  
 Luo, Dan 711  
 Luo, Jiewen 91  
 Lv, Ning 721
- Ma, L. 308  
 Ma, Wenping 661  
 Ma, Yuejin 413  
 Mala, T. 839  
 Marukatrat, Sanparith 454  
 Menekay, Mustafa 701  
 Meyer, Thomas 907  
 Miao, Chun Yan 101  
 Min, Byung-Jae 1104  
 Min, Jianyuan 603  
 Mu, Kedian 130  
 Mu, WenBin 515  
 Muhammad, Rafiq 120  
 Mukerjee, Amitabha 593  
 Murakami, Tomoko 912  
 Myaeng, Sung Hyon 1099
- Nakadai, Kazuhiro 484  
 Nakano, Mikio 484  
 Nam, Mi Young 394, 730  
 Nattee, Cholwich 990  
 Numao, Masayuki 890
- Ogata, Tetsuya 484  
 Oh, Dong-Ik 870  
 Oh, Sung-Kwun 829, 834  
 Okuno, Hiroshi G. 484  
 Ong, Kok-Leong 849, 945  
 Ono, Masahiro 651  
 Orihara, Ryohei 912  
 Ou, Yu-Yen 1017  
 Oyang, Yen-Jen 1017
- P, Deepak 474  
 Padmanabhan, Vineet 31, 150  
 Pan, Haiwei 965  
 Pan, Jeng-Shyang 854  
 Park, ChangHoon 1063  
 Park, Dong-Chul 298, 329, 1104  
 Park, Hogun 1099  
 Park, Jae Young 995  
 Park, Jong-Hwan 747  
 Park, Kang Ryoung 1222  
 Park, Sancho 1104  
 Park, Se-Young 505, 1000  
 Park, Seong-Bae 505, 1000  
 Park, Soo-hong 1063, 1083  
 Pattamavorakun, Suwarin 339  
 Pattamavorakun, Suwat 339  
 Pedrycz, Witold 834  
 Peng, Wei 21  
 Peng, Yuhua 278, 385, 1042  
 Phienthrakul, Tanasanee 975  
 Poh, Kim-Leng 212  
 Poonyutthikul, Sunatashee 231
- Qian, Dongping 923  
 Qian, Junyan 1052  
 Qian, Yuntao 864  
 Qin, Yongsong 1010  
 Qiong, Wang 101  
 Qun, Niu 757
- Rahwan, Iyad 11  
 Rao, Delip 474  
 Rhee, Phill Kyu 394, 583, 730  
 Rong, Jia 1005

- Rotolo, Antonino 31  
 Ryu, Tae-Wan 1171  
 Sato, Shin-ya 818  
 Sedai, Suman 394, 583  
 Seo, Dong Jin 190  
 Shadbolt, Nigel R. 444, 681, 985  
 Shang, Ronghua 661  
 Shen, Bin 859  
 Shen, Ji-hong 319  
 Shi, Chuan 691  
 Shi, Dongyu 140  
 Shi, Yongliang 787  
 Shi, Zhongzhi 91, 691  
 Simmons, Reid G. 190  
 Soh, Young Sung 544, 1227  
 Sohn, Mye M. 901  
 Son, Jeong-Woo 1000  
 Sonenberg, Liz 11, 200  
 Song, Yixu 1109  
 Song, Young-Soo 298  
 Song, Insu 71  
 Song, Jingjing 1242  
 Sornil, Ohm 231  
 Srisawat, Anantaporn 975  
 Stuckey, Peter J. 120  
 Subagdja, Budhitama 11  
 Sugawara, Toshiharu 818  
 Sun, Jun 737  
 Sun, Ming 319  
 Sun, Xu 495  
 Takeda, Ryu 484  
 Tan, Ah-Hwee 200  
 Tang, Changjie 251  
 Tay, Yong Haur 1232  
 Teo, Choon Hui 1232  
 Theeramunkong, Thanarak 885  
 Thubthong, Nuttakorn 885  
 Tian, Jie 1242  
 Tian, Zheng 1166  
 Tran, Chung Nguyen 298, 329, 1104  
 Tsai, Pei-wei 854  
 Tsujino, Hiroshi 484  
 Uma, G.V. 1155  
 Valin, Jean-Marc 484  
 Vellaisamy, Kuralmani 241  
 Venkatesh, K.S. 593  
 Wang, Anhong 1248  
 Wang, Hao 1248  
 Wang, Houfeng 495  
 Wang, Huan 1237  
 Wang, Jiaxin 1109  
 Wang, Jingmin 1022  
 Wang, Lei 1206  
 Wang, Li 1125  
 Wang, Limin 940  
 Wang, Ling 365  
 Wang, Ning 1089  
 Wang, Qiong 1237  
 Wang, Rusong 787  
 Wang, Shiming 797  
 Wang, Shouyang 980  
 Wang, Yakun 544, 1227  
 Wang, Yuanquan 1242  
 Wang, Yuanyong 1150  
 Wang, Zhenyu 918  
 Wang, Ziqiang 261  
 Wei, Kongming 1186  
 Wen, Guihua 444, 681, 985  
 Wen, Jun 444, 681, 985  
 Wen, Xian-Bin 1166  
 Wen, Zhonghua 180  
 Wong, Wing-Kwong 1119  
 Wu, Kanzhang 1022  
 Wu, Shengli 1068  
 Wu, Tingting 524  
 Wu, Zhao 1135  
 Wu, Zhong 1186  
 Xi, Jianhui 1047  
 Xiang, Yanping 212  
 Xiao, Dongrong 563  
 Xie, Pith 563  
 Xie, Xiaoqin 965  
 Xu, Erqing 1130  
 Xu, Jinhua 1176  
 Xu, Wenbo 737  
 Xu, Yao-qun 319  
 Xue, Gui-Rong 81  
 Yamamoto, Shun'ichi 484  
 Yan, Shi-Liang 813  
 Yang, Heng-Li 671  
 Yang, Hua 1135, 1140  
 Yang, Hyun-Seung 1171  
 Yang, Jingyu 1237  
 Yang, Ming 385

- Yang, Shuyuan 1032  
Yang, Tiejun 261  
Yang, Wankou 1237  
Yang, Yahong 880  
Yang, Yu-Fen 1119  
Yang, Zehong 1109  
Yao, Min 859, 864, 1027  
Ye, Yangdong 1005  
Yeh, Chung-Hsing 1160  
Yeh, Hui-Chin 1119  
Yeo, Seong Won 1201  
Yi, Huawei 880  
Yi, Wensheng 859, 1027  
Yoo, Seung Yeol 464, 533  
You, Jinyuan 140  
You, Kang Soo 573  
You, P.-S. 929  
Yu, Daren 423  
Yu, In Gab 1201  
Yu, Jinmei 1089  
Yu, Kyeonah 170  
Yu, Lean 980  
Yu, Yong 81  
Yu, Zhonghua 251  
Yuan, Xin 1206  
Yuehui, Chen 1191  
Yumaryanto, Abdul A. 875  
  
Zhan, Shouyi 603  
Zhang, Bin 940  
Zhang, Caiming 278  
Zhang, Chengqi 222, 711, 1010  
Zhang, Daoqiang 404  
Zhang, Dexian 261  
Zhang, Harry 970  
  
Zhang, Hua 1166  
Zhang, Jian 1010  
Zhang, Ke 797  
Zhang, Mengjie 413, 923  
Zhang, Peiming 934  
Zhang, Qiuyu 880  
Zhang, Rui 1073  
Zhang, Shichao 1010  
Zhang, Tianqing 251  
Zhang, Wei 965  
Zhang, XianFei 515  
Zhang, Xiaodan 1145  
Zhang, Yu 495  
Zhang, Zhiyong 691  
Zhang, Zili 849, 945  
Zhao, Fuqing 880  
Zhao, Lingzhong 1052  
Zhao, Qingjie 1242  
Zhao, Xinyu 524  
Zhao, Yao 1248  
Zheng, Qiuhua 864  
Zheng, Yujie 1237  
Zhong, Ning 2  
Zhou, Chang-Feng 767  
Zhou, Jin 1191  
Zhou, Ke-Feng 813  
Zhou, Xiaohua 1145  
Zhou, Xinhong 385, 1042  
Zhou, Zhi-Hua 5, 404, 553  
Zhu, Mingfang 251  
Zhu, Xiangbin 613  
Zhu, Xiaofeng 1010  
Zhu, Yunlong 960  
Zhuang, Ling 434