

LNAI 8032

Zengchang Qin
Van-Nam Huynh (Eds.)

Integrated Uncertainty in Knowledge Modelling and Decision Making

International Symposium, IUKM 2013
Beijing, China, July 2013
Proceedings



Springer

Lecture Notes in Artificial Intelligence 8032

Subseries of Lecture Notes in Computer Science

LNAI Series Editors

Randy Goebel

University of Alberta, Edmonton, Canada

Yuzuru Tanaka

Hokkaido University, Sapporo, Japan

Wolfgang Wahlster

DFKI and Saarland University, Saarbrücken, Germany

LNAI Founding Series Editor

Joerg Siekmann

DFKI and Saarland University, Saarbrücken, Germany

Zengchang Qin Van-Nam Huynh (Eds.)

Integrated Uncertainty in Knowledge Modelling and Decision Making

International Symposium, IUKM 2013
Beijing, China, July 12-14, 2013
Proceedings



Springer

Volume Editors

Zengchang Qin
Beihang University
School of Automation Science and Electrical Engineering
37 Xueyuan Road, Beijing 100191, China
E-mail: zcqin@buaa.edu.cn

Van-Nam Huynh
Japan Advanced Institute of Science and Technology
School of Knowledge Science
1-1 Asahidai, Nomi, Ishikawa, 923-1292, Japan
E-mail: huynh@jaist.ac.jp

ISSN 0302-9743

e-ISSN 1611-3349

ISBN 978-3-642-39514-7

e-ISBN 978-3-642-39515-4

DOI 10.1007/978-3-642-39515-4

Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2013942382

CR Subject Classification (1998): I.2.6, I.2, H.2.8, F.2, I.4, I.5, F.1, H.3

LNCS Sublibrary: SL 7 – Artificial Intelligence

© Springer-Verlag Berlin Heidelberg 2013

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed. Exempted from this legal reservation are brief excerpts in connection with reviews or scholarly analysis or material supplied specifically for the purpose of being entered and executed on a computer system, for exclusive use by the purchaser of the work. Duplication of this publication or parts thereof is permitted only under the provisions of the Copyright Law of the Publisher's location, in its current version, and permission for use must always be obtained from Springer. Permissions for use may be obtained through RightsLink at the Copyright Clearance Center. Violations are liable to prosecution under the respective Copyright Law.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

This volume contains papers presented at the 2013 International Symposium on Integrated Uncertainty in Knowledge Modelling and Decision Making (IUKM 2013), which was held at Beihang University, Beijing, China, during 12–14 July, 2013. The principal aim of IUKM 2013 was to provide a forum in which researchers could exchange ideas and results on both theoretical and applied research relating to all aspects of uncertainty management and their applications.

The organizers received 49 papers. Each paper was peer reviewed by two members of the Program Committee. Finally, 19 papers were chosen for presentation at IUKM 2013 and for publication in the proceedings. The keynote and invited talks presented at the symposium are also included in this volume.

As a follow-up of the symposium, a special issue of the *International Journal of Approximate Reasoning* is anticipated to include a small number of extended papers selected from the symposium as well as other relevant contributions received in response to subsequent open calls. These journal submissions are to go through a fresh round of reviews in accordance with the journal's guidelines.

We would like to express our appreciation to the members of the Program Committee for their support and cooperation in this publication. Last, but not the least, we wish to thank all the authors and participants for their contributions and fruitful discussions that made this symposium a success.

May 2013

Zengchang Qin
Van-Nam Huynh

Organization

General Co-chairs

Yoshiteru Nakamori
Zhimin Wei
JAIST, Japan
Beihang University, China

Program Chairs

Van-Nam Huynh
Zengchang Qin
JAIST, Japan
Beihang University, China

Local Coordination Committee

Baochang Zhang
Fei Tao
Haibin Duan
Huaining Wu
Shiyin Qin
Zheng Zheng
Beihang University, China
Beihang University, China

Treasurer

Emy Yu
Beihang University, China

Secretary/Web Administrator

Farhan Khawar
Beihang University, China

Technical Program Committee

Andrzej Skowron
Bernadette Bouchon-Meunier
Bernard De Baets
Byeong Seok Ahn
Churn-Jung Liau
Enrique Herrera-Viedma
Etienne E. Kerre
Eyke Hüllermeier
Fabio Cuzzolin
Warsaw University, Poland
Pierre et Marie Curie University, France
Ghent University, Belgium
Chung-Ang University, Korea
Academia Sinica, Taiwan
University of Granada, Spain
Ghent University, Belgium
Philipps-Universität Marburg, Germany
Oxford Brookes University, UK

VIII Organization

Gabriele Kern-Isbner	University of Dortmund, Germany
Gary Gang Feng	Hong Kong City University, Hong Kong
Haibin Duan	Beihang University, China
Hiroaki Kikuchi	Tokai University, Japan
Huaining Wu	Beihang University, China
Hung T. Nguyen	New Mexico State University, USA
Jonathan Rossiter	University of Bristol, UK
Jordi Recasens	UPC Barcelona, Spain
Jun Liu	Jordanstown Campus, University of Ulster, UK
Junzo Watada	Waseda University, Japan
Kaizhu Huang	Chinese Academy of Sciences, China
Kaoru Hirota	Tokyo Institute of Technology, Japan
Laszlo Koczy	Budapest University of Technology and Economics, Hungary
Lluis Godo	Artificial Intelligence Research Institute, CSIC, Spain
Masahiro Inuiguchi	Osaka University, Japan
Milan Vlach	Charles University, Czech Republic
Ming Li	Nanjing University, China
Mitsuru Ikeda	Japan Advanced Institute of Science and Technology, Japan
Noboru Takagi	Toyama Prefectural University, Japan
Radko Mesiar	Slovak University of Technology Bratislava, Slovakia
Reynold Cheng	University of Hong Kong, Hong Kong
Ronald Yager	Machine Intelligence Institute, USA
Sadaaki Miyamoto	University of Tsukuba, Japan
Shiying Qin	Beihang University, China
Tao Wan	Case Western Reserve University, USA
Tetsuya Murai	University of Hokkaido, Japan
Thierry Denoeux	University of Technology of Compiègne, France
Trevor Martin	University of Bristol, UK
Tru H. Cao	Ho Chi Minh City University of Technology, Vietnam
Tu Bao Ho	Japan Advanced Institute of Science and Technology, Japan
Van Hung Dang	Hanoi National University, Vietnam; UNU-IIST, Macau
Vicenc Torra	Artificial Intelligence Research Institute, CSIC, Spain
Vladik Kreinovich	University of Texas at El Paso, USA
Wei Huang	Huazhong University of Science and Technology, China

Weiru Liu
Witold Pedrycz
Yaxin Bi
Yongyong He
Yu-Wang Chen
Zengchang Qin

Queen's University Belfast, UK
University of Alberta, Canada
University of Ulster at Jordanstown, UK
Tsinghua University, China
University of Manchester, UK
Beihang University, China

Table of Contents

Robot Reasoning Using First Order Bayesian Networks	1
<i>Saleha Raza, Sajjad Haider, and Mary-Anne Williams</i>	
Dempster-Shafer Theory with Smoothness	13
<i>Ronald Böck, Stefan Glüge, and Andreas Wendemuth</i>	
A Probabilistic Model for Prioritizing Engineering Design Requirements in Uncertain QFD	23
<i>Hong-Bin Yan and Tieju Ma</i>	
Applying OWA Operator to Model Group Behaviors in Uncertain QFD	37
<i>Hong-Bin Yan, Tieju Ma, and Van-Nam Huynh</i>	
New Propagation Algorithm in Dynamic Directed Evidential Networks with Conditional Belief Functions	50
<i>Wafa Laâmari, Boutheina Ben Yaghlane, and Christophe Simon</i>	
Evidential Grammars for Image Interpretation – Application to Multimodal Traffic Scene Understanding	65
<i>Jean-Baptiste Bordes, Franck Davoine, Philippe Xu, and Thierry Denœux</i>	
Generic Discounting Evaluation Approach for Urban Image Classification	79
<i>Ahmed Samet, Imen Hammami, Eric Lefèvre, and Atef Hamouda</i>	
Spatial Weighting for Bag-of-Features Based Image Retrieval	91
<i>Chuanqian Wang, Baochang Zhang, Zengchang Qin, and Junyi Xiong</i>	
A Random Set and Rule-Based Regression Model Incorporating Image Labels	101
<i>Guanyi Li and Jonathan Lawry</i>	
Mining Statistically Significant Target mRNA Association Rules Based on microRNA	114
<i>Feng Chen, Tingting Wang, Yu Wang, Shengfei Li, and Jing Wang</i>	
Interval Algorithm for Set-Membership Identification of MIMO LTI System	126
<i>Xiaojun Wang, Chen Yang, Baochang Zhang, and Lei Wang</i>	

An Approach for Obtaining Bargaining Set in Multi-choice Games	137
<i>Tianwen Li, Feng Ma, and Weiyi Liu</i>	
An Optimized Method for Selection of the Initial Centers of K-Means Clustering	149
<i>YongJun Zhang and Enxiu Cheng</i>	
Improving Short Text Classification Using Public Search Engines	157
<i>Wang Meng, Lin Lanfen, Wang Jing, Yu Penghua, Liu Jiaolong, and Xie Fei</i>	
Using Social Networks for Integrating a Tacit Knowledge Repository to Support the Performance of Software Development Teams	167
<i>Reyes Juárez-Ramírez, Rafael Pimienta-Romo, and Violeta Ocegueda-Miramontes</i>	
Tacit Knowledge Formalization to Support the Adoption Process of Software Quality Models	180
<i>Violeta Ocegueda-Miramontes and Reyes Juárez-Ramírez</i>	
Design and Implementation of an Office Automation Assistant Utility Using Adobe AIR Technology	192
<i>Xin Liu, Liang Liao, Bin Yang, and Heng Liu</i>	
Modified NFTSM on State in Switched Reluctance Motor Speed Controller	202
<i>Yong Cheng</i>	
Uncertain Nonlinear System Modeling and Identification Using Belief Rule-Based Systems	209
<i>Yu-Wang Chen, Jian-Bo Yang, and Dong-Ling Xu</i>	
Author Index	219

Robot Reasoning Using First Order Bayesian Networks

Saleha Raza¹, Sajjad Haider¹, and Mary-Anne Williams²

¹ Artificial Intelligence Lab, Institute of Business Administration, Karachi, Pakistan
`{saleha.raza,sajjad.haider}@khi.iba.edu.pk`

² Innovation and Enterprise Research Laboratory, University of Technology, Sydney, Australia
`mary-anne.williams@uts.edu.au`

Abstract. This study presents the application of first-order Bayesian Networks (FOBN) to model and reason in domains with complex relational and rich probabilistic structures. The FOBN framework used in this study is ‘multi-entity Bayesian networks (MEBN). MEBN has its roots in Bayesian networks and aims to overcome some key modeling limitations of Bayesian networks by supplementing them with the expressive power of first-order logic. The study has been conducted in the domain of RoboCup Soccer which provides a challenging benchmark platform to evaluate the applicability of any knowledge representation mechanism. The benchmark scenario in this paper comprises of a soccer playing agent who is in possession of the ball and needs to decide the best action to be performed in a specific game situation. Further intricacies of this scenario have been discussed to thoroughly assess the effectiveness of first-order Bayesian network in the domain of RoboCup Soccer and it is found to provide the essential expressive power required to facilitate decision-making in such complex, stochastic domains.

Keywords: RoboCup Soccer, multi-entity Bayesian network (MEBN), first-order Bayesian network, Statistical Relational Learning, first-order logic, Bayesian networks.

1 Introduction

Intelligent decision making in real-world situations requires powerful mechanisms that can model complex situations and support reasoning in the presence of incomplete information. Logic provides a way of reasoning and first-order logic (FOL) enjoys considerable power of expression necessary to model complex real-world situations. However, logics alone are deterministic in nature and that limits their applicability in uncertain environments. Uncertainty, an inherent characteristic of real-life situations, can be captured using probabilistic approaches such as Bayesian networks[1] and as a result Bayesian Networks have become a standard mechanism to model uncertainty in real-world applications. A major drawback, however, is that Bayesian networks are propositional in nature and lack the power of expression required for representing knowledge in complex dynamic real-world.

Several efforts have been made to address the weaknesses logic possesses in modeling uncertainty e.g. possibility logic[2], fuzzy logic[3], and statistical relational

learning [4]etc. Statistical Relational Learning (SRL) is an umbrella term for techniques that use logic-oriented techniques for representing relational aspects of a domain and probabilistic methods to model their uncertainty. SRL can be used to model, reason and learn in domains with complex relational and rich probabilistic structures. A range of SRL approaches have emerged from a variety of subfields and hence they have a wide variety of representations, inference mechanisms, and machine learning frameworks. For instance, approaches like DAPER[4], Plate[5] and probabilistic relational models[5] have their roots in relational models which are augmented with probabilistic reasoning. In contradistinction, tools like BLOG[6], PRISM[7] , Bayesian logic programs (BLP)[8], and IBAL[4] extend logic based representations and mechanisms to handle uncertainty. Certain other formalisms like Markov logic networks [9], multi-entity Bayesian network[10] and relational dynamic Bayesian networks[11] have their origin in probabilistic graphical models. Getoor *et al*[4] provides a comprehensive overview of important SRL approaches. A comparative study of three different SRL approaches (BLP, BLOG, and MEBN) is presented in[12] while a detailed survey of a range of different first-order probabilistic languages is provided in [13].

This paper evaluates one such promising framework, the multi-entity Bayesian network (MEBN)[10], [14], [15], which is a powerful hybrid of first-order logic and Bayesian networks. The test bed selected as the case study for this evaluation is the RoboCup Soccer environment. RoboCup Soccer[16] is one of the most exciting competitions within the field of Artificial Intelligence and cognitive robotics in which teams of autonomous mobile robots play soccer. This paper uses the RoboCuo Soccer Simulation 3D league as the benchmark for evaluation. The RoboCup Soccer Simulation 3D platform provides a dynamic and uncertain environment that is partially observable to the player. As a team-based game, soccer players will, undoubtedly, be more effective if they take coordinated decisions and undertake collective actions to counter the adversarial moves of the opponent team.

The decision making in such a stochastic, incomplete and adversarial environment requires powerful mechanisms to model perceived game situations and to reason about them. Individually, Neither first-order logic nor Bayesian networks can effectively deal with all of the intricacies offered by such domains. However, we show that their synergized form, first-order Bayesian networks, offers enormous promise. The main purpose of this paper is to explore and evaluate the applicability of the MEBN approach to perform reasoning in robot applications, to demonstrate how it can be used, to identify its advantages and shortcomings, and to help accelerate its adoption in the field. This is the first work, to the best of our knowledge, which presents the effectiveness of first-order Bayesian networks in RoboCup Soccer environment. So far, none of the teams in RoboCup Soccer Standard Platform league or Simulation league have applied this mechanism to perform reasoning. It is, therefore, our belief that the work suggests a new modeling paradigm for RoboCup Soccer and establishes a theoretical foundation for future robot reasoning.

The rest of the paper is organized as follows: Section 2 presents main building blocks of MEBN framework, while Section 3 describes their applicability as a

reasoning mechanism of autonomous mobile robots in a dynamic and uncertain domain. Finally, Section 4 provides a conclusion of the whole case study.

2 Multi-Entity Bayesian Networks

Multi-Entity Bayesian network is an emerging framework presented by Laskey[10] as a logical system that unifies Bayesian probability and statistics with classical first-order logic. MEBN provides reasoning and learning mechanism that enables predicate-based representation to Bayesian networks. Each MEBN model, called an *MTheory*, represents a particular domain of discourse. Different subjects/topics of the domain are represented by smaller components known as MEBN fragments (*MFrag*). Similar to a class in an object-oriented model, MFraggs group attributes pertinent to the respective subject. Each MFrag is a Directed Acyclic Graph (DAG) with parameterized nodes that represent attributes of entity and edges that represent dependencies amongst them. MFrag nodes can be of three types, context nodes, resident nodes and input nodes. *Context Nodes* are evaluated to either True/False when substituted with constant values in place of parameters. *Resident Nodes* are local nodes of MFrag that represent random variables with probability distributions. *Input Nodes* serve as input to derive probability distributions of resident nodes. They are resident nodes of other MFraggs where their own probability distribution is defined. UnBBayes[17] is a tool that supports implementation of MEBN models. Once a model has been developed, a query specifying a target node can be executed generating a *situation specific Bayesian network (SSBN)*. SSBNs depict a particular occurrence of a problem.

3 Using MEBN in an Autonomous Robot Benchmark Problem

We use RoboCup 3D simulation league challenge problem as it provides a well understood, yet complex and dynamic environment where simulated robots must respond quickly and strategically to game situations and changes. Deciding the best action by a robot soccer agent during matches typically takes many variables that describe the game situation into account. This section describes and analyses different scenarios of a RoboCup Soccer match and identifies several key modeling limitations of Bayesian networks. Furthermore, it demonstrates how MEBNs overcome these limitations.

3.1 MFraggs for Multiple Instances and Modularity

During a soccer match, if a robot agent is in possession of the ball then it may consider passing the ball to a fellow player or dribbling/kicking it in a particular direction. The decision itself is based on various factors. For example, if the nearest opponent is too close to the ball and a teammate is around, passing to the teammate may be a good option; but if the ball is likely to be intercepted during the pass due to

the positioning of an opponent player, dribbling the ball in some other direction might be a better option. The positions of team members and opponents in the field should influence the player's action about the move. For instance, dribbling might be a reasonable option if just a single opponent player is positioned to intercept the ball but in the presence of two or more opponents in intercepting positions decreases the chances of successful dribble.

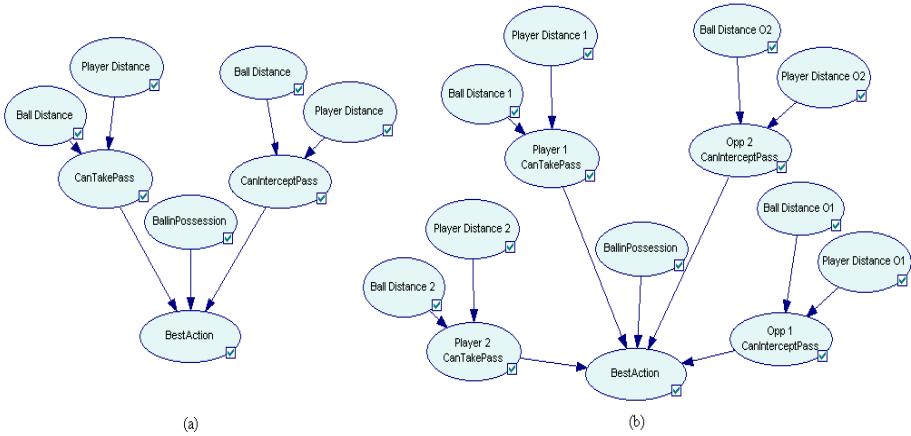


Fig. 1. Typical Bayesian Networks to handle different situations in a soccer match

Fig. 1(a) represents the structure of a simple Bayesian network that can be used to determine the best action in the presence of one teammate and one opponent near the player in possession of the ball. For the sake of simplicity, we have considered ‘Distance’ as the only criteria to assess the feasibility of successful passing and interception. Though, in reality, there are typically other variables involved such as orientation of the players, their velocity and predicted future positions etc.

A major shortcoming with the Bayesian network shown in Fig. 1(a) is that it will not serve if there is a small change to the scenario. For instance, the model shown in Fig 1(a) will not cater to a situation in which there are two opponents and two teammates around, and a revised model like that illustrated in Fig. 1(b) is needed to handle that elaboration. As situations in a soccer match continually change, the number of nearby teammates and opponents varies and each possible configuration of teammates and opponents typically would require a large number of different Bayesian network structures. From a domain modeling perspective, this is a significant problem as there is no single Bayesian model that could serve in all the important and relevant situations. Hence a more generic and flexible representation that allows instantiation of nodes as per the situation is desirable.

Fig. 1 highlights another major shortcoming of Bayesian networks, namely, the lack of modularity. Certain nodes (like BallDistance, PlayerDistance, CanTakePass, CanInterceptPass) are repeatedly drawn for each opponent/teammate despite the fact that they convey the same semantics for different players and importantly possess

same underlying probability distribution. This poses enormous difficulties in handling real-life domains that often contain many repetitive elements that should be handled in a modular way to ease the redundant modeling and management effort of a knowledge engineer and the system maintenance burden.

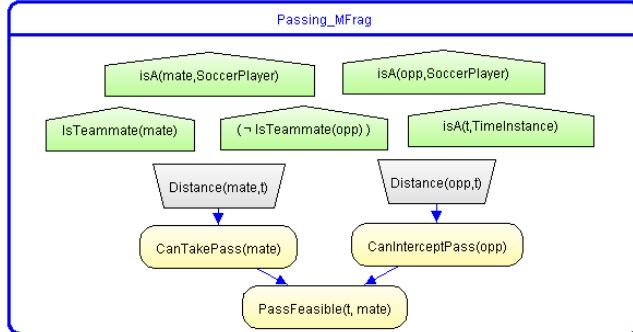


Fig. 2. MFragment for passing behavior of a soccer agent

Both these problems can be effectively handled in the MEBN framework using a theory of MFrags. An MFragment is a parameterized fragment of a MEBN model that groups interrelated random variables which collectively represent a single concept. Similar to the concept of a *class* in an object-oriented model, an MFragment provides a blue print for the underlying concept and can be instantiated multiple times to represent multiple instances. Each instance of an MFragment possesses different values of MFragment parameters. These parameterized fragments enable the instantiation of variable number of entities as per a game situation. Additionally, they provide modularity in a manner that all the related random variables can be grouped at one place, within the MFragment, and can be instantiated as many times as required on the fly by the underlying situation of the game. The nodes in the MFragment can further be used as input to other MFrags and hence connect multiple MFrags together to form a complete model called MTheory[18]. Fig. 2 represents an MFragment for the situation mentioned in Fig. 1. The Yellow rectangles are resident nodes which are equivalent to the random variables in Bayesian networks and their probability distributions are defined in the same MFragment, while the gray trapezoids are resident nodes defined in some other MFragment to maintain modularity and are used here as input nodes to connect two MFrags. All these nodes have certain variables, like ‘mate’, ‘opp’ and ‘t’, that are passed to them as parameters. This parameterized representation allows each node to be instantiated multiple times with different values assigned to the corresponding variables.

3.2 Modeling Consistency through Context Nodes

The green pentagons in Fig. 2 are context nodes which are boolean in nature and must be evaluated to *True* in order for the given MFragment to be instantiated. The probability distribution exhibited by the resident nodes can only be applied if they satisfy all the

context nodes in the MFrag. For example, in Fig. 2, *IsTeammate(mate)* must be *True* to assess the probability of passing to ‘mate’ (*CanTakePass*). Similarly ‘*~IsTeammate(opp)*’ must be *True* to see the chances of the ball being intercepted by ‘opp’ (*CanInterceptPass*). Context nodes, thus, make sure that the correct type of parameters are used to instantiate the MFrag so that the overall model remains semantically correct and uses the full power of the first-order logic representation.

The model shown in Fig. 2 serves as a template and is able to generate both the models shown in Fig. 1(a) and 1(b) and all variations that can result with varying number of players around.

3.3 Flexible Local Probability Distributions

In standard Bayesian networks, a static conditional probability table (CPT) is defined for each node that specifies conditional probabilities of the node given its parents. Since MEBN supports multiple instances of nodes, the exact number of parents is not known at design time and hence a static CPT cannot be specified. Additionally, specifying static CPTs poses a barrier in knowledge acquisition of BN as with an increasing number of parents this becomes a cumbersome and non-intuitive process to specify conditional probabilities for each combination of parent states. MEBNs simplify this problem by providing a generalized language interface to define rules to dynamically compute probabilities based on the state of the node and its parents at any given time. Hence a knowledge engineer is not required to enumerate all combinations of parent states to specify probability distributions. A generalized set of rules, instead, can be defined to compute probabilities in different conditions. For instance, in a given game situation, MEBN allows the knowledge engineer to specify a rule stating that

“if the ball is in the possession of a soccer agent and the number of nearby opponents exceeds one then the probability of giving a pass to a nearby teammate is higher than dribbling the ball.”

In such cases, the ‘if any’ and ‘if all’ clauses of MEBN allows the use of existential and universal quantifiers. Furthermore, a successful pass is a stochastic process and its success may depend upon the number of opponents around. The ‘cardinality’ clause, also known as influence count in MEBN terminology, allows one to compute probabilities based on the exact number of parents instantiated at that time.

Table 1 provides MEBN syntax to specify sample probability distribution for the action (Dribble, Pass, Shoot) to be taken by a soccer player who is in possession of the ball. The first part of this probability distribution states that:

“If the ball is in opponent region surrounded by one or more opponent players, then giving the pass to a teammate (if possible) is the most favorable option (with probability 0.7). In the absence of such nearby teammates, ‘Pass’ is not a viable option and the player will decide either to dribble or shoot the ball depending upon the number of opponent players around.”

Table 1. Local Probability Distribution in MEBN

```

if any g have (BallInRegion=OpponentRegion) [
    if any opp have ( Distance=Near ) [
        if any mate have (PassFeasible=true)
            [Dribble = max(0; 0.3 - ( 0.1*cardinality(opp))),  

             Pass = 0.7,  

             Shoot = min(0.3; 0.1*cardinality(opp))]  

        else
            [Dribble = max(0; 0.95 - ( 0.2*cardinality(opp))),  

             Pass = 0.05,  

             Shoot = min(0.95; 0.2*cardinality(opp))]  

    else
        [ Dribble = 0.7,  Pass = 0.05,  Shoot = 0.25]
]
else[   if any opp have ( Distance=Near )
        [ Dribble = 0.05, Pass = 0.25, Shoot = 0.7]
    Else
        [Dribble = 0.8,Pass = 0.1,Shoot =0.1]]
]

```

3.4 Handling Unknown and Absurd Data

Most of the situations in real-world are partially observable and hence the agent acting in such situations has incomplete information. For instance, a player in the soccer field does not have a global vision of the environment and hence the game state they perceive is partially observable. However, not knowing the position of other players does not necessarily imply that they are not close to the ball. They can still be close to the ball and that may be unknown to the agent, so they may be positioned to affect an intercept. In such cases, the decision making should consider the possibility of players that are not observable but can still have an impact on a decisions outcome. Hence, depending upon the team strategy and the role of each player, an agent could use a default likelihood that a teammate can be at a certain position during the game. This can define the default position of the player with respect to the ball and the knowledge of this default position can be used by fellow players to take certain actions such as undertaking a backward-pass. MEBN allows defining default probability distributions for random variables. This default distribution is used when no specific information regarding the existence of the node is available. Similarly, if a node is Boolean in nature, MEBN permits it to contain the states of *True*, *False* and *Absurd*. The ‘*Absurd*’ state represents an ignorance state in which no explicit information is available regarding the node being *True* or *False*. Thus, if a teammate is not in the visual range of the player and no conclusive evidence regarding its closeness to the ball is available then the ‘*Absurd*’ probability of the node will be high to represent this state of ignorance or indecisiveness.

3.5 Recursive MFraggs

Recursion is an important aspect of any computational modeling paradigm. Bayesian networks lack the capability of defining recursive dependencies, although certain types of recursion are possible in dynamic Bayesian networks with a fixed number of time slices. Recursive situations can be used to conceptualize many real-world scenarios. For instance, in the current example, if a team-mate was previously seen close to the ball but is now no longer in the visual range of a soccer agent, then belief regarding its distances could persist or degrade at a measured rate for some appropriate time. This mimics the behavior of memory in people. Hence if a player is not seen at time $t+1$, its probability of being at a certain position at $t+1$ can be inferred from its position at time t which in turn can be derived from its position at earlier time instances. This belief should not remain constant forever, rather it should decrease as the time passes by and after some time the agent will no longer believe the fellow player is located at the position last seen. This is a type of recursive relationship with a varying number of time slices. Such relationships are prevalent in many real-life scenarios and certainly in human soccer where the number of time slices/the degree of recursion varies.

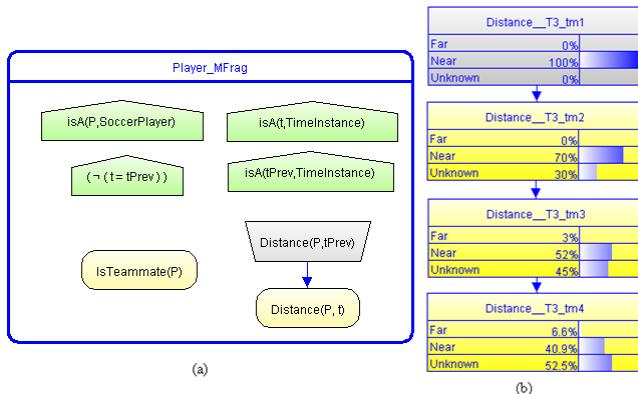
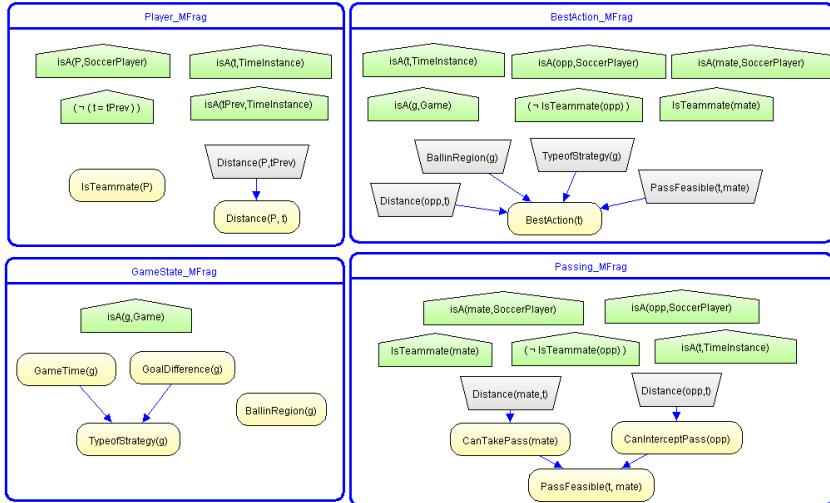


Fig. 3. Recursive relationship in BN

MEBN provides a generic representation of such recursive relationships that supports n^{th} level recursion. Fig. 3(a) illustrates a recursive MFragment in which the ‘Distance’ of a player at time ‘ t_{Prev} ’ derives its distance at time ‘ t ’ with the decreasing degree of belief. The instantiated form of this MFragment can be seen in Fig 3(b) for four time slices.

3.6 Generating Situation Specific Bayesian Network

A complete MTheory represents an infinite number of possibilities. The MTheory shown in Fig. 4 can represent a variety of situations during a RoboCup soccer match with a varying number of opponents and teammates surrounding the ball and seen at

**Fig. 4.** MTheory for RoboCup Soccer scenario

different time steps. The particular instance of the problem is created by adding *findings* to the MTheory. Findings are similar to facts in first-order logic and provide situation specific information to the model. Once instantiated with findings, the model can now be queried to find updated beliefs of different random variables. Instantiating and querying a model result in the generation of a *situation specific Bayesian network* (SSBN). SSBN is a Bayesian network that includes only those nodes from MTheory that are essential to answer a given query. Standard Bayesian inferences are performed in this SSBN to answer the query made by the user.

Table 2. Findings and Query in sample MEBN**Findings:**

$\text{GameTime}(g1 \ (\text{Game})) = \text{firstHalf}$
 $\text{GoalDifference}(g1 \ (\text{Game})) = \text{zero}$
 $\text{BallinRegion}(g1 \ (\text{Game})) = \text{opponentRegion}$
 $\text{isTeammate}(t1 \ (\text{SoccerPlayer})) = \text{true}$
 $\text{isTeammate}(t2 \ (\text{SoccerPlayer})) = \text{true}$
 $\text{isTeammate}(t3 \ (\text{SoccerPlayer})) = \text{true}$
 $\text{isTeammate}(o1 \ (\text{SoccerPlayer})) = \text{false}$
 $\text{isTeammate}(o2 \ (\text{SoccerPlayer})) = \text{false}$
 $\text{isTeammate}(o3 \ (\text{SoccerPlayer})) = \text{false}$
 $\text{Distance}(t1 \ (\text{SoccerPlayer}), tm0 \ (\text{TimeInstance})) = \text{Far}$
 $\text{Distance}(t2 \ (\text{SoccerPlayer}), tm0 \ (\text{TimeInstance})) = \text{Far}$
 $\text{Distance}(t3 \ (\text{SoccerPlayer}), tm0 \ (\text{TimeInstance})) = \text{Near}$
 $\text{Distance}(o1 \ (\text{SoccerPlayer}), tm0 \ (\text{TimeInstance})) = \text{Near}$
 $\text{Distance}(o2 \ (\text{SoccerPlayer}), tm0 \ (\text{TimeInstance})) = \text{Far}$

Query: `BestAction(t0)`

Table 2 shows the set of findings and the corresponding query provided to the model. The given set of findings represents following game situation:

“The first half of the game is going on and the goal difference is 0. The ball is in opponent region. There are 3 teammates (t_1, t_2, t_3) and 3 opponents (o_1, o_2, o_3). Players t_3 and o_2 are ‘Near’ to the player possessing the ball while t_1 , t_2 and o_1 are ‘far’. No explicit information is available regarding the position of o_3 . The model has been asked to suggest the best action for the player “

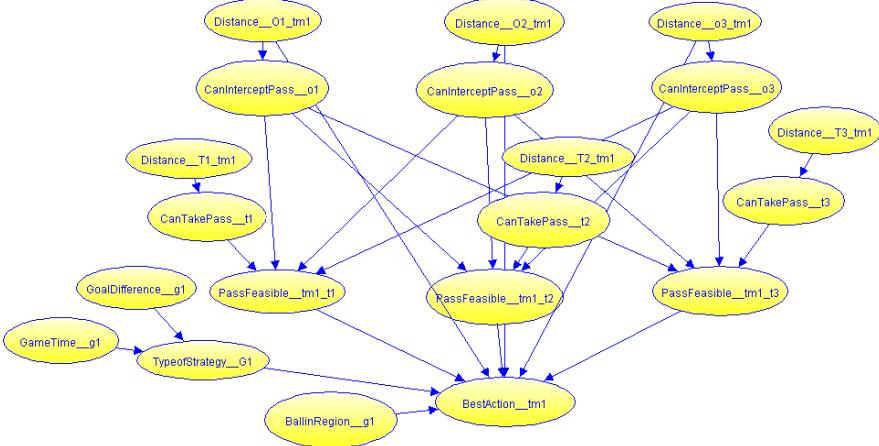


Fig. 5. SSBN for the given set of findings

Fig. 5 shows the resulting SSBN generated by MEBN. The same MTheory shown in Fig. 4 can be used to generate a smaller SSBN for another game situation in which there is one teammate and two opponents around. The resultant SSBN is shown in Fig. 6. The situation presented in Fig. 6 can be complicated if the underlying game situation considers multiple time slices ($tm_1 - tm_3$) in which teammate t_1 and opponents o_1 were seen in the time slice tm_1 but no explicit information about them is available from the vision in the current time slice ‘ tm_3 ’. In this case, the SSBN shown in Fig. 7 will use information available in previous time slices to infer the position of the player in the current time slice. All these models are generated on the fly using MTheory and facilitate decision making under uncertainty with varying number of instances.

4 Conclusion

This study has provided a theoretical foundation for applying first-order Bayesian networks (MEBN) in robot applications using the RoboCup Soccer challenge as the benchmark problem. RoboCup Soccer is a complex team-based multi-agent environment with agents collaborating with teammates in competition with an

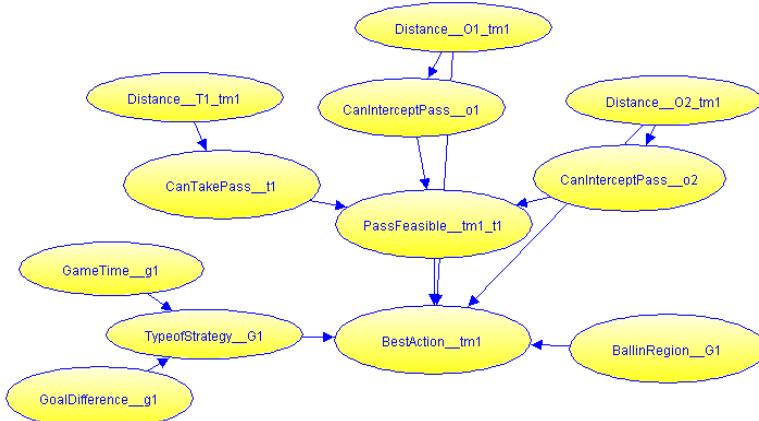


Fig. 6. SSBN for one teammate and two opponents

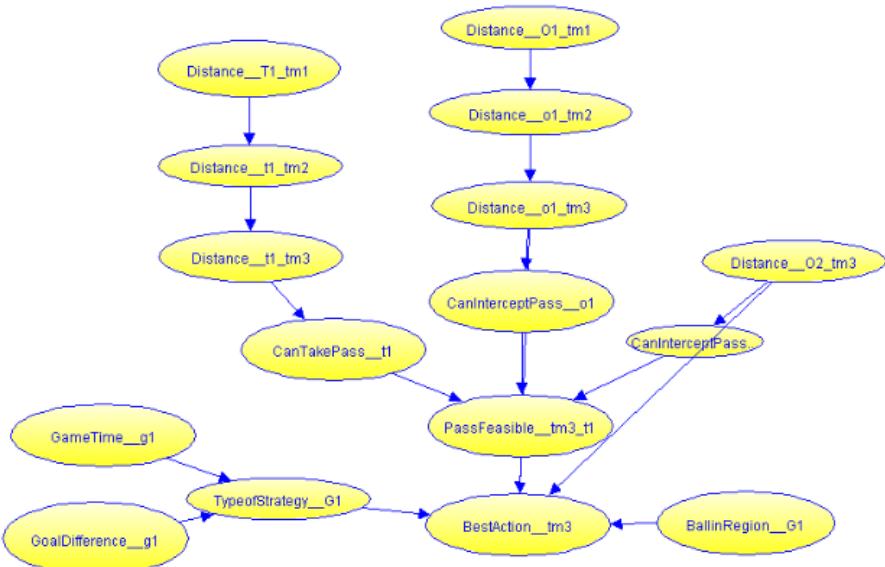


Fig. 7. SSBN with three times slices

opponent team. In this challenging environment, FOBNs have found to be successful in countering the limitations of Bayesian networks and providing sound foundation for modeling, reasoning and learning. With the help of benchmark scenarios, the study has demonstrated that MEBN, a framework of FOBN, can be used to create multiple instances of reasoning scenarios dynamically and on the fly using the available information. MEBN further facilitates defining complex and recursive probabilistic relationships and offer flexible mechanisms for defining probability

distributions based on quantifiers and cardinality of parents. The study has also shown the effectiveness of MEBN to deal with incomplete and unknown information available to soccer agents in robot soccer environment. Although, the study has been conducted in the domain of RoboCup Soccer, its findings can be extended to suitable reasoning mechanisms for autonomous agents achieving goals in any partially known complex dynamic environment.

Acknowledgement. The work was partially supported by Higher Education Commission, Pakistan research grant 20-1815/RCD/105032.

References

- [1] Jensen, F.V.: Bayesian Networks and Decision Graphs. Corrected. Springer (2002)
- [2] Dubois, D., Prade, H.: Possibility Theory, Probability Theory and Multiple-Valued Logics: A Clarification. *Annals of Mathematics and Artificial Intelligence* 32(1-4), 35–66 (2001)
- [3] Gerla, G.: Fuzzy Logic: Mathematical Tools for Approximate Reasoning, 1st edn. Softcover reprint of hardcover. Springer (2010)
- [4] Getoor, L., Taskar, B. (eds.): Introduction to Statistical Relational Learning. The MIT Press (2007)
- [5] Heckerman, D., Meek, C., Koller, D.: 7 Probabilistic Entity-Relationship Models, PRMs, and Plate
- [6] Milch, B., Marthi, B., Russell, S., Sontag, D., Ong, D.L., Kolobov, A.: Blog: Probabilistic models with unknown objects. In: IJCAI, pp. 1352–1359 (2005)
- [7] Sato, T.: A glimpse of symbolic-statistical modeling by PRISM. *J. Intell. Inf. Syst.* 31(2), 161–176 (2008)
- [8] Kersting, K., De Raedt, L.: Bayesian Logic Programs. Albert-Ludwigs University at Freiburg (2001)
- [9] Richardson, M., Domingos, P.: Markov logic networks. *Mach. Learn.* 62(1-2), 107–136 (2006)
- [10] Laskey, K.B.: MEBN: A Logic for Open-World Probabilistic Reasoning. Research Paper, vol. 2005 (2004)
- [11] Manfredotti, C.: Modeling and Inference with Relational Dynamic Bayesian Networks. In: Gao, Y., Japkowicz, N. (eds.) AI 2009. LNCS, vol. 5549, pp. 287–290. Springer, Heidelberg (2009)
- [12] Raza, S., Haider, S.: Modeling First-Order Bayesian Networks (FOBN), pp. V2–571–V2–575 (2010)
- [13] de Salvo Braz, R., Amir, E., Roth, D.: A Survey of First-Order Probabilistic Models (2008)
- [14] da Costa, P.C.G., Laskey, K.B.: Multi-Entity Bayesian Networks Without Multi-Tears
- [15] Costa, P.: Bayesian Semantics for the Semantic Web. School of Information Technology and Engineering, George Mason University (2005)
- [16] RoboCup (February 13, 2012), <http://www.robocup.org/> (accessed: February 13, 2012)
- [17] Costa, P.C.G., Ladeira, M., Carvalho, R.N., Laskey, K.B., Santos, L.L., Matsumoto, S.: A first-order bayesian tool for probabilistic ontologies. In: Proceedings of the 21st Florida AI Research Symposium, FLAIRS (2008)

Dempster-Shafer Theory with Smoothness

Ronald Böck, Stefan Glüge, and Andreas Wendemuth

Otto von Guericke University, Cognitive Systems, IESK,
Universitätsplatz 2, 39106 Magdeburg, Germany
ronald.boeck@ovgu.de
<http://www.kognitivesysteme.de>

Abstract. This paper introduces the idea of a modified Dempster-Shafer theory. We adapt the belief characteristic of expert combination by introducing a penalty term which is specific to the investigated object. This approach is motivated by the observation that final decisions in the Dempster-Shafer theory might tend to fluctuations due to variations in sensor inputs on small time scales, even if the real phenomenological characteristic is stable.

1 Introduction

Each sensor signal is afflicted with uncertainty. Hence, a system which utilises such kind of inputs has to handle it in a proper way. This is especially the case in the combination of several inputs, where it does not matter whether the values are given by a real sensor or by human experts. The latter is the case in questionnaires. By observing sensors, measurements are often done on small time intervals, e.g. less than one second. However, in several applications decisions need to be stable for some time. For instance, in measuring the heart rate we have a sample frequency of 25ms. On the other hand, noticeable change in body reaction occurs in the time scale of seconds. Having a system which draws conclusions every 25ms we may end up with unwanted fluctuations in the decisions. With our approach, we aim to smooth such decisions.

For this, we introduce an additional summand to the belief (cf. Sect. 3), which will be called penalty term. This is intended to smooth decisions that tend to fluctuate, but are usually stable in their phenomenological characteristic. This term reflects the characteristics of the measurements as discussed above.

In this paper, we discuss (especially in Sect. 2 and 5) the penalty term on the example of emotions shown by a user, as each interaction and user reaction, especially in a human-machine interaction, is influenced by the user's behaviour and personality. Furthermore, emotions are part of the user's personality and influence not only the process of decision making, but also the interaction itself. This was already shown, e.g. in [1, 2, 11, 12]. However, the framework which we present is general and introduced in a task independent way.

To merge information from several inputs, e.g. sensors, users, or experts, either learning or stochastical approaches are used (cf. [6]). A well known stochastical framework is the Dempster-Shafer Theory (DST) with its core idea of Dempster's Rule of Combination (DRoC). The framework was introduced in

[3, 10]. The authors also gave definitions for the basic functions of this framework: basic probability assignment (bpa), Belief function, and Plausibility function. The interpretations of the terms are as follows: the bpa is a confidence value which is assigned by a user to a statement, expression, or numerical value. The belief represents the accumulated confidence of a user in a statement, whereas the plausibility is the counterpart, i.e. the remaining accumulated confidence of all statements not currently observed. The mathematical definitions, which are introduced in the following, are based on [10].

Definition 1 (Basic Probability Assignment). *The bpa m is the amount of belief which is precisely assigned to $A \subset \mathcal{P}(X) = 2^X$ where X is a set.*

$$\begin{aligned} m : 2^X &\rightarrow [0, 1] \text{ where} \\ m(\emptyset) &= 0 \\ \sum_{A \subset X} m(A) &= 1 \end{aligned}$$

Definition 2 (Belief Function). *This is the accumulated bpa of a statement $B \in X$, i.e. it is the belief that B is in A .*

$$\begin{aligned} Bel : 2^X &\rightarrow [0, 1] \\ \text{for } A \subset \mathcal{P}(X) : Bel(A) &= \sum_{B \subseteq A} m(B) \end{aligned}$$

Definition 3 (Plausibility Function). *Plausibility measures the assignment of the belief. There are two ways to compute the plausibility function.*

$$\begin{aligned} Pl : 2^X &\rightarrow [0, 1] \\ \text{for } A \subset \mathcal{P}(X) : Pl(A) &= 1 - Bel(\overline{A}) \\ \text{or } Pl(\{s \in A\}) &= \sum_{B \cap \{s\} \neq \emptyset} m(B) \end{aligned}$$

Based on these definitions the DRoC can be expressed as follows:

Definition 4 (Dempster's Rule of Combination). *Given two bpas m_1 and m_2 the combinations with normalisation is defined as*

$$(m_1 \oplus m_2)(D) = \frac{\sum_{A \cap B = D \neq \emptyset} m_1(A) m_2(B)}{1 - \sum_{A \cap B = \emptyset} m_1(A) m_2(B)}$$

In this paper we introduce our approach on the basis of the Belief function. For the remaining functions the application of a penalty term is analogous.

In [9] several other combination rules are presented, e.g. Yager's Rule, Dubois-Prade's Rule, etc. In our work, we concentrate on the standard DRoC. Further, in [7] a penalty logic is introduced to selected appropriate formulas in a knowledge base. In addition, the authors show the link between their penalty logic and the DST. In contrast, in our framework the penalty term is directly introduced in the DST. To the best of our knowledge such an idea has not been presented so far.

2 Motivation for an Extension

The standard DST is a quite powerful framework to combine experts' knowledge, which is defined in terms of belief or trust, and statements. Moreover, it is independent of any domain, i.e. the application and its environment do not matter. Hence, it is universal. Nevertheless, to be applicable some characteristics have to be fulfilled:

- The power set $\mathcal{P}(X)$ has to be constructed.
- Bpas are given as numericals, i.e. a kind of transformation from non-numerical expressions to numerical values is necessary.
- Values have to be in the range of $[0, 1]$.

Note that this is not an exhaustive list of conditions. We listed only the most important ones.

In most cases, these characteristics can be tackled. This paper will not deal with those characteristics and no discussion of possible solutions will be given. We focus on another problem: instability in combination decisions. Note that our approach aims on handling the case where decisions have to be stabilised to approximate observed characteristics by having inputs in small measuring intervals.

As we discussed in Sect. 1 we introduce the penalty term on the example of user emotions. In particular, we consider the following example: Three classifiers are given as experts in emotion recognition from speech. They have the same input and produce an assessment as output. To deal with numerical values we assume that the emotions are given in dimensional space like Pleasure-Arousal or Pleasure-Control space, [8] (cf. Fig. 1). In speech processing it is possible to make decisions on word level, i.e. in less than a second. Several classifiers work also on phoneme level (cf. [12]) which is less than a word and hence, less than 0.5 seconds. Such observations can be made as well, having physiological measurements like heart rate, skin conductance level, etc. (cf. [13]). For biological (for low level reactions cf. [5]) as well as for logical reasons, it is not plausible that changes in emotions are in periods of less than a second. Considering heart rates, for instance, values are measured in small time intervals, but physiological reactions in terms of emotions occur on longer ranges. Unfortunately, the combination of several classifiers, especially, if they are not absolute reliable, can cause the effect that by an outlier the final decision is corrupted. Due to the investigated phenomena such an event can happen and so far, in the DST and DRoC it is not modelled in a proper way. Therefore, a kind of smoothness in the decisions or combination is appreciated. By adding a penalty term to the framework of DST such a smooth combination strategy can be pursued.

3 Attaining Stability by a Penalty Term

Before the penalty term will be introduced some notations have to be declared.

The bpa and *Bel* are defined as given in Sect. 1. Further, as we deal with numerical values and corresponding to [10] the declaration is in an n-dimensional

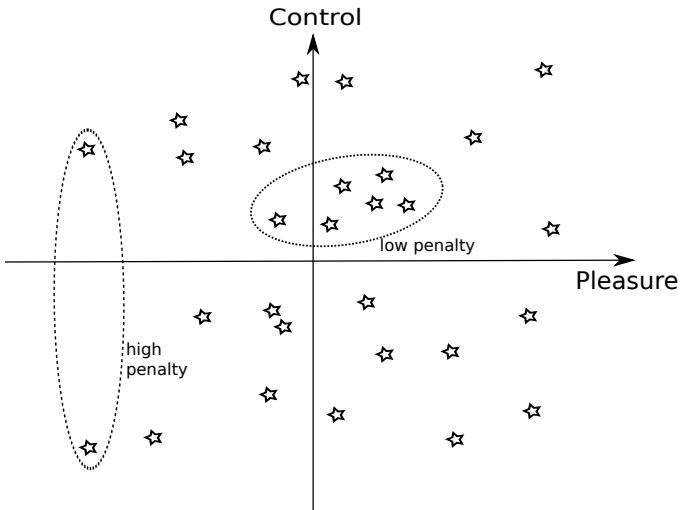


Fig. 1. Samples of discrete emotions in Pleasure-Control space. The two ellipses show examples for low and high penalty values (cf. Sect. 3) depending on the distance.

space, a distance measure has to be defined. This will be used to generate a smoothing of the decisions as part of the so-called penalty or chastisement (C). Without any restriction we do the description of the approach in a two-dimensional space and for the Belief function only. As a distance measure the Euclidian distance $\|\cdot\|_2$ is applied and therefore, we base the penalty on the distance of two statements in the space. For instance, in Fig. 1 the two ellipses represent two cases, where depending on the distance either low or high values of penalties are assigned to the penalty term.

Finally, we distinguish two notations of the symbol x_i (cf. (7)): (i) x_i is the representative of the vector, i.e. a textual description of the value or a phrase, for instance, $x_i = \text{"low satisfaction"}$ or $x_i = \text{"concentrated"}$. (ii) On the other hand, \underline{x}_i is the vector in a multi-dimensional space itself, e.g. $\underline{x}_i = \begin{pmatrix} 2 \\ 1 \end{pmatrix}$. By its design, this vector represents a subset of the underlying set X .

As already mentioned \underline{x}_i is a vector in a multi-dimensional space where a mapping from the representative x_i onto \underline{x}_i is applied. The mapping is not considered in this paper since it does not influence the method as such; that means the smoothing is independent from the mapping function. In general, the mapping is highly related to the application underlying the combination task. Therefore, each mapping function has to be defined by the design appropriate to the specific situation. It is obvious that hence, the dimension of the space is varying as well.

Having these definitions and descriptions we can define the penalty term C as follows:

$$C(\underline{x}_i, \underline{x}_j, A) \quad (1)$$

where \underline{x}_i is the current point in the n-dimensional space, \underline{x}_j is the previous point, and $A \subset \mathcal{P}(X)$. In this case, \underline{x}_j represents a kind of history of decisions already made by the expert on successive time steps. If the decision is stable, the penalty value is zero, otherwise depending on the distance a penalty is assigned to the decision to avoid unwanted fluctuations.

Why do we need this? As discussed in Sect. 2 long-term evolution of events (e.g. emotions, moods) can be based on short-term observations but need a kind of stability in the decisions. Therefore, we have to consider the previous judgements of experts. This can be done by propagating (at least) the predecessor to the current combination.

According to (1) and Def. 2 the Belief function can be extended:

$$Bel(A | \underline{x}_j) = \sum_{B \subseteq A} m(B) - C(\underline{x}_i, \underline{x}_j, A). \quad (2)$$

For the Plausibility function and the Combination Rule the extensions are analogous.

In this framework, we compute the actual penalty based on the Euclidian distance. The value of it is given by

$$c(\underline{x}_i, \underline{x}_j) = \|\underline{x}_i - \underline{x}_j\|_2. \quad (3)$$

4 Problem of Scaling

Using the introduced method we modify the characteristic of Bel , which is $Bel(A) \in [0, 1]$, because by subtracting C , Bel can be negative and we end up with a value that can be outside this interval. Hence, scaling the result is necessary.

As we are in an n-dimensional space and apply the Euclidian distance, scaling can be reduced to the computation of an appropriate scaling factor. Again, with the Euclidian distance such a factor is more or less given by definition. Let c_{\max} be the maximum value of the pairwise Euclidian distance

$$c_{\max} = \max_{(i,j)} \|\underline{x}_i - \underline{x}_j\|_2 \quad \forall i, j, \quad i > j, \quad (4)$$

then the scaled penalty \bar{c} is defined as

$$\bar{c}(\underline{x}_i, \underline{x}_j) = \frac{\|\underline{x}_i - \underline{x}_j\|_2}{c_{\max}}. \quad (5)$$

As it is well known (cf. Def. 2), the Bel value of an expression, e.g. $\{ab\} \in \mathcal{P}(X)$, is composed from single statements by summation over the bpas, i.e. $Bel(\{ab\}) = m(\{a\}) + m(\{b\}) + m(\{ab\})$. This also holds for the penalty value. Hence, for complex expressions the scaled average can be defined as

$$\bar{c}^*(\underline{x}_i, \underline{x}_j, A) = \sum_{x_i=B \subseteq A} \bar{c}(\underline{x}_i, \underline{x}_j) = \sum_{x_i=B \subseteq A} \frac{\|\underline{x}_i - \underline{x}_j\|_2}{c_{\max}}, \quad (6)$$

where \underline{x}_i is the currently observed vector and \underline{x}_j is the vector representing the history in the decisions. Therefore, \underline{x}_j is a kind of a target for the smoothing which is intended by the penalty term. Finally, A is the subset of X including at least x_i .

In particular cases such a way of scaling is convincing, i.e. we can assure that the calculated values are in the interval of $[0, 1]$. In general, it can happen that (2) becomes negative if $\bar{c}^*(\underline{x}_i, \underline{x}_j, A)$ is greater than $\sum_{B \subseteq A} m(B)$. To avoid this we have to improve the scaling method. Our approach has in some parts reference to work of Lianwen Zhang (cf. [14]), but also major differences as we are handling a kind of cardinality. We introduce the number of summands, which are part of the complex statement, $/A/$ as an additional parameter for the scaling, e.g. according to the previous example $|\{ab\}| = 3$. The way of computing $/A/$ is given in (8). Thus, in extension of (6) scaled penalty \tilde{c} can be written as

$$\tilde{c}(\underline{x}_i, \underline{x}_j, A) = \frac{1}{/A/} \sum_{x_i = B \subseteq A} \bar{c}(\underline{x}_i, \underline{x}_j) = \frac{1}{/A/} \sum_{B \subseteq A} \bar{c}(B, \underline{x}_j). \quad (7)$$

In this context the question arises how to compute the cardinality of A in an efficient way. This is related to efficient algorithms to compute functions in DST (cf. [4]). The notational schema is as follows: repeat the following two steps for all single statements in X :

1. list a new single statement,
2. copy the previous list (preserving the order) and append the new single statement to each term in the copied list.

Hence, the set has the following format (the terms are examples only)

$$X = \{\emptyset \ a \ b \ ab \ c \ ac \ bc \ abc \ \dots\}$$

with the index $I_k \in \{0, 1, 2, 3, \dots\}$ of the k^{th} element. Assuming this schema the cardinality of a subset A of X can be calculated as

$$/A/ = 2^\iota - 1 \quad (8)$$

where ι is the number of ones in the binarised index I_k . For example, $|\{ac\}| = 2^2 - 1 = 3$, where $\iota = (5)_{\text{bin}}, \#1' = 2$.

So far, we introduced the penalty value which is the smoothness of a decision represented by the Belief function. The penalty value is, especially for complex expression, constructed as the sum over single statements' penalties. However, the influence of each single penalty is unconsidered. Therefore, we are looking for a proper representation of such influence, i.e. a weighting of single penalties. In analogy to the information theoretical entropy, such a weighting can be done by a probability measure which reflects the significance of the statements' penalty term. In the framework of DST an appropriate measure is given by the bpa m . Moreover, by applying m to the penalty we incorporate the (expert) knowledge contained in the input values' structure.

$$\tilde{c}^*(\underline{x}_i, \underline{x}_j, A) = \frac{1}{/A/} \sum_{B \subseteq A} m(B) \bar{c}(B, \underline{x}_j). \quad (9)$$

Finally, combining (2) and (9) we get the Belief function which also takes the history into account to get a smoothed, i.e. penalised, behaviour.

$$Bel(A | \underline{x}_j) = \sum_{B \subseteq A} m(B) - \frac{1}{|A|} \sum_{B \subseteq A} m(B) \bar{c}(B, \underline{x}_j). \quad (10)$$

The function presented in (10) is a Belief-like function. As shown in Sect. A the value of the function is $0 \leq Bel \leq 1$, but 1 is only achieved in the case where $\bar{c}(\underline{x}_i, \underline{x}_j) = 0$, $\forall \underline{x}_i, \underline{x}_j$; otherwise, $Bel < 1$. The future research is oriented to handle this circumstance.

5 Illustrative Example

In this section an example as a proof of concept for our framework is given. Again, we show the result for the Belief function only. To prove our idea, we implemented the standard DST framework as well as ours in Matlab applying the methods proposed in [4] providing a fast computation.

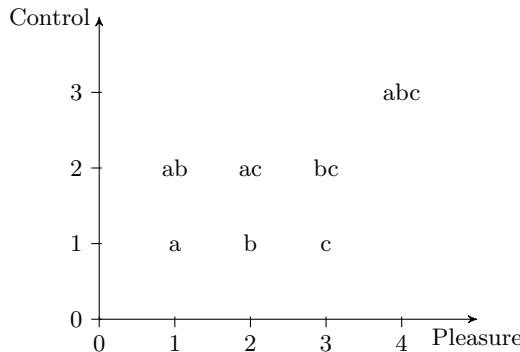


Fig. 2. Collection of samples in a two-dimensional space where the letters represents the corresponding label. The more letters are in the label, the more complex the expression.

In the example, the set is defined as $\mathcal{P}(X) = \{\emptyset, a, b, ab, c, ac, bc, abc\}$ where each letter represents a single statement and $\{ab\}, \{ac\}, \{bc\}, \{abc\}$ are complex expressions. In terms of emotions (cf. Sect. 2) such a statement could be a = “angry”, etc. The vectorised representation in the Pleasure-Control space is given in Fig. 2, e.g. $\{ac\} = \underline{x}_j = \binom{2}{2}$. In this figure, only the first quadrant is shown but this is no limitation for the framework. Further, the positions of the statements/expressions represents no order which might influence the simulation.

Again, the mapping of the representatives x_i onto the vectors \underline{x}_i in this example is not influenced by any application. As discussed in Sect. 3, it is further not important for the method as such. Therefore, the values are assigned without any high-level order. Any other mapping function could be applied as well, not

influencing the general outcome of the combination, given that the structure of the task is kept fix.

To compute the results given in Table 1 we kept the x_j value, i.e. the history, fixed with $x_j = \binom{2}{1}$, which is $\{b\}$. Comparing the belief values with and without penalty we see that the penalised values are smaller than the non-penalised ones, except for the statement which is equal to the previous decision. Thus, it results in a smaller belief for non-stable decisions. Furthermore, we see that the effect depends on the distance between two points/decisions, too. Thus, changing from $\{b\}$ to $\{a\}$, $\{c\}$, or $\{ac\}$ has a lower Euclidian distance as, e.g. $\{b\}$ to $\{ab\}$ (cf. Fig. 2). Analogous observations can be made for the Plausibility function and the Combination rule.

Table 1. For the set A the input values, i.e. the m values, are given. Further, we show the final belief value without (Def. 2) and with penalty (10). Moreover, the normalised penalty values according to (9) are given.

	\emptyset	a	b	ab	c	ac	bc	abc
Input values	0.0	0.100	0.100	0.200	0.100	0.300	0.150	0.050
Belief without penalty	0.0	0.100	0.100	0.400	0.100	0.500	0.350	1.000
Normalised penalty	0.0	0.020	0.000	0.038	0.020	0.040	0.028	0.012
Belief with penalty	0.0	0.080	0.100	0.362	0.080	0.460	0.322	0.988

6 Conclusion

In this paper, we presented an approach towards more stable decisions in the framework of DST, regarding frequent changes in the sensor inputs and the reasonable rate of changes in decisions. Such changes usually occur in sensor systems where the sampling rate is quite short and for each input a combined output has to be generated. On the other hand, in general, long term decisions are based on measures on short time periods. To reduce the effect of unnecessary changes that are often wrong a smoothing term is introduced to the DST. The evolution of the method was shown for the Belief function. The extensions for the Plausibility function and the Combination Rule are analog to those done for the Belief function Bel . The example (cf. Sect. 5) demonstrated the effect of the penalty term for single decisions.

In the future, the method will be applied to systems, where the history x_j is also a sequence. In this context, the approach is transferred to technical applications; that means, this is used in classifiers of emotions. It is to expect that the fluctuations will be reduced. In fact, the approach is not constructed to avoid any kind of changes, but to attenuate the effect of fluctuations.

So far, we showed that the derived function is a Belief-like function and thus, concentrate the further work to extend the idea towards a Belief function.

In future, we will apply this framework to real data sets given by various, multi-modal corpora. This provides us with several options of fusion, e.g. several

classifiers for one modality or multiple classifiers for different input modalities. Especially in the latter case, our approach has advantages in producing less fluctuations in the final decisions.

Acknowledgement. We acknowledge continued support by the Transregional Collaborative Research Centre SFB/TRR 62 “Companion-Technology for Cognitive Technical Systems” funded by the German Research Foundation (DFG). Further, we thank Ingo Siegert for fruitful discussions. Moreover, we acknowledge collaboration within the Center of Behavioral Brain Sciences.

A Proof of Nonnegativity of the Penalised Belief Function

In this section we show that (10) is a valid Belief-like function. Therefore, the result of

$$\text{Bel}(A | \underline{x}_j) = \sum_{B \subseteq A} m(B) - \frac{1}{|A|} \sum_{B \subseteq A} m(B) \bar{c}(\underline{B}, \underline{x}_j)$$

has to be in $[0, 1]$.

Given $\sum_{B \subseteq A} m(B) \in [0, 1]$ and after scaling $\frac{1}{|A|} \sum_{B \subseteq A} m(B) \bar{c}(\underline{B}, \underline{x}_j)$ is as well in $[0, 1]$.

Hence, we have to consider the subtraction only and end up with: is $\text{Bel}(A | \underline{x}_j) \geq 0$?

Rewriting the Belief function, we have to show:

$$\sum_{B \subseteq A} m(B) \left(1 - \frac{1}{|A|} \bar{c}(\underline{B}, \underline{x}_j) \right) \geq 0.$$

Since $m(B) \geq 0$, a sufficient condition is

$$1 - \frac{1}{|A|} \bar{c}(\underline{B}, \underline{x}_j) \geq 0 \quad \forall B \subseteq A.$$

This holds since $|A| \leq 1$ and $\bar{c}(\underline{B}, \underline{x}_j) \in [0, 1]$. □

References

- [1] Böck, R., Limbrecht, K., Siegert, I., Glüge, S., Walter, S., Wendemuth, A.: Combining mimic and prosodic analyses for user disposition classification. In: Proceedings of the 23. Konferenz Elektronische Sprachsignalverarbeitung (ESSV 2012), pp. 220–228 (2012)
- [2] Böck, R., Limbrecht, K., Walter, S., Hrabal, D., Traue, H.C., Glüge, S., Wendemuth, A.: Intraindividual and interindividual multimodal emotion analyses in human-machine-interaction. In: 2012 IEEE International Multi-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support (CogSIMA), pp. 59–64 (2012)

- [3] Dempster, A.P.: A generalization of bayesian inference. In: Yager, R.R., Liu, L. (eds.) *Classic Works of the Dempster-Shafer Theory of Belief Functions*. STU FUZZ, vol. 219, pp. 73–104. Springer, Heidelberg (2008); reprint of a talk given at a Research Methods Meeting of the Royal Statistical Society (February 14, 1968)
- [4] Kennes, R., Smets, P.: Fast algorithms for Dempster-Shafer theory. In: Bouchon-Meunier, B., Zadeh, L.A., Yager, R.R. (eds.) *IPMU 1990*. LNCS, vol. 521, pp. 14–23. Springer, Heidelberg (1991)
- [5] Koelstra, S., Mühl, C., Patras, I.: Eeg analysis for implicit tagging of video data. In: 3rd International Conference on Affective Computing and Intelligent Interaction and Workshops, pp. 1–6. IEEE (2009)
- [6] Krell, G., Glodek, M., Panning, A., Siegert, I., Michaelis, B., Wendemuth, A., Schwenker, F.: Fusion of fragmentary classifier decisions for affective state recognition. In: Proceedings of the First IAPR Workshop on Multimodal Pattern Recognition of Social Signals in Human Computer Interaction, Tsukuba Science City, Japan (2012) (to appear)
- [7] Dupin de Saint-Cyr, F., Lang, J., Sabatier, P., Schiex, T.: Penalty logic and its link with dempster-shafer theory. In: Proc. of the 10th Conf. on Uncertainty in Artificial Intelligence, pp. 204–211. Morgan Kaufmann (1994)
- [8] Scherer, K.R.: What are emotions? and how can they be measured? *Social Science Information* 44(4), 695–729 (2005)
- [9] Sentz, K., Ferson, S.: Combination of evidence in dempster-shafer theory. Tech. rep., Sandia National Laboratories (2002)
- [10] Shafer, G.: *A Mathematical Theory of Evidence*. Princeton University Press (1976)
- [11] Vlasenko, B., Böck, R., Wendemuth, A.: Modeling affected user behavior during human-machine interaction. In: Proceedings of the 5th International Conference on Speech Prosody 2010, Chicago, Illinois, USA, pp. 44–47 (2010)
- [12] Vlasenko, B., Philippou-Hübner, D., Prylipko, D., Böck, R., Siegert, I., Wendemuth, A.: Vowels formants analysis allows straightforward detection of high arousal emotions. In: Proceedings of the IEEE International Conference on Multimedia and Expo, ICME 2011, Barcelona, Spain (2011), elec. resource
- [13] Walter, S., et al.: Multimodal Emotion Classification in Naturalistic User Behavior. In: Jacko, J.A. (ed.) *Human-Computer Interaction, Part III, HCII 2011*. LNCS, vol. 6763, pp. 603–611. Springer, Heidelberg (2011)
- [14] Zhang, L.: Representation, independence, and combination of evidence in the dempster-shafer theory. In: Yager, R.R., Fedrizzi, M., Kacprzyk, J. (eds.) *Advances in the Dempster-Shafer Theory of Evidence*, pp. 51–69. Wiley (1994)

A Probabilistic Model for Prioritizing Engineering Design Requirements in Uncertain QFD

Hong-Bin Yan* and Tieju Ma

School of Business, East China University of Science and Technology
Meilong Road 130, Shanghai 200237, P.R. China
{hbyan,tjma}@ecust.edu.cn

Abstract. Quality function deployment (QFD) is a planning and problem-solving tool gaining wide acceptance for translating customer requirements (CRs) into the design requirements (DRs) of a product. Deriving the priority order of DRs from input variables is a crucial step in applying QFD. Due to the inherent vagueness or imprecision in QFD, the use of fuzzy linguistic variables for prioritizing DRs has become more and more important in QFD applications. This paper proposes a probabilistic model for prioritizing engineering DRs in QFD based on the order-based semantics of linguistic information and fuzzy preference relations of linguistic profiles, under random interpretations of customers, design team, and CRs. A case study taken from the literature is used to illuminate the proposed technique and to compare with the previous techniques. This approach enhances the fuzzy-computation-based models proposed in the previous studies by eliminate the burden of quantifying qualitative concepts.

1 Introduction

Quality function deployment (QFD) frameworks are useful tools for constructing a new product development plan that enables the clear itemization of the voice of the customer and for translating them into the various stages of *product planning, design, engineering, and manufacturing* by systematically evaluating each solution to achieve higher customer satisfaction [1]. A most commonly seen QFD consists of four inter-linked stages: Stage I to translate CRs into design requirements (DRs); Stage II to translate important DRs into product characteristics; Stage III to translate important product characteristics into manufacturing operations; and Stage IV to translate key manufacturing operations into operations and control. The first stage of QFD, usually called house of quality (HOQ), is of fundamental and strategic importance in the QFD system, since it is in this stage that the CRs for the product are identified and converted into appropriate DRs to fulfil customer satisfaction. The structures and analyzing methods of

* Corresponding author.

the other three QFD stages are essentially the same as the first one [9]. Therefore, the HOQ stage will be studied in this paper, in which deriving the final prioritization of DRs from input variables is a crucial step [3].

The inherent vagueness or impreciseness in QFD presents a special challenge to the effective prioritisation of DRs [3]. Therefore, numerical studies have been conducted on how to prioritize DRs with fuzzy linguistic variables [16] semantically represented by fuzzy sets [15]. For example, Khoo and Ho [6] developed an approach based on possibility theory and fuzzy arithmetic to address the ambiguity involved in various relationships and outlined the framework of a fuzzy linguistic QFD. Zhou [17] proposed an approach to prioritize DRs through a fuzzy ranking procedure and to optimize improvements using a mixed integer programming model, in which the relative importance of CRs were determined by the analytical hierarchy process (AHP) and were assumed to be crisp numbers. Wang [12] used fuzzy arithmetic to compute the technical importance ratings of DRs and the outranking approach based on possibility and necessity measures to prioritize DRs. Shen [11] employed fuzzy arithmetic to calculate the fuzzy priority weights of DRs and defuzzified them using the mean of maxima method and the centroid defuzzification method. Liu [10] devised a method that could prioritize DRs without knowing their exact membership functions by means of fuzzy weighted average (FWA) method [5]. Chen *et al.* [3] calculated the priority weights of DRs using the FWA and fuzzy expected value (FEV) operator [8]. Kwong *et al.* [7] proposed a two-stage fuzzy group decision making approach. In their approach, the NLP-FWA method and fuzzy expected value operator were first used to derive a ranking order of DRs under each customer's judgment, and then all customers' preferences on the ranking of DRs were synthesised via 0-1 integer programming to achieve a consensus ranking. Wang [14] proposed a different fuzzy group decision making procedure for prioritising DRs under uncertainty. The proposed approach did not aggregate the individual judgments of customers and design team members, but rather aggregated the technical importance ratings of DRs.

Essentially, the above-mentioned studies make use of the associated fuzzy membership functions of linguistic labels based on the fuzzy extension principle [15]. Despite their great success in prioritizing DRs in QFD, existing approaches simultaneously has, as any fuzzy-computation-based approach, an unavoidable limitation of information loss caused by the process of linguistic approximation, which consequently implies a lack of precision in the final result. Such an information loss has been well verified in the general context of fuzzy linguistic decision by [4]. In addition, the fuzzy-set-based semantics of linguistic labels is often defined subjectively and context-dependently, which may sensitively influence the final prioritization results. For example, even the same linguistic labels were used to assess the importance of CRs, different membership functions were defined by [7,12,14]. In this sense, direct computations with words (labels) may provide a better solution to the fuzzy linguistic QFD.

Toward this end, the main focus of this paper is to propose an alternative fuzzy linguistic model for prioritizing DRs in QFD, which performs computation solely

based on the order-based semantics of linguistic labels and fuzzy preference relations of linguistic profiles. The rest of this paper is organized as follows. Section 2 briefly describes the linguistic evaluation framework and main tasks for implementation of the evaluation. Section 3 introduces the concept of stochastic dominance degrees (fuzzy preference relations) of linguistic profiles and its several interesting properties. Section 4 presents a probabilistic model for prioritizing DRs based on random interpretations of customers, design team, and CRs. Section 5 examines a numerical example to show the effectiveness of the proposed approach, results are also compared with previous techniques. Finally, Section 6 concludes the paper with some concluding remarks.

2 The Fuzzy Linguistic Assessment Based Framework

Prioritizing DRs includes both “CR management” and “product development” systems, which begins by sampling the desires and preferences of customers of a product through marketing surveys or interviews, and organizes them as a set of CRs [2]. Formally, let $\mathcal{CR} = \{CR_1, CR_2, \dots, CR_M\}$ be a set of CRs. The importance of CRs has then to be determined to continue the QFD process. In order to obtain the importance of the CRs, a set of customers to be surveyed in a target market is collected such that $\mathcal{C} = \{C_1, C_2, \dots, C_K\}$ with a weighting vector $(\gamma_1, \gamma_2, \dots, \gamma_K)$, $\sum_k \gamma_k = 1$. Each customer has to provide his/her judgment of importance toward each CR with linguistic variable $\mathcal{L}^I = \{L_1^I, L_2^I, \dots, L_{G_I}^I\}$, where G_I is the cardinality of the linguistic variable \mathcal{L}^I . Formally, the importance judgement for customer requirement CR_m provided by customer C_k is denoted by $CRI_{mk} \in \mathcal{L}^I$, as shown in Table 1.

Moreover, a set of DRs is collected from the design team for a QFD problem such that $\mathcal{DR} = \{DR_1, DR_2, \dots, DR_N\}$. The QFD is based on a process involving teams of multidisciplinary representatives from all stages of product development and production. Translations from CRs to engineering DRs are carried out by a QFD design team. A design team is collected as $\mathcal{E} = \{E_1, E_2, \dots, E_J\}$, which is also assigned a weighting vector $(\beta_1, \beta_2, \dots, \beta_J)$, $\sum_{j=1}^J \beta_j = 1$. The design team members are then asked to provide their judgments of the relationships between CRs and DRs using linguistic variable $\mathcal{L}^R = \{L_1^R, L_2^R, \dots, L_{G_R}^R\}$, where G_R is the cardinality of the linguistic variable \mathcal{L}^R . Formally, the linguistic judgement for the relationship between customer need CR_m and design requirement DR_n given by design team member E_j is denoted by $R_{mnj} \in \mathcal{L}^R$, as shown in Table 1.

From the linguistic evaluation data collected, we then aim at developing a suitable computing method that allows for deriving a weighting vector $\mathbf{V} = \{V_1, V_2, \dots, V_N\}$, $\sum_n V_n = 1$ of the DRs, so as to support for the prioritizing decision of QFD.

3 Fuzzy Preference Relations from Linguistic Profiles

Let Z_1 and Z_2 be two independent discrete random variables with probability distributions $p_1(z)$ and $p_2(z)$ over a finite set of linguistic labels $\mathcal{L} = \{L_1, L_2, \dots, L_G\}$

Table 1. Fuzzy linguistic data of a HOQ

Customer requirements (CRs)	Importance rating of CRs	Design requirements (DRs)			
		DR ₁	DR ₂	...	DR _N
CR_1	$C_1 : CRI_{11}$	$E_1 : R_{111}$	$E_1 : R_{121}$...	$E_1 : R_{1N1}$
	$C_2 : CRI_{12}$	$E_2 : R_{112}$	$E_2 : R_{122}$...	$E_2 : R_{1N2}$
	\vdots	\vdots	\vdots	\ddots	\vdots
	$C_K : CRI_{1K}$	$E_J : R_{11J}$	$E_J : R_{12J}$...	$E_J : R_{1NJ}$
CR_2	$C_1 : CRI_{21}$	$E_1 : R_{211}$	$E_1 : R_{221}$...	$E_1 : R_{2N1}$
	$C_2 : CRI_{22}$	$E_2 : R_{212}$	$E_2 : R_{222}$...	$E_2 : R_{2N2}$
	\vdots	\vdots	\vdots	\ddots	\vdots
	$C_K : CRI_{2K}$	$E_J : R_{21J}$	$E_J : R_{22J}$...	$E_J : R_{2NJ}$
\vdots	\vdots	\vdots	\vdots	\ddots	\vdots
CR_M	$C_1 : CRI_{M1}$	$E_1 : R_{M11}$	$E_1 : R_{M21}$...	$E_1 : R_{MN1}$
	$C_2 : CRI_{M2}$	$E_2 : R_{M12}$	$E_2 : R_{M22}$...	$E_2 : R_{MN2}$
	\vdots	\vdots	\vdots	\ddots	\vdots
	$C_K : CRI_{MK}$	$E_J : R_{M1J}$	$E_J : R_{M2J}$...	$E_J : R_{MNJ}$
		V_1	V_2	...	V_N
		Weighting vector of DRs			

with $L_1 < L_2 < \dots < L_G$, referred to as linguistic profiles. Let $L_{g_1} \in \mathcal{L}$ and $L_{g_2} \in \mathcal{L}$ be outcomes of Z_1 and Z_2 , respectively. Let $\Pr(L_{g_1} \geq L_{g_2})$, $\Pr(L_{g_1} = L_{g_2})$, and $\Pr(L_{g_1} \leq L_{g_2})$ denote the probabilities of $L_{g_1} \geq L_{g_2}$, $L_{g_1} = L_{g_2}$, and $L_{g_1} \leq L_{g_2}$, respectively. Since Z_1 and Z_2 are stochastically independent, we have

$$\begin{aligned}\Pr(L_{g_1} \geq L_{g_2}) &= \sum_{g_1=1}^G \sum_{g_2=1}^{g_1} p_1(L_{g_1}) \cdot p_2(L_{g_2}) \\ \Pr(L_{g_1} = L_{g_2}) &= \sum_{g_1=1}^G p_1(L_{g_1}) \cdot p_2(L_{g_1}) \\ \Pr(L_{g_1} \leq L_{g_2}) &= \sum_{g_1=1}^G \sum_{g_2=g_1}^G p_1(L_{g_1}) \cdot p_2(L_{g_2})\end{aligned}\quad (1)$$

Accordingly, we have

$$\begin{aligned}\Pr(L_{g_1} > L_{g_2}) &= \Pr(L_{g_1} \geq L_{g_2}) - \Pr(L_{g_1} = L_{g_2}) \\ \Pr(L_{g_1} < L_{g_2}) &= \Pr(L_{g_1} \leq L_{g_2}) - \Pr(L_{g_1} = L_{g_2})\end{aligned}$$

The case $L_{g_1} = L_{g_2}$ can be regarded as a situation where $L_{g_1} > L_{g_2}$ and $L_{g_1} < L_{g_2}$ occur with the same probability simultaneously. Thus, in the situation of $L_{g_1} = L_{g_2}$, the probabilities $\Pr(L_{g_1} > L_{g_2} | L_{g_1} = L_{g_2}) = \Pr(L_{g_1} < L_{g_2} | L_{g_1} = L_{g_2}) = 0.5$.

Due to the above analysis, we give the definition of stochastic dominance degree of two discrete probability distributions over a linguistic label set as follows.

Definition 1. Let Z_1 and Z_2 be two independent discrete random variables with (discrete) probability distributions p_1 and p_2 over a finite set of linguistic labels $\mathcal{L} = \{L_1, L_2, \dots, L_G\}$ with $L_1 < L_2 < \dots < L_G$, then the stochastic dominance degrees of p_1 over p_2 (p_2 over p_1) are given by

$$\begin{aligned} D_{12} &= \Pr(L_{g_1} \geq L_{g_2}) - 0.5\Pr(L_{g_1} = L_{g_2}) \\ D_{21} &= \Pr(L_{g_2} \geq L_{g_1}) - 0.5\Pr(L_{g_1} = L_{g_2}) \end{aligned} \quad (2)$$

Extending two random variables to a vector of N random variables $\mathbf{Z} = (Z_1, Z_2, \dots, Z_N)$, we are able to derive a matrix \mathbf{D} of stochastic dominance degrees of the N discrete random variables. such a matrix of stochastic dominance degrees has the following interesting corollaries.

Corollary 1. Let $\mathbf{D} = [D_{nl}]_{N \times N}$ be a matrix of stochastic dominance degrees of the N discrete random variables, then we have $D_{nl} + D_{ln} = 1$, for each $n, l \in \{1, 2, \dots, N\}$.

Proof. By Eq. (2), we can obtain

$$\begin{aligned} D_{nl} + D_{ln} &= \Pr(L_{g_n} \geq L_{g_l}) + \Pr(L_{g_l} \geq L_{g_n}) - \Pr(L_{g_n} = L_{g_l}) \\ &= \sum_{g_n=1}^G \left\{ p_n(L_{g_n}) \cdot \left[\sum_{g_l=1}^{g_n} p_l(L_{g_l}) + \sum_{g_l=g_n+1}^G p_l(L_{g_l}) \right] \right\} - \sum_{g_n=1}^G p_n(L_{g_n}) \cdot p_l(L_{g_n}) \\ &= \sum_{g_n=1}^G \left\{ p_n(L_{g_n}) \cdot \left[\sum_{g_l=1}^G p_l(L_{g_l}) + p_l(L_{g_n}) \right] \right\} - \sum_{g_n=1}^G p_n(L_{g_n}) \cdot p_l(L_{g_n}) \\ &= \sum_{g_n=1}^G p_n(L_{g_n}) \cdot \sum_{g_l=1}^G p_l(L_{g_l}) \end{aligned} \quad (3)$$

Since $\sum_{g_n=1}^G p_n(L_{g_n}) = 1$ and $\sum_{g_l=1}^G p_l(L_{g_l}) = 1$, we know $D_{nl} + D_{ln} = 1$.

Corollary 2. Let $\mathbf{D} = [D_{nl}]_{N \times N}$ be a matrix of stochastic dominance degrees, the stochastic dominance degree of one discrete random variable over itself is $D_{nn} = 0.5$, for each $n \in \{1, 2, \dots, N\}$.

Proof. From Corollary 1, we know $D_{nn} + D_{nn} = 1$. Therefore, we know $D_{nn} = 0.5, n = 1, \dots, N$.

Interestingly, the matrix \mathbf{D} of stochastic dominance degrees with respect to a vector of N random variables $\mathbf{Z} = (Z_1, Z_2, \dots, Z_N)$ satisfies the following properties of fuzzy preference relations.

1. When $D_{nl} = 1$, it indicates that Z_n is absolutely preferred to Z_l , i.e., indicates the maximum degree of preference of Z_n over Z_l .
2. When $0.5 < D_{nl} < 1$, it indicates that Z_n is slightly preferred to Z_l .
3. When $D_{nl} = 0.5$, there is no preference between Z_n and Z_l .
4. When $0 < D_{nl} < 0.5$, it indicates that Z_l is slightly preferred to Z_n .
5. When $D_{nl} = 0$, it indicates that Z_l is absolutely preferred to Z_n .

Therefore, the matrix of stochastic dominance degrees of the random variable set \mathcal{Z} is in fact a matrix of fuzzy preference relations formulated as

$$\mu_{\mathbf{D}} : (Z_n, Z_l) \in \mathcal{Z} \times \mathcal{Z} \longrightarrow D_{nl} \in [0, 1], \quad (4)$$

where $n, l = 1, \dots, N$, and D_{nl} reflects the degree of fuzzy preference relation of Z_n over Z_l .

4 Our Proposed Probabilistic Model

4.1 Aggregations of Individual Judgments

From a practical point of view, given a customer requirement, if there is an ideal customer, say C_I , whose judgment on the importance of the customer requirement the decision-maker completely believes in, then it is enough for the decision-maker to use C_I 's assessment as the importance of the customer requirement. However, this is not generally the case in practice. A pool of multi-disciplinary and multi-functional customers from a survey market is therefore called to express their judgments regarding the importance of customer needs, on one hand, to collect enough information for the QFD problem from various points of view; and on the other hand, to reduce the subjectivity of the QFD problem. In this regards, $p_C(C_k)$, for each $k = 1, 2, \dots, K$, may be interpreted as the probability that the decision-maker randomly selects customer C_k from the customer set \mathcal{C} as a sufficient source of information for the purpose of QFD. Such a probability distribution p_C may come from the decision-maker's knowledge about the customers. Lacking any such knowledge, a uniform probability distribution will be assumed as $p_C(C_k) = \frac{1}{|\mathcal{C}|} = \frac{1}{K}, k = 1, 2, \dots, K$.

From the perspective of group decision analysis, a weighting vector $(\gamma_1, \gamma_2, \dots, \gamma_K)$ is also often associated with the set of customers \mathcal{C} such that $\gamma_k \in [0, 1]$ and $\sum_{k=1}^K \gamma_k = 1$. In this sense, the set of customers plays the role of states of the world and the weighting vector serves as the subjective probabilities assigned to the states of the world such that $p_C(C_k) = \gamma_k, k = 1, 2, \dots, K$.

In addition, when a customer C_k provides his/her judgment for the importance of a customer requirement CR_m , a probability distribution $p_{\mathcal{L}^I}(L_g^I | CR_m, C_k)$ of his/her opinion over the linguistic variable \mathcal{L}^I can be elicited as

$$p_{\mathcal{L}^I}(L_g^I | CR_m, C_k) = \begin{cases} 1, & \text{if } CRI_{mk} = L_g^I; \\ 0, & \text{otherwise.} \end{cases} \quad (5)$$

where $k = 1, \dots, K, g = 1, \dots, G_I$. The value $p_{\mathcal{L}^I}(L_g^I | CR_m, C_k)$ can be seen as a prior probability that the decision-maker believes in the linguistic label L_g^I is appropriate enough to describe the importance of customer requirement CR_m , given the judgment of customer C_k .

Taking the subjective probability distribution $p_C(C_k)$ into consideration, a posterior probability (collective probability) distribution of customer requirement CR_m over linguistic variable \mathcal{L}^I can be obtained as

$$\begin{aligned}
p_{\mathcal{L}^I}(L_g^I | CR_m) &= \sum_{k=1}^K p_C(C_k) \cdot p_{\mathcal{L}^I}(L_g^I | CR_m, C_k) \\
&= \sum_{k=1}^K \gamma_k \cdot p_{\mathcal{L}^I}(L_g^I | CR_m, C_k)
\end{aligned} \tag{6}$$

where $m = 1, \dots, M, g = 1, \dots, G_I$. The derived probability distribution

$$[p_{\mathcal{L}^I}(L_1^I | CR_m), p_{\mathcal{L}^I}(L_2^I | CR_m), \dots, p_{\mathcal{L}^I}(L_{G_I}^I | CR_m)]$$

will be refereed to as *linguistic importance profile* of CR_m .

Furthermore, a design team \mathcal{E} involved in the QFD problem is also chosen to provide their absolute linguistic judgments regarding the relationships between CRs and DRs. Similar to the formulation of linguistic performance profiles of CRs, the design team plays the role of states of the world and the weighting vector of the design team serves as the subjective probabilities assigned to the states of the world such that $p_{\mathcal{E}}(E_j) = \beta_j, j = 1, 2, \dots, J$. A probability distribution $p_{\mathcal{L}^R}(L_g^R | CR_m, DR_n, E_j)$ of design team member E_j 's judgment with respect to the relationship between customer need CR_m and design requirement DR_n over linguistic variable \mathcal{L}^R can be easily elicited. The value $p_{\mathcal{L}^R}(L_g^R | CR_m, DR_n, E_j)$ can be seen as a prior probability that the decision-maker believes in the linguistic label L_g^R is appropriate enough to describe the relationship between customer requirement CR_m and design requirement DR_n , given design team member E_j 's judgment. Taking the subjective probability distribution $p_{\mathcal{E}}(E_j)$ into consideration, a posterior probability distribution of the relationship between customer requirement CR_m and design requirement DR_n over linguistic variable \mathcal{L}^R can be obtained by

$$p_{\mathcal{L}^R}(L_g^R | CR_m, DR_n) = \sum_{j=1}^J \beta_j \cdot p_{\mathcal{L}^R}(L_g^R | CR_m, DR_n, E_j) \tag{7}$$

where $g = 1, \dots, G_R, n = 1, \dots, N$, and $(\beta_1, \beta_2, \dots, \beta_J)$ is the weighting vector of the design team. The derived probability distribution

$$[p_{\mathcal{L}^R}(L_1^R | CR_m, DR_n), p_{\mathcal{L}^R}(L_2^R | CR_m, DR_n), \dots, p_{\mathcal{L}^R}(L_{G_R}^R | CR_m, DR_n)]$$

will be refereed to as *linguistic relationship profile* between customer requirement CR_m and design requirement DR_n .

4.2 Aggregation of Individual Linguistic Relationship Profiles

In our context, the importance information of each customer requirement CR_m is expressed in terms of a probability distribution over \mathcal{L}^I such that

$$[p_{\mathcal{L}^I}(L_1^I | CR_m), p_{\mathcal{L}^I}(L_2^I | CR_m), \dots, p_{\mathcal{L}^I}(L_{G_I}^I | CR_m)]$$

where $m = 1, \dots, M$. Consequently, a matrix $\mathbf{D}(\mathcal{CR})$ of fuzzy preference relations (stochastic dominance degrees) of the CRs can be derived from the linguistic importance profiles with respect to the set of CRs as

$$\mathbf{D}(\mathcal{CR}) = \begin{matrix} & CR_1 & CR_2 & \dots & CR_M \\ CR_1 & 0.5 & D_{12|\mathcal{CR}} & \dots & D_{1M|\mathcal{CR}} \\ CR_2 & D_{21|\mathcal{CR}} & 0.5 & \dots & D_{2M|\mathcal{CR}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CR_M & D_{M1|\mathcal{CR}} & D_{M2|\mathcal{CR}} & \dots & 0.5 \end{matrix} \quad (8)$$

where $D_{ml|\mathcal{CR}}$ denotes the stochastic dominance degree (fuzzy preference relation) of CR_m over CR_l , calculated by Definition 1 in Section 3. A weighting vector $\mathbf{W} = (W_1, W_2, \dots, W_M)$ of the set of CRs can then be induced from the matrix of stochastic dominance degrees $\mathbf{D}(\mathcal{CR})$ in Eq. (8) as

$$W_m = \frac{1}{M} \sum_{l=1}^M \frac{D_{nl|\mathcal{CR}}}{\sum_{n=1}^M D_{nl|\mathcal{CR}}} \quad (9)$$

In addition, we know the linguistic relationship profile between customer need CR_m and design requirement DR_n is expressed by a probability distribution over \mathcal{L}^R . Similar to the probabilistic interpretation of the customers and design team members in Section 4.1, the derived weights \mathbf{W} can also be viewed as a probability distribution $p_{\mathcal{CR}}$, i.e., a random interpretation of customer requirements. From a practical point of view, if there is an ideal customer need, denoted as CR_I , which the decision-maker completely believes in to represent the factor(s) of customer satisfaction, then it is enough to use the ideal customer need CR_I in the QFD. However, the QFD is typically characterised by multiple customer needs. In this sense, the set of CRs plays the role of states of the world and the derived weights \mathbf{W} play the role of subjective probabilities assigned to the states such that

$$p_{\mathcal{CR}}(CR_m) = W_m, m = 1, \dots, M. \quad (10)$$

In our QFD, the probability $p_{\mathcal{L}^R}(L_g^R|CR_m, DR_n)$ can be viewed as a prior probability that the decision-maker believes in the linguistic label L_g^R is appropriate enough to describe the relationship between overall customer satisfaction and design requirement DR_n . Taking $p_{\mathcal{CR}}(CR_m)$ into consideration, a posterior probability (collective probability) can be obtained as

$$\begin{aligned} p_{\mathcal{L}^R}(L_g^R|DR_n) &= \sum_{n=1}^N p_{\mathcal{CR}}(CR_m) \cdot p_{\mathcal{L}^R}(L_g^R|CR_m, DR_n) \\ &= \sum_{n=1}^N W_m \cdot p_{\mathcal{L}^R}(L_g^R|CR_m, DR_n), g = 1, 2, \dots, G_R. \end{aligned} \quad (11)$$

which represents the overall profile of design requirement DR_n , i.e., the relationship between design requirement DR_n and overall customer satisfaction.

4.3 Choice Function

With the overall profiles of DRs obtained, a matrix $\mathbf{D}(\mathcal{DR})$ of stochastic dominance degrees of the relationship between design requirement DR_n and overall customer satisfaction is constructed as

$$\mathbf{D}(\mathcal{DR}) = \begin{matrix} & DR_1 & DR_2 & \dots & DR_N \\ DR_1 & 0.5 & D_{12|\mathcal{DR}} & \dots & D_{1N|\mathcal{DR}} \\ DR_2 & D_{21|\mathcal{DR}} & 0.5 & \dots & D_{2N|\mathcal{DR}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ DR_N & D_{N1|\mathcal{DR}} & D_{N2|\mathcal{DR}} & \dots & 0.5 \end{matrix} \quad (12)$$

where $D_{nl|\mathcal{DR}}$ denotes the fuzzy preference relation of DR_n over DR_l , calculated by Definition 1 in Section 3. Finally, a choice function for prioritising the design requirements is introduced as

$$V_n = \frac{1}{N} \sum_{l=1}^N \frac{D_{ml|\mathcal{DR}}}{\sum_{m=1}^N D_{ml|\mathcal{DR}}} \quad (13)$$

The vector $\mathbf{V} = (V_1, V_2, \dots, V_N)$ will be used to prioritize the DRs. The greater V_n is, the higher priority the design requirement is.

5 A Case Study—The Design of a Flexible Manufacturing System

5.1 Problem Descriptions

The example discussing the basic design requirements of a flexible manufacturing system (FMS) [3,14] is applied here to illustrate the idea proposed. In that example, eight major CRs are identified to represent the biggest concerns of the customers for the design of an FMS, as shown in Table 2 (indexed by CRs). Based on the design team's experience and expert knowledge, 10 DRs are identified responding to the eight major CRs, as shown in Table 2 (indexed by DRs). According to the customer, the design team needs to translate the customer's wishes into technical specifications from the DRs. These requirements must be in balance with available expertise and the given time and cost frame of the project. Consequently, the design team needs to determine the priorities of the design requirements in developing a new model of an FMS and to provide an objective means of assuring that requirements have been met.

In the QFD planning process, the first step is to determine the weighting vector of customer needs \mathcal{CR} . The linguistic variable for representing the importance of \mathcal{CR} is provided in Eq. (14), indexed by \mathcal{L}^I . The 10 surveyed customers, $\mathcal{C} = \{C_1, C_2, \dots, C_{10}\}$ were then asked to assess the 8 customer needs by making use of linguistic variable \mathcal{L}^I . Similar to the CRs, the relationship measure between CRs and DRs was assessed by a QFD design team with three team

Table 2. Customer requirements and design engineering requirements

Customer requirements (CRs)		Design requirements (DRs)	
CR_1	High production volume	DR_1	Automatic gauging
CR_2	Short setup time	DR_2	Tool change system
CR_3	Load-carrying capacity	DR_3	Tool monitoring system
CR_4	User-friendliness	DR_4	Coordinate measuring machine
CR_5	Associated functions	DR_5	Automated guided vehicle
CR_6	Modularity	DR_6	Conveyor
CR_7	Wide tool variety	DR_7	Programmable logic controller (PLC)
CR_8	Wide product variety	DR_8	Storage and retrieval system (S&R system)
		DR_9	Modular fixture
		DR_{10}	Robots

members, $\mathcal{E} = \{E_1, E_2, E_3\}$. The linguistic variable used for assessing the relationships between CRs and DRs by the design team is presented in Eq. (14), indexed by \mathcal{L}^R .

$$\begin{aligned}
 \mathcal{L}^I &= \{L_1^I, L_2^I, L_3^I, L_4^I, L_5^I, L_6^I, L_7^I\} \\
 &= \{\text{Very unimportant}, \text{Quite unimportant}, \text{Unimportant}, \text{Slightly important}, \\
 &\quad \text{Moderately important}, \text{Important}, \text{Very important}\} \\
 \mathcal{L}^R &= \{L_1^R, L_2^R, L_3^R, L_4^R\} \\
 &= \{\text{Very weak}, \text{Weak}, \text{Moderate}, \text{Strong}\}
 \end{aligned} \tag{14}$$

The fuzzy linguistic importance of the 8 CRs assessed by the 10 surveyed customers and the fuzzy linguistic relationship matrix between the 8 customer needs and the 10 design requirements assessed by each of the 3 design team members, are shown in [14], respectively.

5.2 Prioritizing Engineering Design Requirements

As mentioned previously, each customer and each design team member may have a different importance weight in light of his/her design experience and domain knowledge, respectively. Similar to [14], the importance weights of customers and design team members are assumed as follows.

$$\gamma_1 = \gamma_2 = \frac{3}{14}, \gamma_3 = \dots = \gamma_{10} = \frac{1}{14} \text{ and } \beta_1 = \frac{1}{2}, \beta_2 = \frac{1}{3}, \beta_3 = \frac{1}{6}. \tag{15}$$

In order to prioritize the 10 DRs, we first calculate the linguistic importance profiles of CRs and the linguistic relationship profiles between DRs and CRs. With the random interpretation of 10 customers, the weighting vector $(\gamma_1, \gamma_2, \dots, \gamma_{10})$ can be viewed as a subjective probability distribution defined over the customer set \mathcal{C} . According to Eq. (6), we can obtain the linguistic importance profiles of the 8 customer requirements: $\{CR_1, CR_2, \dots, CR_8\}$. The importance profile

Table 3. Overall linguistic profiles of DRs

Linguistic variable \mathcal{L}^R	Design requirements \mathcal{DR}									
	DR_1	DR_2	DR_3	DR_4	DR_5	DR_6	DR_7	DR_8	DR_9	DR_{10}
L_1^R	0.161	0.075	0.106	0.07	0.288	0.225	0.228	0.065	0.129	0.059
L_2^R	0.324	0.261	0.079	0.475	0.488	0.337	0.25	0.286	0.567	0.591
L_3^R	0.282	0.536	0.455	0.333	0.16	0.364	0.244	0.405	0.24	0.331
L_4^R	0.233	0.128	0.36	0.123	0.064	0.073	0.278	0.243	0.064	0.02

of each customer requirement results with a probability distribution over the linguistic variable \mathcal{L}^I . Consequently, a matrix $\mathbf{D}(\mathcal{CR})$ of fuzzy preference relations (stochastic dominance degrees) of the 8 CRs can be derived from the linguistic importance profiles according to Definition 1 in Section 3. According to Eq. (9), a normalised weighting vector of the 8 CRs can be induced from the matrix $\mathbf{D}(\mathcal{CR})$ of fuzzy preference relations of the 8 CRs as

$$\mathbf{W} = (0.154, 0.099, 0.152, 0.118, 0.128, 0.083, 0.147, 0.12), \quad (16)$$

which indicates that “ CR_1 : high production volume, CR_3 : load-carrying capacity, and CR_7 : wide tool variety” are the most three important customer requirements among the 8 ones.

Second, we consider the relationships between CRs and DRs. Similar to the formulation of linguistic importance profiles of CRs, we can also obtain a linguistic relationship profile between each design requirement and each customer requirement via Eq. (7), which results with a probability distribution over linguistic variable \mathcal{L}^R . Taking the weighting vector \mathbf{W} (Eq. (16)) of the 8 CRs into consideration, a linguistic relationship profile between each design requirement and the overall customer satisfaction can be obtained according to Eq. (11), as shown in Table 3.

Finally, according to Definition 1 in Section 3, we can obtain a matrix $\mathbf{D}(\mathcal{DR})$ of fuzzy preference relations of the 10 design requirements. A vector of choice values can then be derived from the matrix $\mathbf{D}(\mathcal{DR})$ via (13) as follows

$$\mathbf{V} = (0.1042, 0.1138, 0.1342, 0.0992, 0.069, 0.0875, 0.1034, 0.1189, 0.0826, 0.0872),$$

which indicates the following prioritisation of the 10 DRs

$$DR_3 \succ DR_8 \succ DR_2 \succ DR_1 \succ DR_7 \succ DR_4 \succ DR_6 \succ DR_{10} \succ DR_9 \succ DR_5.$$

The choice values will be transferred as weights of the design requirements to the second stage of the QFD.

5.3 Comparative Studies with Two Known Approaches

In this section, we compare our proposed approach with [3,14], both of which are based on the NLP-FWA method. Before doing so, we briefly recall the existing three approaches.

In their seminal work of fuzzy linguistic QFD, [3] proposed a method by means of the NLP-FWA method and fuzzy expected value operator [8] to rank DRs in fuzzy linguistic QFD. In detail, their approach can be summarised as follows.

1. First, fuzzy importance weights of CRs and fuzzy relationships between CRs and DRs provided by multiple customers and QFD team members are averaged.
2. Second, the NLP-FWA method is used to determine the fuzzy weights of DRs.
3. Finally, the fuzzy weights of DRs are defuzzified using the fuzzy expected value operator for prioritising DRs.

[14] proposed a different fuzzy group decision making procedure for prioritising DRs under uncertainty. The proposed approach does not aggregate the individual judgments of customers and QFD design team members, but rather aggregates the technical importance ratings of DRs. Wang (2012)'s approach can be summarised as follows.

1. The NLP-FWA method is used to determine the fuzzy weights of DRs with respect to each customer and design team member.
2. The NLP-FWA method is used to determine the overall fuzzy weights of DRs involving different combinations of any customer and any design team member.
3. The fuzzy weights of DRs are defuzzified using the α -level-sets-based centroid method [13] for prioritising DRs.

The prioritisation results by our approach, Chen *et al.*'s, and Wang (2011)'s approaches are shown in Table 4, which differ from the ranking obtained by our proposed method in that the rankings between DR_1 and DR_7 and between DR_6 and DR_{10} are reversed.

1. There is one difference between our approach and Chen *et al.* (2006)'s method. Our proposed approach prioritizes DR_6 and DR_{10} as $DR_6 \succ DR_{10}$, whereas Chen *et al.* (2006)'s method ranks the two DRs as $DR_{10} \succ DR_6$.
2. There are two differences between our approach and Wang (2011)'s approach. Wang (2011)'s approach also prioritizes DR_6 and DR_{10} as $DR_{10} \succ DR_6$. Moreover, Wang (2011)'s approach prioritizes DR_1 and DR_7 as $DR_7 \succ DR_1$, whereas our approach ranks DR_1 and DR_7 as $DR_1 \succ DR_7$, which is consistent with Chen *et al.* (2006)'s approach.

From the above illustrative comparison results, it is clear that our approach have slight difference with each of the three existing approaches. The main reason may come from the information loss. Central to the existing three approaches is the use of NLP-FWA method. Unfortunately, a critical drawback of the FWA aggregation approach is the information loss, since approximation processes are necessary. First, the aggregation function is solved by the fractional programming approach, in which different α -cuts are necessary to derive the final results. However, there is no guidance for users on how to choose the suitable number

Table 4. Prioritization results

Approaches	Prioritisation results
[3]	$DR_3 \succ DR_8 \succ DR_2 \succ \underbrace{DR_1 \succ DR_7}_{\text{group 1}} \succ DR_4 \succ \underbrace{DR_{10} \succ DR_6}_{\text{group 2}} \succ DR_9 \succ DR_5$
[14]	$DR_3 \succ DR_8 \succ DR_2 \succ \underbrace{DR_7 \succ DR_1}_{\text{group 1}} \succ DR_4 \succ \underbrace{DR_{10} \succ DR_6}_{\text{group 2}} \succ DR_9 \succ DR_5$
Our model	$DR_3 \succ DR_8 \succ DR_2 \succ \underbrace{DR_1 \succ DR_7}_{\text{group 1}} \succ DR_4 \succ \underbrace{DR_6 \succ DR_{10}}_{\text{group 2}} \succ DR_9 \succ DR_5$

of alpha values. In the literature, a finite set of α values $\{0, 0.1, \dots, 0.9, 1\}$ are therefore used by the existing three approaches, and this will of course lose information. Second, the final step of three existing approaches is the use of defuzzification methods. There is no uniform guidance, since different defuzzification approaches can be used. For example, Chen *et al.* (2006) used the fuzzy expected value operator to defuzzify the fuzzy weights of DRs; whereas Wang (2011) utilized the α -level-sets-based centroid method to defuzzify the fuzzy weights of DRs. In summary, the fuzzy-computation-based approach is always critiqued by its information loss, which is confirmed by [4].

Moreover, the fuzzy-set-based semantics of each linguistic label is often defined subjectively and context-dependently, which can sensitively influence the final prioritisation. In particular, with different definitions of fuzzy membership functions associated with the corresponding linguistic labels, different prioritisation results may be obtained.

6 Concluding Remarks

In this paper, we have proposed a distinct fuzzy linguistic decision making approach for prioritizing DRs in QFD under uncertainties. The proposed approach first introduces random interpretations of customers, design team members, and CRs. With the random interpretations, linguistic importance profiles of CRs and linguistic relationship profiles between CRs and DRs can be obtained. The stochastic dominance degrees of linguistic profiles are then used to derive the weighting vector of CRs and the choice values for prioritising DRs. The proposed approach has been examined and illustrated with a real design case of a flexible manufacturing system and the results have been compared with those obtained by three known approaches. By performing direct computation on linguistic labels via the stochastic dominance degrees of linguistic profiles in the proposed approach, the burden of quantifying qualitative concepts as well as performing complicated computation on fuzzy numbers can be also eliminated. Consequently, there will be no information in the proposed approach.

Acknowledgement. This study was supported by the National Natural Sciences Foundation of China (NSFC) under Grants #71101050, #71125002, sponsored by Shanghai Pujiang Program, and supported by the Fundamental Research Funds for the Central Universities in China under Grant #WN1123002.

References

1. Chen, L.H., Ko, W.C.: Fuzzy nonlinear models for new product development using four-phase quality function deployment processes. *IEEE Trans. Syst. Man Cybern. Syst. Hum.* 41(5), 927–945 (2011)
2. Chen, L.H., Weng, M.C.: An evaluation approach to engineering design in QFD processes using fuzzy goal programming models. *Eur. J. Oper. Res.* 172, 230–248 (2006)
3. Chen, Y., Fung, R.Y., Tang, J.: Rating technical attributes in fuzzy QFD by integrating fuzzy weighted average method and fuzzy expected value operator. *Eur. J. Oper. Res.* 174(3), 1553–1566 (2006)
4. Herrera, F., Martínez, L.: A 2-tuple fuzzy linguistic representation model for computing with words. *IEEE Trans. Fuzzy Syst.* 8(6), 746–752 (2000)
5. Kao, C., Liu, S.T.: Fractional programming approach to fuzzy weighted average. *Fuzzy Set Syst.* 120(3), 435–444 (2001)
6. Khoo, L.P., Ho, N.C.: Framework of a fuzzy quality function deployment system. *Int. J. Prod. Res.* 34(2), 299–311 (1996)
7. Kwong, C.K., Ye, Y., Chen, Y., Choy, K.L.: A novel fuzzy group decision making approach to prioritising engineering characteristics in QFD under uncertainties. *Int. J. Prod. Res.* 49(19), 5801–5820 (2011)
8. Liu, B., Liu, K.L.: Expected value of fuzzy variable and fuzzy expected value models. *IEEE Trans. Fuzzy Syst.* 10(4), 445–450 (2002)
9. Liu, C.H., Wu, H.H.: A fuzzy group decision-making approach in quality function deployment. *Quality and Quantity* 42(4), 527–540 (2008)
10. Liu, S.T.: Rating design requirements in fuzzy quality function deployment via a mathematical programming approach. *Int. J. Prod. Res.* 43(3), 497–513 (2005)
11. Shen, X.X., Tan, K.C., Xie, M.: The implementation of quality function deployment based on linguistic data. *J. Intell. Manuf.* 12(1), 65–75 (2001)
12. Wang, J.: Fuzzy outranking approach to prioritize design requirements in quality function deployment. *Int. J. Prod. Res.* 37(4), 899–916 (1999)
13. Wang, Y.M.: Centroid defuzzification and the maximizing set and minimizing set ranking based on alpha level sets. *Comput. Ind. Eng.* 57(1), 228–236 (2009)
14. Wang, Y.M.: A fuzzy-normalisation-based group decision making approach for prioritising engineering design requirements in QFD under uncertainty. *Int. J. Prod. Res.* 50(23), 6963–6977 (2012)
15. Zadeh, L.A.: Fuzzy sets. *Inform. Contr.* 8(3), 338–353 (1965)
16. Zadeh, L.A.: The concept of a linguistic variable and its application to approximate reasoning—Part I. *Inform. Sci.* 8(3), 199–249 (1975)
17. Zhou, M.: Fuzzy logic and optimization models for implementing QFD. *Comput. Ind. Eng.* 35(1-2), 237–240 (1998)

Applying OWA Operator to Model Group Behaviors in Uncertain QFD

Hong-Bin Yan^{1,*}, Tieju Ma¹, and Van-Nam Huynh²

¹ School of Business, East China University of Science and Technology
Meilong Road 130, Shanghai 200237, P.R. China
{hbyan,tjma}@ecust.edu.cn

² School of Knowledge Science, Japan Advanced Institute of Science and Technology
1-1 Asahidai, Nomi City, Ishikawa, 923-1292, Japan
huynh@jaist.ac.jp

Abstract. It is a crucial step to derive the priority order of design requirements (DRs) from customer requirements (CRs) in quality function deployment (QFD). However, it is not straightforward to prioritize DRs due to two types of uncertainties: human subjective perception and user variability. This paper proposes an OWA based group decision-making approach to uncertain QFD with an application to a flexible manufacturing system design. The proposed model performs computations solely based on the order-based semantics of linguistic labels so as to eliminate the burden of quantifying qualitative concepts in QFD. Moreover, it incorporates the importance weights of users and the concept of fuzzy majority into aggregations of fuzzy preference relations of different DRs in order to model the group behaviors in QFD. Finally, based on a quantifier-guided net flow score procedure, the proposed model derives a priority ranking with a classification of DRs into important and unimportant ones so as to provide a better decision-support to the decision-maker.

1 Introduction

As an effective customer-driven quality management system, quality function deployment (QFD) incorporates the “voice of the customer” into appropriate company requirements at various product development stages, ranging from product planning and process design to manufacturing and delivery, to create higher customer satisfaction for the product. Among the four inter-linked stages of QFD [1], the first stage of QFD, usually called house of quality (HOQ), is of fundamental and strategic importance, since it is in this stage that the CRs for the product are identified and converted into appropriate DRs to fulfil customer satisfaction. In other words, HOQ links the “voice of the customer” to the “voice of the technician”, through which the process and production plans can be developed in the other stages of the QFD system. The structures and analyzing methods of the other three QFD stages are essentially the same as the first one [8].

* Corresponding author.

Successful implementation of QFD often requires a significant number of subjective judgments from both customers and QFD design team [1,12]. Traditional QFD assumes that most of the input variables are precise and treated as crisp numerical data such as 1-3-9 or 1-5-9. However, the inherent subjective vagueness or impreciseness in QFD presents a special challenge to effective prioritization of DRs [3]. Therefore, numerical studies have been conducted on how to prioritize DRs with fuzzy linguistic variables [16] semantically represented by fuzzy sets [15], e.g. [3,9]. Another type of uncertainty in QFD is the involvement of many customers and design team members in the evaluation of input information of QFD. Input information may have an uncertainty associated with user (customer or design team member) heterogeneity because each user may have a different opinion. In this context, several studies have considered fuzzy group decision-making approaches, e.g. [1,6,12].

In summary, existing studies perform calculations with the associated fuzzy membership functions of linguistic labels based on fuzzy extension principle [15]. However, quantification in terms of fuzzy sets is in fact the process of transforming an ordinal information into a cardinal scale that represents an “arbitrary passage”, which may sometimes be dangerous [4], since it is easy to generate different results by choosing different scales from which to draw the ordinals. Moreover, the fuzzy-set-based semantics of linguistic labels is often defined subjectively and context-dependently, which may sensitively influence the final prioritization results. Even if the quantification process used is rational, existing approaches to prioritize DRs in QFD simultaneously has, as any fuzzy-computation-based approach, an unavoidable limitation of information loss caused by the process of linguistic approximation, which consequently implies a lack of precision in the final result [5]. Regarding the second type of uncertainty, it is necessary to consider the group behaviors of users (both customers and designer team members). On one hand, the information provided from several users can be combined to improve data reliability and accuracy and to include some features that are impossible to perceive with individual users [10]. The users can be treated unequally considering their possible importance differences reflecting the reliability of each information source. For example, product users can provide more valuable judgmental information than non-product users and potential product users [12]. On the other hand, as a basic element underlying group decision-making, the concept of *fuzzy majority* is accepted by most of its members in practice, since it is quite difficult for the solution to be accepted by all users.

Due to the above observations, the main focus of this paper is to propose an OWA based group decision-making approach to prioritize a number of DRs in uncertain QFD with an application to a flexible manufacturing system design. The proposed model, on one hand, performs computations solely based on the order-based semantics of linguistic labels so as to eliminate the burden of quantifying qualitative concepts. On the other hand, it performs group aggregations of fuzzy preference relations based on the weighted ordered weighted average method so as to incorporate the importance weights of users and the concept of fuzzy majority. Moreover, based on a quantifier-guided net flow score procedure, the proposed model derives a

priority ranking with a classification of DRs into important and unimportant ones so as to provide a better decision-support to the decision-maker.

The rest of this paper is organized as follows. After presenting the basic notations in Section 2, Section 3 presents a novel group decision making-approach to prioritize DRs based on the order-based semantics of linguistic labels and aggregation techniques of fuzzy preference relations. Section 4 examines a comparative numerical example to show the effectiveness of the proposed model. The paper is concluded in Section 5.

2 Basic Notations

Prioritizing DRs includes both “CR management” and “product development” systems, which begins by sampling the desires and preferences of customers of a product through marketing surveys or interviews, and organizes them as a set of CRs [2]. Formally, let $\mathcal{CR} = \{CR_1, CR_2, \dots, CR_M\}$ be a set of CRs. The importance of CRs has then to be determined to continue the QFD process. In order to obtain the importance of the CRs, a set of customers to be surveyed in a target market is collected such that $\mathcal{C} = \{C_1, C_2, \dots, C_K\}$ with a weighting vector $(\gamma_1, \gamma_2, \dots, \gamma_K)$, $\sum_k \gamma_k = 1$. Each customer has to provide his/her judgment of importance toward each CR with linguistic variable $\mathcal{L}^1 = \{L_1^1, L_2^1, \dots, L_{G_1}^1\}$. Formally, the importance judgement for customer requirement CR_m provided by customer C_k is denoted by $CRI_{mk} \in \mathcal{L}^1$.

Moreover, a set of DRs is collected from the design team for a QFD problem such that $\mathcal{DR} = \{DR_1, DR_2, \dots, DR_N\}$. The QFD is based on a process involving teams of multidisciplinary representatives from all stages of product development and production. Translations from CRs to engineering DRs are carried out by a QFD design team. A design team is collected as $\mathcal{E} = \{E_1, E_2, \dots, E_J\}$, which is also assigned a weighting vector $(\beta_1, \beta_2, \dots, \beta_J)$, $\sum_{j=1}^J \beta_j = 1$. The design team members are then asked to provide their judgments of the relationships between CRs and DRs using linguistic variable $\mathcal{L}^2 = \{L_1^2, L_2^2, \dots, L_{G_2}^2\}$. Formally, the linguistic judgement for the relationship between customer need CR_m and design requirement DR_n given by design team member E_j is denoted by $R_{mnj} \in \mathcal{L}^2$.

3 Proposed Method

3.1 Fuzzy Preference Relations from Linguistic Information

We have a set of M customer needs $\mathcal{CR} = \{CR_1, CR_2, \dots, CR_M\}$ and a set of K customers $\mathcal{C} = \{C_1, C_2, \dots, C_K\}$. The importance judgment for customer need CR_m provided by customer C_k is denoted by $CRI_{mk} \in \mathcal{L}^1$. Then, an customer C_k 's individual fuzzy preference relation in $\mathcal{CR} \times \mathcal{CR}$ assigns a value in the unit interval $[0, 1]$ for the preference of one customer need over another such that

$$\mu_{\mathbf{D}(C_k)} : (CR_m, CR_l) \in \mathcal{CR} \times \mathcal{CR} \longrightarrow D_{ml}(C_k) \in [0, 1], m, l = 1, \dots, M. \quad (1)$$

The value $D_{ml}(C_k)$ reflects the degree of fuzzy preference relation of CR_m over CR_l under a customer C_k 's subjective judgment, calculated by

$$D_{ml}(C_k) = \begin{cases} 1, & \text{if } CRI_{mk} > CRI_{lk} \\ 0.5, & \text{if } CRI_{mk} = CRI_{lk} \\ 0, & \text{if } CRI_{mk} < CRI_{lk} \end{cases}. \quad (2)$$

The matrix $\mathbf{D}(C_k) = [D_{ml}(C_k)]_{M \times M}$ has the following properties.

- When $D_{ml}(C_k) = 1$, it indicates that CR_m is absolutely preferred to CR_l , i.e. indicates the maximum degree of preference of CR_m over CR_l .
- When $0.5 < D_{ml}(C_k) < 1$, it indicates that CR_m is slightly preferred to CR_l .
- When $D_{ml}(C_k) = 0.5$, there is no preference (i.e. indifference) between CR_m and CR_l .
- When $0 < D_{ml}(C_k) < 0.5$, it indicates that CR_l is slightly preferred to CR_m .
- When $D_{ml}(C_k) = 0$, it indicates that CR_l is absolutely preferred to CR_m .

Such a value function is in fact based on the order-based semantics of linguistic labels, and consequently we will obtain K matrices of fuzzy preference relations of different customer needs \mathcal{CR} under the K customers' judgments.

Then, for each customer C_k , a weighting vector of the M customer requirements can be induced from the matrix of fuzzy preference relations $\mathbf{D}(C_k)$ as $\mathbf{W}^{\mathbf{CR}}(C_k) = (W_1^{\mathbf{CR}}(C_k), W_2^{\mathbf{CR}}(C_k), \dots, W_M^{\mathbf{CR}}(C_k))$, where $W_m^{\mathbf{CR}}(C_k)$ is calculated as follows

$$W_m^{\mathbf{CR}}(C_k) = \frac{1}{M} \sum_{l=1}^M \frac{D_{ml}(C_k)}{\sum_{n=1}^M D_{nl}(C_k)}, \quad k = 1, \dots, K, m = 1, \dots, M. \quad (3)$$

Moreover, the linguistic judgment for the relationship between customer need CR_m and design requirement DR_n given by design team member E_j is denoted by $R_{mnj} \in \mathcal{L}^2$. Similarly, we can derive a matrix $\mathbf{D}(CR_m, E_j) = [D_{nl}(CR_m, E_j)]_{N \times N}$ of fuzzy preference relations in $\mathcal{DR} \times \mathcal{DR}$ for each customer need under each designer team member, where $D_{nl}(CR_m, E_j)$ denotes the fuzzy preference relation of DR_n over DR_l under a design term member's E_j subjective judgment with respect to customer need CR_m , calculated by the following equation

$$D_{nl}(CR_m, E_j) = \begin{cases} 1, & \text{if } R_{mnj} > R_{mlj} \\ 0.5, & \text{if } R_{mnj} = R_{mlj} \\ 0, & \text{if } R_{mnj} < R_{mlj} \end{cases}. \quad (4)$$

Using the weighting vector $\mathbf{W}^{\mathbf{CR}}(C_k)$ of the M customer requirements under each customer $C_k, k = 1, \dots, K$, we are able to derive a matrix $\mathbf{D}(C_k, E_j) = [D_{nl}(C_k, E_j)]_{N \times N}$ of fuzzy preference relations in $\mathcal{DR} \times \mathcal{DR}$ for a combination of each customer and each design team member, where

$$D_{nl}(C_k, E_j) = \sum_{m=1}^M W_m^{\mathbf{CR}}(C_k) \cdot D_{nl}(CR_m, E_j). \quad (5)$$

Obviously, there will be $K \times J$ combinations of fuzzy preference relations in $\mathcal{DR} \times \mathcal{DR}$. The matrix $\mathbf{D}(C_k, E_j)$ represents the matrix of fuzzy preference relations

in $\mathcal{DR} \times \mathcal{DR}$ under the combination of customer C_k and design team member E_j , where $k = 1, 2, \dots, K$ and $j = 1, 2, \dots, J$. Such a formulation is motivated by [12]. By this way, our approach takes into account the group behaviors of both customers and QFD design team members. In the next section, we will consider how to synthesize individual matrices of fuzzy preference relations into an overall one.

3.2 Group Aggregations of Individual Fuzzy Preference Relations

The $K \times J$ combinations of different customers and different design team members can be viewed as combined information sources in our QFD context. Each combination produces an individual matrix of fuzzy preference relations for the design requirements. Consequently, the uncertain QFD transforms to a group decision-making problem, which needs to synthesize the individual matrices $\mathbf{D}(C_k, E_j)$, $k = 1, \dots, K, j = 1, \dots, J$ into an overall representative matrix $\mathbf{D} = [D_{nl}]_{N \times N}$ in $\mathcal{DR} \times \mathcal{DR}$.

Note that the customers and the design team are assigned importance weighting vectors as $(\gamma_1, \gamma_2, \dots, \gamma_K)$ and $(\beta_1, \beta_2, \dots, \beta_J)$, respectively. Therefore, each preferential combination of a customer and a QFD design team member can be associated with an importance weight P . For the sake of convenience, let us re-denote the individual matrix $\mathbf{D}(C_k, E_j)$ of fuzzy preference relations of each preferential combination in $\mathcal{DR} \times \mathcal{DR}$ as \mathbf{D}^i with its associated importance weight as

$$P_i = \gamma_k \cdot \beta_j, \quad (6)$$

where $k = 1, 2, \dots, K, j = 1, 2, \dots, J, i = 1, 2, \dots, K \times J$. Since $\sum_k^K \gamma_k = 1$ and $\sum_j^J \beta_j = 1$, it is obvious that $\sum_i^{K \times J} P_i = 1$. In order to synthesize the individual matrix \mathbf{D}^i of fuzzy preference relations into an overall representative matrix in $\mathcal{DR} \times \mathcal{DR}$, one commonly used way is to apply the weighted average (WA) method with the following value function

$$\begin{aligned} D_{nl} &= \mathcal{F}_{\text{WA}}(D_{nl}^1, D_{nl}^2, \dots, D_{nl}^{K \times J}) \\ &= \sum_{i=1}^{K \times J} P_i \cdot D_{nl}^i \end{aligned} \quad (7)$$

where $n, l = 1, 2, \dots, N$. The importance weight P_i associated with each combined information source in fact reflects its reliability, i.e., each combined information source has an attached weight that measures its reliability.

We also want to measure the importance of a value D_{nl}^i (in relation to other values) with independence of each information source that has captured it. A basic element underlying group decision-making is the concept of a *majority*, that is a solution is accepted by most of its members since in practice it is quite difficult for the solution to be accepted by all. A natural line of reasoning is to somehow make that strict concept of majority closer to its real human perception by making it more vague, called *fuzzy majority*. A natural manifestation of such a “soft” majority is the so-called linguistic quantifiers as, e.g., most, almost all, much more than half, etc. [17] suggested a formal representation of

these linguistic quantifiers using fuzzy sets [15], i.e., any relative quantifier can be expressed as a fuzzy subset Q of the unit interval $[0, 1]$. In this representation for any proportion $x \in [0, 1]$, $Q(x)$ indicates the degree to which x satisfies the concept conveyed by the term Q . [13] further defined a Regular Increasing Monotone (RIM) quantifier to represent the linguistic quantifier, defined as follows.

Definition 1. A fuzzy subset Q of the real line is called a Regular Increasing Monotone (RIM) quantifier if $Q(0) = 0$, $Q(1) = 1$, and $Q(x) \geq Q(y)$ for $x \geq y$.

Examples of this kind of quantifier are all, most, many, at least α . A quantifier's membership function is often determined by intuition. For example, the following membership function for RIM quantifier has been widely used [14].

$$Q(x) = \begin{cases} 0, & \text{if } x < a \\ \frac{x-a}{b-a}, & \text{if } a \leq x \leq b \\ 1, & \text{if } x > b \end{cases}$$

Using the RIM quantifiers, a linguistically quantified statement for our group aggregations can be written as “ Q information sources are convinced”, which may be exemplified by “most (Q) of information sources are convinced”. Fortunately, Yager [14] proposed a special class of aggregation operators, called ordered weighted averaging (OWA for short) operators, which seem to provide an even better and general aggregation in the sense of being able to simply and uniformly model a large class of fuzzy linguistic quantifiers.

Definition 2. Let (D_1, D_2, \dots, D_N) be a set of values, an OWA operator of dimension N is a mapping $\mathcal{F}_{\text{OWA}} : \mathbb{R}^N \rightarrow \mathbb{R}$ if associated with \mathcal{F} is a weighting vector $\mathbf{W} = (W_1, \dots, W_N)$ such that: $W_n \in [0, 1]$, $\sum_{n=1}^N W_n = 1$, and

$$\mathcal{F}_{\text{OWA}}(D_1, D_2, \dots, D_N) = \sum_{n=1}^N W_n \cdot D_{\sigma(n)}, \quad (8)$$

where $\{D_{\sigma(1)}, D_{\sigma(2)}, \dots, D_{\sigma(N)}\}$ is a permutation of D_1, D_2, \dots, D_N such that $D_{\sigma(n-1)} \geq D_{\sigma(n)}$ for all $n = 2, \dots, N$. (i.e., $D_{\sigma(n)}$ is the n th largest element in the collection (D_1, D_2, \dots, D_N)). The weighting vector \mathbf{W} is derived from the RIM quantifier Q .

Yager [13] proposed a method for obtaining the OWA weighting vector via linguistic quantifiers, especially the RIM quantifiers, which can provide information aggregation procedures guided by verbally expressed concepts and a dimension independent description of the desired aggregation. By using the OWA operator and a RIM quantifier Q , the overall matrix of fuzzy preference relations of different DRs is derived by

$$\begin{aligned} D_{nl} &= \mathcal{F}_{\text{OWA}}(D_{nl}^1, D_{nl}^2, \dots, D_{nl}^{K \times J}) \\ &= \sum_{i=1}^{K \times J} W_i \cdot D_{nl}^{\sigma(i)} \\ &= \sum_{i=1}^{K \times J} \left[Q\left(\frac{i}{K \times J}\right) - Q\left(\frac{i-1}{K \times J}\right) \right] \cdot D_{nl}^{\sigma(i)} \end{aligned} \quad (9)$$

where $\{D_{nl}^{\sigma(1)}, D_{nl}^{\sigma(2)}, \dots, D_{nl}^{\sigma(K \times J)}\}$ is a permutation of $D_{nl}^1, D_{nl}^2, \dots, D_{nl}^{K \times J}$ such that $D_{nl}^{\sigma(i-1)} \geq D_{nl}^{\sigma(i)}$ for all $i = 2, \dots, K \times J$.

Essentially, we want to synthesize the individual matrices of fuzzy preference relations of different design requirements into an overall representative matrix by taking into account the importance weight of each information source in Eq. (7) and the weight of each value based on the concept of *fuzzy majority* in Eq. (9). In this case, a linguistically quantified statement may be generally written as “ Q important information sources are convinced”, which may be exemplified by “most (Q) of the important information sources are convinced”. Such linguistic quantifiers can be, fortunately enough, dealt with by the weighted ordered weighted averaging (WOWA for short) operator [10], defined as follows.

Definition 3. Let (D_1, D_2, \dots, D_N) be a set of values, \mathbf{P} (importance weights) and \mathbf{W} (value weights) be weighting vectors of dimension N ($\mathbf{P} = (P_1, P_2, \dots, P_N)$, $\mathbf{W} = (W_1, W_2, \dots, W_N)$) such that: (1) $P_n \in [0, 1]$ and $\sum_n^N P_n = 1$; (2) $W_n \in [0, 1]$ and $\sum_n^N W_n = 1$. In this case, a mapping $\mathcal{F}_{\text{WOWA}} : \mathbb{R}^N \rightarrow \mathbb{R}$ is a WOWA operator of dimension N if

$$\mathcal{F}_{\text{WOWA}}(D_1, D_2, \dots, D_N; P_1, P_2, \dots, P_N) = \sum_{n=1}^N \omega_n \cdot D_{\sigma(n)} \quad (10)$$

where $\{D_{\sigma(1)}, D_{\sigma(2)}, \dots, D_{\sigma(N)}\}$ is a permutation of D_1, D_2, \dots, D_N such that $D_{\sigma(n-1)} \geq D_{\sigma(n)}$ for all $n = 2, \dots, N$, i.e., $D_{\sigma(n)}$ is the n th largest element in the collection (D_1, D_2, \dots, D_N) , and the weight ω_n is defined as

$$\omega_n = W^* \left(\sum_{l \leq n} P_{\sigma(l)} \right) - W^* \left(\sum_{l < n} P_{\sigma(l)} \right) \quad (11)$$

with W^* a monotone increasing function that interpolates the points $(i/n, \sum_{l \leq n} P_{\sigma(l)})$ together with the point $(0, 0)$. The value $P_{\sigma(l)}$ means the permutation according to $\{D_{\sigma(1)}, D_{\sigma(2)}, \dots, D_{\sigma(N)}\}$.

When W^* is replaced with a RIM quantifier introduced in Definition 1, then $\omega_n = Q \left(\sum_{l \leq n} P_{\sigma(l)} \right) - Q \left(\sum_{l < n} P_{\sigma(l)} \right)$, $n = 1, \dots, N$, which indicates that the WOWA operator becomes the importance weighted quantifier guided aggregation method [13], i.e., the the importance weighted quantifier guided aggregation method is a special case of the WOWA operator. Using the WOWA operator and RIM quantifiers, the overall matrix of fuzzy preference relations of different DRs is derived by

$$\begin{aligned} D_{nl} &= \mathcal{F}_{\text{WOWA}}(D_{nl}^1, D_{nl}^2, \dots, D_{nl}^{K \times J}; P_1, P_2, \dots, P_{K \times J}) \\ &= \sum_{i=1}^{K \times J} \omega_i \cdot D_{nl}^{\sigma(i)} \\ &= \sum_{i=1}^{K \times J} \left[Q \left(\sum_{l \leq i} P_{\sigma(l)} \right) - Q \left(\sum_{l < i} P_{\sigma(l)} \right) \right] \cdot D_{nl}^{\sigma(i)} \end{aligned} \quad (12)$$

where Q is a RIM quantifier introduced in Definition 1.

3.3 Choice Function

We will use two quantifier-guided choice degrees of design requirements: a dominance degree and a non-dominance degree. In particular,

- let Φ_n^+ be the dominant degree which is a measure that design requirement DR_n is dominating the other design requirements, referred to as *leaving flow* in the terminology of decision-making;
- let Φ_n^- be the non-dominant degree which is a measure that design requirement DR_n is dominated by the remaining design requirements, referred to as *entering flow* in the terminology of decision-making.

Here, Φ_m^+ and Φ_m^- can be defined by the following formulas, respectively:

$$\begin{aligned}\Phi_n^+ &= \mathcal{F}_{\text{OWA}}(D_{nl}, l = 1, \dots, N, n \neq l) \\ \Phi_n^- &= \mathcal{F}_{\text{OWA}}(D_{ln}, l = 1, \dots, N, n \neq l)\end{aligned}\tag{13}$$

In addition, let Φ_n be the relative dominant degree which measures the difference between dominant degree and non-dominant degree of design requirement DR_n over the remaining design requirements such that $\Phi_n = \Phi_n^+ - \Phi_n^-, n = 1, 2, \dots, N$. Φ_n is referred to as *net flow* in the terminology of decision-making. The greater Φ_n is, the higher priority design requirement DR_n will be. When Φ_n is a positive value, DR_n is important; when Φ_n is a negative value, DR_n is unimportant. Therefore, according to net flows $\{\Phi_1, \Phi_2, \dots, \Phi_N\}$, we can not only determine a priority order of all the design requirements, but also divide the design requirements into important and unimportant classes, which is important for the decision-makers.

4 A Comparative Application Case Study

In this section, we shall apply our model to prioritize the basic design requirements of a flexible manufacturing system.

4.1 Problem Descriptions

The example discussing the basic design requirements of a flexible manufacturing system (FMS) [3,12] is applied here to illustrate the idea proposed. In that example, eight major CRs are identified to represent the biggest concerns of the customers for the design of an FMS, as shown in Table 1 (indexed by CRs). Based on the design team's experience and expert knowledge, 10 DRs are identified responding to the eight major CRs, as shown in Table 1 (indexed by DRs).

In the QFD planning process, the first step is to determine the weighting vector of customer needs \mathcal{CR} . The linguistic variable for representing the importance of \mathcal{CR} is provided in Eq. (14), indexed by \mathcal{L}^1 . The 10 surveyed customers, $\mathcal{C} = \{C_1, C_2, \dots, C_{10}\}$ were then asked to assess the 8 customer needs by making use of linguistic variable \mathcal{L}^1 . Similar to the CRs, the relationship measure between CRs and DRs was assessed by a QFD design team with three team members, $\mathcal{E} = \{E_1, E_2, E_3\}$. The linguistic variable used for assessing the relationships between CRs and DRs by the design team is presented in Eq. (14), indexed by \mathcal{L}^2 .

Table 1. Customer requirements and design engineering requirements

Customer requirements (CRs)		Design requirements (DRs)	
CR_1	High production volume	DR_1	Automatic gauging
CR_2	Short setup time	DR_2	Tool change system
CR_3	Load-carrying capacity	DR_3	Tool monitoring system
CR_4	User-friendliness	DR_4	Coordinate measuring machine
CR_5	Associated functions	DR_5	Automated guided vehicle
CR_6	Modularity	DR_6	Conveyor
CR_7	Wide tool variety	DR_7	Programmable logic controller (PLC)
CR_8	Wide product variety	DR_8	Storage and retrieval system (S&R system)
		DR_9	Modular fixture
		DR_{10}	Robots

$$\begin{aligned}
\mathcal{L}^I &= \{L_1^I, L_2^I, L_3^I, L_4^I, L_5^I, L_6^I, L_7^I\} \\
&= \{\text{Very unimportant}, \text{Quite unimportant}, \text{Unimportant}, \text{Slightly important}, \\
&\quad \text{Moderately important}, \text{Important}, \text{Very important}\} \\
\mathcal{L}^R &= \{L_1^R, L_2^R, L_3^R, L_4^R\} \\
&= \{\text{Very weak}, \text{Weak}, \text{Moderate}, \text{Strong}\}
\end{aligned} \tag{14}$$

The fuzzy linguistic importance of the 8 CRs assessed by the 10 surveyed customers and the fuzzy linguistic relationship matrix between the 8 customer needs and the 10 design requirements assessed by each of the 3 design team members, are shown in [12], respectively.

4.2 Prioritizing Engineering Design Requirements

First, we calculate a matrix $\mathbf{D}(C_k)$ of fuzzy preference relations in $\mathcal{CR} \times \mathcal{CR}$ under each customer C_k 's linguistic judgment. A weighting vector $\mathbf{W}(C_k)$ can then be derived under each customer C_k 's linguistic judgment according to Eq. (3). The weighting vectors of the 8 customer needs under different customers' judgments are shown in Table 2.

Second, we can obtain a matrix $\mathbf{D}(CR_m, E_j)$ of fuzzy preference relations in $\mathcal{DR} \times \mathcal{DR}$ under each design team member's $E_j, j = 1, 2, 3$ judgment with respect to each customer need $CR_m, m = 1, \dots, 8$. Thirdly, by incorporating the importance weights of the 8 customer needs under different customers, we can derive a matrix $\mathbf{D}(C_k, E_j)$ of the fuzzy preference relations in $\mathcal{DR} \times \mathcal{DR}$ under each combination of a customer C_k and a design team member E_j , where $k = 1, \dots, K, j = 1, \dots, J$. It is obvious that the combinations generate $8 \times 3 = 24$ matrices of fuzzy preference relations of DRs. Taking customer C_1 and design team member E_3 as an example, the derived matrix of fuzzy preference relations in $\mathcal{DR} \times \mathcal{DR}$ is derived as

Table 2. Importance weights of the 8 CRs under each customer

Customers	Importance weights							
	CR_1	CR_2	CR_3	CR_4	CR_5	CR_6	CR_7	CR_8
C_1	0.203125	0.0625	0.203125	0.0625	0.203125	0.109375	0.140625	0.015625
C_2	0.09375	0.09375	0.15625	0.21875	0.015625	0.046875	0.15625	0.21875
C_3	0.046875	0.21875	0.109375	0.15625	0.046875	0.046875	0.15625	0.21875
C_4	0.203125	0.109375	0.203125	0.015625	0.078125	0.140625	0.203125	0.046875
C_5	0.125	0.0625	0.0625	0.1875	0.125	0.015625	0.234375	0.1875
C_6	0.21875	0.140625	0.015625	0.171875	0.109375	0.0625	0.21875	0.0625
C_7	0.234375	0.078125	0.1875	0.03125	0.109375	0.140625	0.03125	0.1875
C_8	0.125	0.203125	0.078125	0.203125	0.203125	0.125	0.03125	0.03125
C_9	0.1875	0.015625	0.109375	0.046875	0.1875	0.078125	0.1875	0.1875
C_{10}	0.078125	0.140625	0.21875	0.078125	0.21875	0.015625	0.171875	0.078125

	DR_1	DR_2	DR_3	DR_4	DR_5	DR_6	DR_7	DR_8	DR_9	DR_{10}
DR_1	0.5	0.41	0.43	0.35	0.42	0.31	0.29	0.36	0.52	0.23
DR_2	0.59	0.5	0.45	0.59	0.66	0.52	0.63	0.48	0.55	0.39
DR_3	0.57	0.55	0.5	0.55	0.55	0.55	0.45	0.43	0.66	0.32
DR_4	0.65	0.41	0.45	0.5	0.48	0.32	0.44	0.48	0.48	0.22
$D(C_1, E_3) =$	DR_5	0.58	0.34	0.45	0.52	0.5	0.52	0.49	0.45	0.6
	DR_6	0.69	0.48	0.45	0.68	0.48	0.5	0.64	0.45	0.59
	DR_7	0.71	0.38	0.55	0.56	0.51	0.36	0.5	0.58	0.41
	DR_8	0.64	0.52	0.57	0.52	0.55	0.55	0.42	0.5	0.66
	DR_9	0.48	0.45	0.34	0.52	0.4	0.41	0.59	0.34	0.5
	DR_{10}	0.77	0.61	0.68	0.78	0.55	0.58	0.69	0.58	0.66

Fourthly, we have to aggregate the individual matrices of the fuzzy preference relations in $\mathcal{DR} \times \mathcal{DR}$ generated by each combination of each customer and each design team member into an overall one. As mentioned previously, each customer and each design team member may have a different importance weight in light of his/her design experience and domain knowledge, respectively. Similar to Wang [12], the importance weights of customers and design team members are assumed as $\gamma_1 = \gamma_2 = \frac{3}{14}, \gamma_3 = \dots = \gamma_{10} = \frac{1}{14}$ and $\beta_1 = \frac{1}{2}, \beta_2 = \frac{1}{3}, \beta_3 = \frac{1}{6}$. In order to incorporate the concept of fuzzy majority to model the group behaviors, the WOWA method is used to perform the group aggregations of individual matrices of fuzzy preference relations into an overall one. The linguistic quantifiers [17] will be used in our aggregation. According to Section 3.3, the choice information with its ranking order is calculated, as shown in Table 3.

It is clear that different priority rankings are obtained with different linguistic quantifiers. There are some interesting observations.

- The priority ranking generated by “there exists” is consistent with the one generated by “for all”.
- The design requirement DR_3 is always the most important one with a positive priority value and the design requirement DR_5 is always the most unimportant one with a negative priority value.

Table 3. Choice values of different design requirements under linguistic quantifiers

Q	Index	Design requirements									
		DR ₁	DR ₂	DR ₃	DR ₄	DR ₅	DR ₆	DR ₇	DR ₈	DR ₉	DR ₁₀
there exists	Φ	0.0	0.09	0.28	0.13	-0.16	-0.07	0.04	0.11	-0.04	-0.08
		DR ₃ ⊳ DR ₄ ⊳ DR ₈ ⊳ DR ₂ ⊳ DR ₇ ⊳ DR ₁ ⊳ DR ₉ ⊳ DR ₆ ⊳ DR ₁₀ ⊳ DR ₅									
for all	Φ	0.0	0.09	0.28	0.13	-0.16	-0.07	0.04	0.11	-0.04	-0.08
		DR ₃ ⊳ DR ₄ ⊳ DR ₈ ⊳ DR ₂ ⊳ DR ₇ ⊳ DR ₁ ⊳ DR ₉ ⊳ DR ₆ ⊳ DR ₁₀ ⊳ DR ₅									
identity	Φ	0.042	0.166	0.384	0.034	-0.365	-0.178	0.058	0.207	-0.185	-0.162
		DR ₃ ⊳ DR ₈ ⊳ DR ₂ ⊳ DR ₇ ⊳ DR ₁ ⊳ DR ₄ ⊳ DR ₁₀ ⊳ DR ₆ ⊳ DR ₉ ⊳ DR ₅									
at least half	Φ	0.109	0.155	0.423	0.073	-0.36	-0.172	0.111	0.168	-0.129	-0.184
		DR ₃ ⊳ DR ₈ ⊳ DR ₂ ⊳ DR ₇ ⊳ DR ₁ ⊳ DR ₄ ⊳ DR ₉ ⊳ DR ₆ ⊳ DR ₁₀ ⊳ DR ₅									
as many as possible	Φ	0.109	0.155	0.423	0.073	-0.359	-0.172	0.111	0.167	-0.129	-0.183
		DR ₃ ⊳ DR ₈ ⊳ DR ₂ ⊳ DR ₇ ⊳ DR ₁ ⊳ DR ₄ ⊳ DR ₉ ⊳ DR ₆ ⊳ DR ₁₀ ⊳ DR ₅									
most	Φ	0.087	0.187	0.415	0.035	-0.375	-0.18	0.095	0.199	-0.187	-0.223
		DR ₃ ⊳ DR ₈ ⊳ DR ₂ ⊳ DR ₇ ⊳ DR ₁ ⊳ DR ₄ ⊳ DR ₆ ⊳ DR ₉ ⊳ DR ₁₀ ⊳ DR ₅									

- The six DRs {DR₁, DR₂, DR₃, DR₄, DR₇, DR₈} are always important ones with positive priority values regardless of the priority rankings of them; whereas, the four DRs {DR₅, DR₆, DR₉, DR₁₀} are always unimportant ones with negative priority values, regardless of the rankings of them.

4.3 Comparative Studies with Two Known Approaches

Our uncertain QFD context has been widely investigated in the literature by quantifying qualitative concepts with fuzzy sets. For example, the linguistic variables in Eq. (14) can be semantically represented by the following function [6,12].

$$\begin{aligned} \mathcal{L}^1 &= \{L_1^1, L_2^1, L_3^1, L_4^1, L_5^1, L_6^1, L_7^1\} \\ &= \{(0, 0, 2), (0, 2, 4), (2, 3.5, 5), (3, 5, 7), (5, 6.5, 8), (6, 8, 10), (8, 10, 10)\} \\ \mathcal{L}^2 &= \{L_1^2, L_2^2, L_3^2, L_4^2\} \\ &= \{(0.1, 0.2, 0.3), (0.3, 0.4, 0.5), (0.5, 0.6, 0.7), (0.7, 0.8, 0.9)\} \end{aligned} \quad (15)$$

where (\cdot, \cdot, \cdot) is used to represent a triangular fuzzy number. Two well-known fuzzy approaches to uncertain QFD are proposed by Chen *et al.* [3] and Wang [12], both of which are based on the nonlinear programming based fuzzy weighted average (NLP-FWA for short) method, introduced as follows.

In their seminal work of fuzzy linguistic QFD, Chen *et al.* [3] proposed a method by means of the NLP-FWA method and fuzzy expected value operator [7] to prioritize DRs. In particular, their approach can be summarized as follows.

1. First, fuzzy importance weights of CRs and fuzzy relationships between CRs and DRs provided by multiple customers and QFD team members are averaged.
2. Second, the NLP-FWA method is used to determine the fuzzy weights of DRs.

3. Finally, the fuzzy weights of DRs are defuzzified using the fuzzy expected value operator for prioritizing DRs.

Wang [12] proposed a different fuzzy group decision-making procedure for prioritizing DRs under uncertainty. The proposed approach does not aggregate the individual judgments of customers and QFD design team members, but rather aggregates the technical importance ratings of DRs, which can be summarized as follows.

1. The NLP-FWA method is used to determine the fuzzy weights of DRs with respect to each customer and design team member.
2. The NLP-FWA method is used to determine the overall fuzzy weights of DRs involving different combinations of any customer and any design team member.
3. The fuzzy weights of DRs are defuzzified using the centroid method [11] for prioritizing DRs.

With the importance weights of customers and design team members, the prioritization results by Chen *et al.*'s, and Wang's approaches are shown as

- Chen *et al.*: $DR_3 \succ DR_8 \succ DR_2 \succ \underbrace{DR_1 \succ DR_7}_{DR_1 \succ DR_7} \succ DR_4 \succ DR_{10} \succ DR_6 \succ DR_9 \succ DR_5$.
- Wang: $DR_3 \succ DR_8 \succ DR_2 \succ \underbrace{DR_7 \succ DR_1}_{DR_7 \succ DR_1} \succ DR_4 \succ DR_{10} \succ DR_6 \succ DR_9 \succ DR_5$.

It is obvious that the priority ranking generated by Chen *et al.* is slightly different from the one generated by Wang in terms of the ranking between DR_1 and DR_7 , the main reason comes from the group behaviors of QFD. Both our approach and Wang have incorporated the group behavior of QFD; whereas, Chen *et al.* have ignored such a phenomenon. With different linguistic quantifiers, different priority rankings may be obtained, which is missed in Wang's work.

5 Conclusion

It is a crucial step to derive the priority order of DRs from CRs in QFD. However, it is not straightforward to prioritize DRs due to two types of uncertainties: human subjective perception and user variability. To address the two uncertainties simultaneously in prioritizing DRs, a novel group decision-making method was proposed in this paper. First, the order-based semantics of linguistic information was used to derive the individual matrix of fuzzy preference relations with respect to each customer and each design team member. Second, the weighted OWA method was used to synthesize the individual matrix of fuzzy preference relations into an overall one. Thirdly, a quantifier-guided choice approach was developed to prioritize the DRs with a classification. A sample FMS design was used to illustrate the proposed approach.

In summary, our model can eliminate the burden of quantifying the qualitative concepts and capture the group behaviors of uncertain QFD. Moreover, since

the quantifier-guided net flow score procedure is used to prioritize DRs with a classification of DRs into positive and negative ones, our model will provide a better decision-support the decision-maker.

Acknowledgement. This study was supported by the National Natural Sciences Foundation of China (NSFC) under Grants #71101050, #71125002, sponsored by Shanghai Pujiang Program, and supported by the Fundamental Research Funds for the Central Universities in China under Grant #WN1123002.

References

1. Chen, L.H., Ko, W.C., Tseng, C.Y.: Fuzzy approaches for constructing house of quality in QFD and its applications: A group decision-making method. *IEEE Transactions on Engineering Management* (2012), doi:10.1109/TEM.2012.2204063
2. Chen, L.H., Weng, M.C.: An evaluation approach to engineering design in QFD processes using fuzzy goal programming models. *Eur. J. Oper. Res.* 172, 230–248 (2006)
3. Chen, Y., Fung, R.Y., Tang, J.: Rating technical attributes in fuzzy QFD by integrating fuzzy weighted average method and fuzzy expected value operator. *Eur. J. Oper. Res.* 174(3), 1553–1566 (2006)
4. Han, C.H., Kim, J.K., Choi, S.H.: Prioritizing engineering characteristics in quality function deployment with incomplete information: A linear partial ordering approach. *Int. J. Prod. Econ.* 91, 235–249 (2004)
5. Herrera, F., Martínez, L.: A 2-tuple fuzzy linguistic representation model for computing with words. *IEEE Trans. Fuzzy Syst.* 8(6), 746–752 (2000)
6. Kwong, C.K., Ye, Y., Chen, Y., Choy, K.L.: A novel fuzzy group decision making approach to prioritising engineering characteristics in QFD under uncertainties. *Int. J. Prod. Res.* 49(19), 5801–5820 (2011)
7. Liu, B., Liu, K.L.: Expected value of fuzzy variable and fuzzy expected value models. *IEEE Trans. Fuzzy Syst.* 10(4), 445–450 (2002)
8. Liu, C.H., Wu, H.H.: A fuzzy group decision-making approach in quality function deployment. *Quality and Quantity* 42(4), 527–540 (2008)
9. Shen, X.X., Tan, K.C., Xie, M.: The implementation of quality function deployment based on linguistic data. *J. Intell. Manuf.* 12(1), 65–75 (2001)
10. Torra, V.: The weighted OWA operator. *Int. J. Intell. Syst.* 12, 153–166 (1997)
11. Wang, Y.M.: Centroid defuzzification and the maximizing set and minimizing set ranking based on alpha level sets. *Comput. Ind. Eng.* 57(1), 228–236 (2009)
12. Wang, Y.M.: A fuzzy-normalisation-based group decision making approach for prioritising engineering design requirements in QFD under uncertainty. *Int. J. Prod. Res.* 50(23), 6963–6977 (2012)
13. Yager, R.R.: Quantifier guided aggregation using OWA operators. *Int. J. Intell. Syst.* 11, 49–73 (1996)
14. Yager, R.R.: On ordered weighted averaging operators in multi-criteria decision making. *IEEE Trans. Syst. Man Cybern.* 18(1), 183–190 (1988)
15. Zadeh, L.A.: Fuzzy sets. *Inform. Contr.* 8(3), 338–353 (1965)
16. Zadeh, L.A.: The concept of a linguistic variable and its application to approximate reasoning—Part I. *Inform. Sci.* 8(3), 199–249 (1975)
17. Zadeh, L.: A computational approach to fuzzy quantifiers in natural languages. *Comput. Math. Appl.* 9, 149–184 (1983)

New Propagation Algorithm in Dynamic Directed Evidential Networks with Conditional Belief Functions

Wafa Laâmari¹, Boutheina Ben Yaghlane¹, and Christophe Simon²

¹ LARODEC Laboratory - Institut Supérieur de Gestion de Tunis, Tunisia

² CRAN - Université de Lorraine - CNRS, UMR 7039, France

Abstract. Proposed as a subclass of directed evidential network with conditional belief functions (DEVN), dynamic directed evidential network with conditional belief functions (DDEVN) was introduced as a new approach for modeling systems evolving in time. Considered as an alternative to dynamic Bayesian network and dynamic possibilistic network, this framework enables to reason under uncertainty expressed in the belief function formalism. In this paper, we propose a new propagation algorithm in DDEVNs based on a new computational structure, namely the mixed binary join tree, which is appropriate for making the exact inference in these networks.

1 Introduction

Several famous frameworks have been developed in artificial intelligence for reasoning under uncertainty characterizing the real world domains. Such domains are also characterized by vast amounts of information that rapidly change over time. The belief function framework [7], also referred to as evidence framework or Dempster-Shafer framework, is a general mathematical framework for managing various forms of uncertainty and representing in a very flexible way the full knowledge, the partial ignorance and the total ignorance.

Based on the belief function theory, dynamic evidential networks represent an efficient tool for handling changes over time and modeling the uncertainty in the existing knowledge. Two principal dynamic evidential graphical models have been proposed, in the literature, for handling the temporal dimension with the belief function formalism: Dynamic evidential networks (DENs) [12] which extend static evidential networks (ENs) [10] and dynamic directed evidential networks with conditional belief functions (DDEVNs) [3] which are an extension of directed evidential networks with conditional belief functions (DEVNs) [1].

Using an extension of the Bayes' theorem to the representation of the Dempster-Shafer's belief function theory, DENs do not fully exploit the abilities of the evidential formalism. Unlike DENs, DDEVNs are based solely on formal rules of the belief function theory [3].

In this paper, we are interested in exact inference in DDEVN. More precisely, we present a new algorithm which permits an accurate inference in DDEVNs contrary to the inference algorithm presented in [3].

The remainder of the paper is organized as follows. In section 2, we present the basic operations in the evidence theory, then we survey in section 3 the architectures proposed to date for exact inference in the evidential networks called the binary join tree (BJT) [8] and the modified binary join tree (MBJT) [1].

In section 4, we overview the DDEVN and some of its basic concepts. Section 5 introduces our algorithm for inference in DDEVNs and present a modified version for constructing the MBJT leading to a new computational structure that we call mixed binary join tree (MixBJT). Section 6 is devoted to a short example illustrating the evidential temporal reasoning in the DDEVNs using the proposed algorithm. We conclude our discussion of future work in section 7.

2 Basic Operations in the Evidence Theory

Let $U = \{A, B, C, D, \dots\}$ be a finite set of variables, where each variable ϑ in U is associated with a finite set of all its possible values called the frame of discernment Θ_ϑ . Let X and Y be two disjoint subsets of U . Their frames are the Cartesian product of the frames for the variables they include. The product space $\Theta_X \times \Theta_Y$ is shortly denoted by $X \times Y$ and a basic belief assignment (bba) m defined on the product space $X \times Y$ is shortly denoted by m^{XY} .

2.1 Marginalization

Let ψ_1 be a potential (valuation)¹ with the associated bba m^{XY} . The marginalization of m^{XY} to X produces a potential ψ_2 with the associated bba m^X which is defined as follows:

$$m^{XY \downarrow X}(A) = \sum_{B \subseteq (X \times Y), B \downarrow X = A} m^{XY}(B) \quad (1)$$

where $B \downarrow X$ is the projection of $B \subseteq X \times Y$ to X obtained by dropping extra coordinates in each element of B .

2.2 Combination

Given two potentials ψ_1 and ψ_2 with their associated bba's m_1^X and m_2^X , the combination of these two potentials produces a new potential $\psi_{12} = \psi_1 \otimes \psi_2$ with an unnormalized bba m^X as follows:

$$m^X(A) = \sum_{B \cap C = A} m_1^X(B) \cdot m_2^X(C) \quad (2)$$

¹ In the evidence theory, beliefs are represented by mass functions called potentials.

2.3 Ballooning Extension

Let $m^X[y_i](x)$ be a conditional belief mass distribution defined on X for $y_i \in Y$. Its ballooning extension denoted by $m^{X \uparrow XY}$ is defined on $X \times Y$ as follows:

$$m^{X \uparrow XY}(\omega) = \begin{cases} m^X[y_i](x) & \text{if } \omega = (x, y_i) \cup (X, \overline{y_i}), \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

3 Architectures for Inference in Evidential Networks

Binary join trees (BJTs) and modified binary join trees (MBJTs) are two frameworks for local computations on evidential graphical models.

3.1 Binary Join Tree

The BJT structure was introduced by Shenoy in [8] as a significant improvement of the Shenoy-Shafer (SS) architecture [9], called also the join tree. The BJT construction algorithm compiles a directed acyclic graph (DAG) G into a tree.

Binary Join Tree Construction. To get a binary join tree, Shenoy's fusion algorithm [8] which is considered as a recursive variable elimination is applied using an elimination sequence S specifying the order in which the variables must be eliminated and a hypergraph H containing subsets of variables in G .

Let $V = \{X_1, X_2, \dots, X_n\}$ denote the set of nodes in G . The hypergraph H is formally defined by the following equation:

$$H = \{\{X_1 \cup Pa(X_1)\}, \{X_2 \cup Pa(X_2)\}, \dots, \{X_n \cup Pa(X_n)\}\}. \quad (4)$$

where $Pa(X_i)$ are the parent nodes of X_i .

Based on H which is obtained by the equation (4) and a specified elimination sequence $S = \vartheta_1 \vartheta_2 \dots \vartheta_n : \vartheta_i \in V$, the BJT is constructed step-by-step by eliminating one variable after another.

Let $\Psi_0 = \{\psi_1, \psi_2, \dots, \psi_n\}$ be the set of given potentials. There exists one to one mapping between subsets Sub_i in H and potentials in Ψ_0 . This means that every potential ψ_i in Ψ_0 is defined exactly on a domain $D(\psi_i) = Sub_i$ in H .

Let Ψ_R denote the potentials that have not yet been combined. At the beginning of the process $\Psi_R = \Psi_0$. To eliminate the variable ϑ_1 , the fusion algorithm combines all the potentials of Ψ_R which contain ϑ_1 in their domains on a binary basis, i.e. the combination operations are done two at a time [8].

We define $\Psi_1 = \{\psi_i \in \Psi_R : \vartheta_1 \in D(\psi_i)\}$ as the set of the potentials having ϑ_1 in their domains. At each time, two potentials ψ_i and ψ_j with the smallest domains are selected from Ψ_1 to be combined. Combining ψ_i and ψ_j by equation (2) leads to a new intermediate potential $\psi_i \otimes \psi_j$ defined on a new domain $D(\psi_i \otimes \psi_j)$ and produces a partial BJT with nodes labeled as $D(\psi_i)$, $D(\psi_j)$ and $D(\psi_i \otimes \psi_j)$. The intermediate potential $\psi_i \otimes \psi_j$ is then added to $\Psi_1 - \{\psi_i, \psi_j\}$. Combinations of potentials that remain in Ψ_1 are repeated recursively in this binary way and new nodes are added to the BJT, till we obtain the potential $\phi_1 = \otimes\{\psi_i \in \Psi_1\}$ with a corresponding node labeled as $D(\phi_1)$.

ϑ_1 is then eliminated from the domain $D(\phi_1)$ of the potential ϕ_1 . As a result, a new potential $\psi_{n+1} = \phi_1^{D(\phi_1) \downarrow (D(\phi_1) - \{\vartheta_1\})}$ is obtained by the marginalization of ϕ_1 on the new domain $D(\psi_{n+1}) = D(\phi_1) - \{\vartheta_1\}$ using equation (1), and a new node is added to the BJT for ψ_{n+1} . Ψ_R can be now updated to have only the remaining potentials whose domains do not contain the eliminated variable ϑ_1 and a new set $\Psi_R = (\Psi_R - \{\psi_i \in \Psi_1\}) \cup \{\psi_{n+1}\}$ is obtained. Note that the new set Ψ_R will be used to select all the potentials containing ϑ_2 in their domains and the same procedure is repeated to eliminate the second variable ϑ_2 .

To summarize, at each step k of the binary fusion algorithm, we are particular interested in eliminating the k^{th} variable in the elimination sequence S . The algorithm creates a new set Ψ_k by selecting from Ψ_R the potentials which have ϑ_k in their domains. Eliminating ϑ_k creates a partial binary join tree and leads to a new set $\Psi_R = (\Psi_R - \{\psi_i \in \Psi_k\}) \cup \{\psi_{n+k}\}$ containing the potentials that remain after the elimination of ϑ_k , where $\psi_{n+k} = \phi_k^{D(\phi_k) \downarrow (D(\phi_k) - \{\vartheta_k\})}$ and $\phi_k = \otimes \{\psi_i \in \Psi_k\}$. After eliminating ϑ_n at the end of the process, only one potential remains whose domain is the empty set.

Example 1. Let us consider the directed acyclic graph $G: A \rightarrow C \leftarrow B$ and let us assume that ψ_1 , ψ_2 and ψ_3 are three potentials with the corresponding domains $D(\psi_1) = \{A\}$, $D(\psi_2) = \{B\}$ and $D(\psi_3) = \{A,C,B\}$. The BJT constructed by the fusion algorithm for the elimination sequence ACB is shown in Fig. 1 (a). Another BJT constructed according to the elimination sequence BAC is shown in Fig. 1 (b). Clearly, the binary join tree structure depends strongly on the elimination sequence used when applying the fusion algorithm. Using different elimination sequences leads to different BJTs with different nodes. For instance, $\{C,B\}$ is a node in the first BJT while it is not in the second one. This is explained by the fact that when changing the order in which the variables are eliminated, the combination operations performed do not give the same intermediate results.

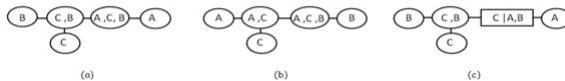


Fig. 1. Binary join trees (a,b) and Modified binary join tree (c)

3.2 Modified Binary Join Tree

The MBJT [1] is an adaptation of the BJT presented previously. It was proposed to avoid the loss of information about the (in)dependencies among the variables arising when transforming the original evidential network into a BJT.

Modified Binary Join Tree Construction. The MBJT construction process is similar to that of the BJT with some modifications used for showing explicitly the conditional relationships between variables in the obtained tree. Rather than using joint nodes (i.e circular nodes) containing just the set of these variables like in the BJT, the MBJT integrates conditional nodes (i.e. rectangular nodes)

emphasizing the conditional relationships. Using conditional nodes in the MBJT for representing the conditional (in)dependencies is computationally more efficient than using joint nodes in the corresponding BJT [1].

Example 2. Considering the same DAG of the previous example, the MBJT obtained for the elimination sequence ACB is shown in Fig. 1 (c). The conditional node $C|A,B$ in this MBJT shows explicitly that A and B are the parent nodes of C. The joint node A,C,B in the corresponding BJT constructed using the same elimination sequence (Fig. 1 (a)) shows only these variables without maintaining the relationship between them.

4 Dynamic Directed Evidential Network with Conditional Belief Functions Representation (DDEVN)

On the basis of the relationships specified between variables in time, dynamic directed evidential networks with conditional belief functions (DDEVNs) allow to compactly model systems or processes that evolve over time [3]. Unfortunately, the number of time slices in these systems can not be known in advance, and as a consequence the number of variables is potentially infinite and the entire unrolled model quickly becomes intractably large. To deal with this intractability, two assumptions are typically made in the DDEVNs: the Markov assumption and the stationary assumption [3]. The first assumption means that every node at a particular time slice t depends only on its parent nodes in the same time slice t and in the immediately previous time slice $t - 1$. Whereas the second one means that the transition-belief mass distributions representing the temporal dependencies between the variables do not depend on the time slice.

Under these two assumptions, systems can be simply modeled with a DDEVN unrolled slice by slice. This DDEVN is defined in a concise way by two evidential networks D_0 and D_t , where:

1. D_0 is an evidential network that represents the joint mass distribution over variables of the initial time slice $t = 0$.
2. D_t is an evidential network made up of only two consecutive time indexed slices $t - 1$ and t , namely a 2-time slices DEVN (2-TDEVN). If the state of a variable X_{t-1} at the time slice $t - 1$ directly affects the state of a variable Y_t at the time slice t , then this temporal dependency between the two variables is qualitatively represented by a directed arc connecting them. This arc denotes a transition-belief mass distribution which quantifies how Y_t depends on X_{t-1} .

Definition 1. The incoming interface $\text{in-}I_{t-1}$ is the set of nodes of the last time slice $t - 1$ which have one child node or more in the current time slice t . The outgoing interface $\text{out-}I_t$ is the set of nodes of the current time slice t which have one child node or more in the next time slice $t + 1$.

More formally, the in- I_{t-1} and the out- I_t are defined in the DDEVN as follows:

$$\text{in-}I_{t-1} = \{Z_{t-1} \in N_{t-1} | (Z_{t-1}, X_t) \in E(t-1, t) \text{ and } X_t \in N_t\} . \quad (5)$$

$$\text{out-}I_t = \{X_t \in N_t | (X_t, Y_{t+1}) \in E(t, t+1) \text{ and } Y_{t+1} \in N_{t+1}\} . \quad (6)$$

where N_{t-1} , N_t and N_{t+1} are the set of nodes modeling respectively the time slices $t-1$, t , and $t+1$ and $E(t, t+1)$ is the set of edges between the two successive time slices t and $t+1$.

Definition 2. A 1.5 DDEVN unrolled for time slices $(t-1, t)$ is a graph which consists of all the nodes of the current time slice t and all the incoming interface nodes from the preceding time slice $t-1$, i.e. the in- I_{t-1} . It results from a 2-TDEVN by the elimination of nodes which belong to the time slice $t-1$ but not to the incoming interface in- I_{t-1} .

According to *Definition 1* and *Definition 2*, the 1.5 DDEVN unrolled for time slices $(t-1, t)$ emphasizes the in- I_{t-1} and the out- I_t interfaces. The in- I_{t-1} allows to take into account the effect of the time slice $t-1$ on the current time slice t , and the out- I_t allows to summarize the information about variables in the current time slice t influencing the future time slice $t+1$.

4.1 Inference Algorithm in DDEVN

When modeling a system that evolves over time by a DDEVN, we are interested in knowing the state of each variable in the system at a time slice $t = T$. More precisely, in front of an observed situation at the initial time slice $t = 0$ and based on transition distributions illustrating the temporal dependencies between two consecutive time slices, the aim of the inference algorithm is to compute the belief mass distribution of each variable X at any time step.

The Markov assumption of DDEVNs justifies the fact that the future time slice is conditionally independent of the past ones given the present time slice. Therefore, the outgoing interface out- I_t d-separates the past from the future, and encapsulates the information about previous time slices (time slices before the time t) which is the only necessary information to perform propagation in next time slices (time slice $t+1$ and later) [3].

The algorithm proposed in [3] for propagation in DDEVN creates a MBJT M_0 for the initial time slice and another one, called M_t , for the corresponding unrolled 1.5 DDEVN. This algorithm approximates the joint distribution α over the interface as the product of the marginals α_i of the involved variables. More precisely, both the outgoing interface and the incoming interface are represented in the MBJTs by many nodes. Therefore, to advance propagation for example from time slice t to time slice $t+1$, marginals α_i of variables in the out- I_t are computed in nodes representing the out- I_t in the 1.5 DDEVN unrolled for time slices $(t-1, t)$, then they are multiplied onto the nodes representing the in- I_t in the 1.5 DDEVN unrolled for time slices $(t, t+1)$.

5 New Inference Algorithm in Dynamic Directed Evidential Networks with Conditional Belief Functions

The accuracy of the inference algorithm in dynamic models depends on the number of nodes used to represent the interfaces. The exact interface algorithms proposed for reasoning in dynamic probabilistic and dynamic possibilistic networks require the use of a single node for the outgoing interface and a single node for the incoming interface in the computational structures [5,2]. Using more than one node sacrifice intuitively some accuracy for speed.

The algorithm proposed in [3] for inference in DDEVNs uses many nodes for representing the interfaces. Therefore, it corresponds to approximate inference algorithm since the requirement that all variables in the interface need to be in a single node is dropped.

In order to accurately perform the propagation process in DDEVNs, we propose in this section a new algorithm for exact inference in these networks.

The main principle underlying this algorithm is the use of the BJT structure with necessary modifications to perform the exact inference. We refer to the obtained tree structure as mixed binary join tree. Like the algorithm that we have proposed in [3,4] for inference in the DDEVNs based on MBJTs, the one proposed in this paper builds also using the fusion algorithm a tree structure that will be used to propagate messages from one node to another. However, rather than using several nodes for representing the outgoing interface and several nodes for representing the incoming interface, this algorithm imposes the restriction that all the variables in the outgoing interface (respectively all the variables in the incoming interface) must belong to one node in the obtained tree structure to enable the application of the exact inference.

The basic idea of our algorithm is :

- ✓ to create for the initial time slice $t = 0$ of the DDEVN a mixed binary join tree structure Mix_0 in which variables of the out- I_0 belong to a single node that we call N_0 . N_0 encapsulates the sufficient information needed for propagation in the next time slice $t = 1$. Mix_0 is used to run belief propagation in the DDEVN at the time slice $t = 0$.

- ✓ to create for the 1.5 DDEVN a mixed binary join tree structure Mix_t in which there are two nodes representing the in- I_{t-1} and the out- I_t . We respectively refer to these nodes as N_{t-1} and N_t . When advancing the time in the DDEVN from $t - 1$ to t , N_{t-1} allows to take into account the effect of the time slice $t - 1$ on the current time slice t , whereas N_t encapsulates the sufficient information needed in the next time slice $t + 1$. Mix_t is used for inference in the DDEVN at each time slice $t \geq 1$.

5.1 From DDEVN to Mixed Binary Join Tree (MixBJT)

To transform a DDEVN into a BJT or a MBJT, Shenoy's fusion algorithm presented in section 3.1 eliminates variables in the model one after another.

Example 1 has shown that based on two different elimination sequences, the fusion algorithm creates for a DAG two different BJT_s with nodes having different labels or domains.

Thus, when applying the fusion algorithm to construct a BJT for a given DDEVN, one can not successfully determine beforehand whether or not a chosen elimination sequence would guarantee to make the out-I₀ a node in the BJT Mix₀. In addition, neither the incoming interface in-I_{t-1} nor the outgoing interface out-I_t are guaranteed to be two nodes in the BJT Mix_t. If, for example our aim were to construct for the DAG of *Example 1* a BJT containing a node having as a label {C, B}, then we could not know in advance that BAC is not the appropriate elimination sequence and that it leads to a BJT which does not contain this node. Nevertheless, it is far from trivial to generate all the possible elimination sequences and to build all the corresponding BJT_s in order to verify which best suits our need.

Since we are sure that in each BJT there is one node for each subset in the hypergraph used to construct it, one way to overcome the difficulty is to add, to the hypergraph, the subsets of variables for which nodes must exist in the resulting tree. More precisely, if we add the out-I₀ as a subset in the hypergraph used to construct Mix₀, this guarantees to obtain in the resulting structure the node N₀ whose label is the domain of the out-I₀. Adding also the in-I_{t-1} and the out-I_t of the 1.5 DDEVN unrolled for time slices (t - 1, t) as subsets in the hypergraph used to build Mix_t, enables the fusion algorithm to generate a tree containing the two nodes N_{t-1} and N_t corresponding respectively to the in-I_{t-1} and the out-I_t.

Therefore, two new hypergraphs H₀ and H_t are used to construct Mix₀ and Mix_t. Given a 2-TDEVN and its corresponding 1.5 DDEVN unrolled for time slices (t - 1, t), we define the hypergraphs H₀ and H_t that we will use for constructing Mix₀ and Mix_t by the following equations:

$$H_0 = \{\{X_0^1 \cup Pa(X_0^1)\}, \{X_0^2 \cup Pa(X_0^2)\}, \dots, \{X_0^n \cup Pa(X_0^n)\}\} \cup \text{out-I}_0 . \quad (7)$$

$$H_t = \{\{X_t^1 \cup Pa(X_t^1)\}, \dots, \{X_t^n \cup Pa(X_t^n)\}\} \cup \{\text{in-I}_{t-1}, \text{out-I}_t\} . \quad (8)$$

where X_tⁱ refers to the i-th variable of the time slice t in the 2-TDEVN.

With the aim of reducing the dimensionality of the belief functions involved in the computation and in order to maintain the (in)dependence relationships between variables of the original DDEVN, we suggest to exploit the structure of the model when generating the mixed BJT_s Mix₀ and Mix_t by using the conditional nodes in the produced tree structures for showing explicitly the conditional relationships between variables. Associated with conditional belief functions, these conditional nodes allow to reduce, like in the MBJT [1], the amount of computations during the propagation process since representing belief functions using the conditional form allows to avoid the computation of joint belief functions on the product space.

Naturally, a question crosses the mind: If conditional nodes which represent the transformations essentially made on the produced BJT_s to obtain the mixed binary join trees Mix₀ and Mix_t, are the same transformations made on BJT_s to

obtain the MBJTs, why are the proposed structures called "mixed binary join trees" ? The following example attempts to answer this question by showing that when there is an independence relationship between variables in the interfaces, the tree structure proposed in this paper for inference in DDEVN is neither a purely binary join tree nor a purely modified binary join tree.

Example 3. Let us consider the 2-TDEVN shown in Fig. 2(a) and its corresponding 1.5 DDEVN given in Fig. 2(b). The outgoing interface out- I_0 for the time slice $t = 0$ is represented by the subset $\{A_0, C_0\}$. When constructing Mix_0 that represents the initial time slice, it is required to obtain in this tree the node $N_0 = \{A_0, C_0\}$ which encapsulates the exact quantity relative to the joint distribution over variables in the out- I_0 that we will use to perform inference in the time slice $t = 1$. The hypergraph H_0 that we use to satisfy this requirement is obtained using equation 7: $H_0 = \{\{A_0\}, \{B_0, A_0\}, \{C_0, A_0\}, \{E_0, B_0\}\}$. If we apply the fusion algorithm described in section 3.1 for constructing Mix_0 using H_0 and the elimination sequence $C_0 A_0 B_0 E_0$, we obtain the BJT shown in Fig. 2(c) in which variables A_0 and C_0 of the out- I_0 belong to the single node $N_0 = \{A_0, C_0\}$. Now, if we exploit the structure of the DDEVN to represent explicitly in Mix_0 all the conditional dependencies, we obtain the MBJT depicted in Fig. 2(d) which integrates three conditional nodes $B_0|A_0$, $E_0|B_0$ and $C_0|A_0$. The node N_0 which was initially a joint node $\{A_0, C_0\}$ in Fig. 2(c) has become a conditional one in Fig. 2(d) since C_0 is the child node of A_0 in the model.

During the inference through the MBJT [1], conditional nodes are asked neither to send nor to receive messages because they are regarded as bridges between joint nodes: Each conditional node holds a decisive role allowing to determine whether the message sent from a joint node to another joint node on side of this conditional node is a parent message or a child message. For example, when the joint nodes A_0 and C_0 exchange messages between each other in the MBJT of Fig 2.(d), the conditional node $N_0 = C_0|A_0$ which is initialized by means of the conditional belief function $m[A_0](C_0)$ allows us just to reveal that A_0 is the parent node of C_0 . At the end of the belief propagation process, we have marginals in the MBJT for only joint nodes and one can not compute distributions of variables in the conditional nodes since they do not computations.

Thus, despite having in Mix_0 the two variables of the out- I_0 in a single node N_0 , this node does not yield after the propagation process the joint distribution over out- I_0 since it is a conditional node. This problem arises in this case because the DDEVN presents a conditional relationship between variables of the interface. To deal with this problem, the solution is based on the following simple idea: Since our algorithm takes advantage of the structure of the DDEVN by making each node having as a label $X_t^i \cup Pa(X_t^i)$ a conditional node, we should pay attention to the fact that node N_0 in Mix_0 and nodes N_{t-1} and N_t in Mix_t must be maintained as joint nodes even if they represent implicitly a conditional relation over the involved variables. When maintaining nodes representing interfaces as joint nodes, we must distinguish between them and other joint nodes (which do not represent implicitly conditional relationships over the involved variables). For this reason, we propose to qualitatively represent them

by mixed nodes, i.e. rectangles containing a joint relationship between the involved variables. This representation reveals explicitly that there is a conditional dependence between the variables of the mixed node in question, but this node is maintained a joint node when making modifications in an attempt to have the joint distribution over the corresponding interface.

By imposing this new requirement, we guarantee to have at the end of the propagation process the joint distribution needed to advance the inference from time slice to another. Considering the new requirement for example during the construction process of Mix_0 and Mix_t of the DDEVN in Fig. 2(a) leads to the resulting tree structures shown respectively in Fig 2.(e) and Fig 3. The mixed node $N_0 = \{C_0, A_0\}$ will provide at the end of the propagation a joint distribution over the out- I_0 because when making modifications on the tree Mix_0 , this node was maintained a joint node despite the presence of a conditional relationship between C_0 and A_0 . The obtained tree can not be considered as a MBJT since the algorithm for constructing a MBJT imposes to represent all the conditional relationships between variables in the model by the corresponding conditional nodes. We can not also consider the obtained structure as a BJT because this latter contains only joint nodes since the algorithm for constructing a BJT pays no attention to the (in)dependencies relationships.

Since this structure shows explicitly some (in)dependencies relationships in the DDEVN (like MBJT) and pays no heed to others (like BJT), we call it **mixed binary join tree** (MixBJT).

It is noteworthy to mention that the obtained structures Mix_0 and Mix_t collapse into MBJTs when there is no relationship between the variables in the interfaces.

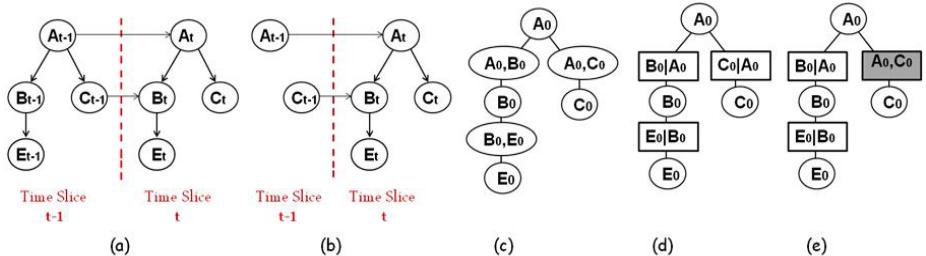


Fig. 2. A 2-TDEVN, a 1.5 DDEVN, a BJT, a MBJT and a Mix_0 for the time slice $t = 0$

Mixed Binary Join Tree Mix_0 Construction. Given a 2-TDEVN and an elimination sequence S_0 , the general construction process of the MixBJT Mix_0 is described by the following steps:

1. Identify the out- I_0 interface in the 2-TDEVN using equation (6).
2. Remove each node belonging to the time slice 1 from the 2-TDEVN (obtain the evidential network D_0 which represents only the time slice $t = 0$).

3. Determine the hypergraph H_0 for the resulting structure D_0 by equation (7).
4. Apply the FA to construct a BJT B_0 for D_0 using S_0 and H_0 .
5. Make each joint node in B_0 (except N_0 which represents the out- I_0) representing an (in)dependence relationship between the involved variables a conditional node.

Mixed Binary Join Tree Mix_t Construction. Given a 2-TDEVN and an elimination sequence S_t , the general construction process of the MixBJT Mix_t is described by the following steps:

1. Identify the in- I_{t-1} and the out- I_t in the 2-TDEVN by equations (5) and (6).
2. Remove each node belonging to the time slice $t - 1$ and not to the in- I_{t-1} from the 2-TDEVN (obtain a 1.5 DDEVN unrolled for time slices $(t-1, t)$).
3. Determine the hypergraph H_t for the resulting structure by equation (8).
4. Apply the FA to construct a BJT B_t for the 1.5 DDEVN using S_t and H_t .
5. Make each joint node in B_t (except N_{t-1} and N_t which represent the in- I_{t-1} and the out- I_t interfaces) representing an (in)dependence relationship between the involved variables a conditional node.

Mixed Binary Join Trees Initialization. The MixBJT Mix₀ is initialized by potentials relative to time slice $t = 0$. Whereas the MixBJT Mix_t is initialized by potentials relative only to time slice t . Composed of both joint and conditional nodes, a MixBJT is initialized, like a MBJT, by means of joint belief functions for not conditional nodes (i.e. joint nodes) and conditional belief functions for conditional ones. Mixed nodes which represent conditional relationships between the involved variables $X_t^i \cup Pa(X_t^i)$ and which we maintain as joint nodes should receive a specific attention when initializing the MixBJTs. Naturally a mixed node must be initialized by a joint belief function $m^{X_t^i \times Pa(X_t^i)}$ since it is a joint node. However we have the conditional belief function $m[Pa(X_t^i)](X_t^i)$ which quantifies the conditional relationship between variables of this mixed node in the DDEVN. One way to solve this problem is to construct the joint belief distribution from the conditional one, i.e. to compute the joint form of the conditional belief function. This is done using the following equation:

- a. Compute using equation (2) the ballooning extension $m[p(X_i)]^{\uparrow X_i \times P(X_i)}$ of $m[p(X_i)]$ for each instance $p(X_i)$ of the parent nodes of X_i denoted $P(X_i)$.
- b. Combine all the ballooning extensions for all the instances using the Dempster's rule of combination.

(9)

The exact inference algorithm in the DDEVN is given by the following algorithm:

Algorithm 1. Exact inference in DDEVNs

Data: Mix₀, Mix_t

Result: Marginal Distributions in the time slice $t = T$

begin

- 1. Performing the propagation in Mix₀^a;
- 2. Out_distr=Joint_distribution(out-I₀);
- 3. **if** $T > 0$ **then**
 - for** $i=1$ **to** T **do**
 - Associate Out_distr to node N_{i-1} representing the in-I_{i-1} in Mix_t;
 - Performing the propagation in Mix_t ;
 - Out_distr= Joint_distribution(out-I_i);
 - end**
- end**
- 4. **if** $T=0$ **then**
 - Compute_Marginals(Mix₀);
- else**
 - Compute_Marginals(Mix_t);
- end**

end

^a The propagation process is performed in the MixBJT as in the MBJT since both of them are composed of joint and conditional nodes. For details, the reader is referred to [1].

6 Illustrative Case Study

For the sake of illustration, let us consider again the evolving-time system with four variables modeled by the 2-TDEVN displayed in Fig. 2(a). Each node X_t in the 2-TDEVN represents the variable X at the time slice t which has two states $(\{x\}, \{\bar{x}\})$. Fig. 2(b) shows the 1.5 DDEVN created from the 2-TDEVN given in Fig. 2(a) by removing all nodes in the time slice $t - 1$ not belonging to I_{t-1} .

Table 1 represents the a priori mass distribution of A at the time step 0. The conditional mass distributions relative to nodes B_t, C_t and E_t are respectively given in Tables 2, 3 and 4. The belief mass distribution of variables A_t and B_t at the time slice t which depend respectively on the distributions of A_{t-1} and C_{t-1} at the time step $t - 1$ are respectively represented in Tables 5 and 6.

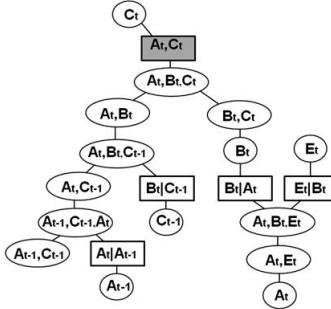
Tables 1, 2, 3 and 4. A priori mass table $M(A_0)$ and Conditional mass tables $M[A_t](B_t)$, $M[A_t](C_t)$ and $M[B_t](E_t)$

$B_t A_t$	a	\bar{a}	$a \cup \bar{a}$	$C_t A_t$	a	\bar{a}	$E_t B_t$	b	\bar{b}	$b \cup \bar{b}$
a	0.7	0.2	0.1	b	0.6	0.7	0.5	c	0.7	0.5
\bar{a}	0.2	0.7	0.9	\bar{b}	0.2	0.2	0.4	\bar{c}	0.2	0.3
$a \cup \bar{a}$	0.1	0.9	1.0	$b \cup \bar{b}$	0.2	0.1	0.3	$c \cup \bar{c}$	0.1	0.2

Tables 5 and 6. Conditional mass tables $M[A_{t-1}](A_t)$ and $M[C_{t-1}](B_t)$

$A_t A_{t-1}$	a	\bar{a}	$a \cup \bar{a}$	$B_t C_{t-1}$	c	\bar{c}	$c \cup \bar{c}$
a	0.4	0.5	0.7	b	0.5	0.6	0.7
\bar{a}	0.4	0.1	0.2	b	0.2	0.3	0.1
$a \cup \bar{a}$	0.2	0.4	0.1	$b \cup b$	0.3	0.1	0.2

Construction and Initialization of Mix_0 and Mix_t . To construct the two MixBJTs Mix_0 and Mix_t , we first define using equations (7) and (8) the hypergraphs H_0 and H_t that respectively characterize the subsets of variables that will be used by the fusion algorithm: $H_0 = \{\{A_0\}, \{B_0, A_0\}, \{C_0, A_0\}, \{E_0, B_0\}\}$ and $H_t = \{\{A_t, A_{t-1}\}, \{B_t, A_t\}, \{B_t, C_{t-1}\}, \{C_t, A_t\}, \{E_t, B_t\}, \{A_{t-1}, C_{t-1}\}\}$. Given these two hypergraphs, the 2-TDEVN in Fig.2 (a) and two elimination sequences S_0 and S_t , Mix_0 and Mix_t are generated using respectively algorithms 1 and 2. For $S_0 = C_0 A_0 B_0 E_0$ and $S_t = A_{t-1} C_{t-1} A_t C_t B_t E_t$, we obtain the MixBJTs Mix_0 and Mix_t depicted respectively in Fig.2 (e) and Fig.3.

**Fig. 3.** The mixed binary join tree Mix_t of the DDEVN in Fig.2 (a)

Using the a priori and the conditional mass tables, Mix_0 and Mix_t are initialized by assigning each belief function distribution to the corresponding node (the fifth step of algorithms 2 and 3). Since we have the conditional belief function $M[A_t](C_t)$, we must compute the joint form of this belief function to initialize the joint rectangular nodes $N_0 = \{A_0, C_0\}$ and $N_t = \{A_t, C_t\}$ in the mixed binary join trees Mix_0 and Mix_t . The first step of computing this joint form consists in computing the ballooning extensions of $m[a]^{C_t \uparrow C_t \times A_t}$ and $m[\bar{a}]^{C_t \uparrow C_t \times A_t}$ using equation (3) as shown in Table 7.

Now that we have computed the ballooning extension of the conditional belief function $M[A_t](C_t)$, we compute the joint form of this conditional belief function by combining the ballooning extensions obtained in Table 7 using Dempster's rule of combination. The resulting joint form of $M[A_t](C_t)$ which will be used to initialize nodes N_0 and N_t is depicted in Table 8.

Table 7. Ballooning extension of the conditional belief function

		$m[A_t]^{C_t \uparrow C_t \times A_t}$
as	$m(\{c, a\}, \{c, \bar{a}\}, \{\bar{c}, \bar{a}\}) = m[a](c) = 0.7$ $m(\{\bar{c}, a\}, \{c, \bar{a}\}, \{\bar{c}, \bar{a}\}) = m[a](\bar{c}) = 0.2$ $m(\{C_t \times A_t\}) = m[a](\{c, \bar{c}\}) = 0.1$	
is	$m(\{c, \bar{a}\}, \{c, a\}, \{\bar{c}, a\}) = m[\bar{a}](c) = 0.5$ $m(\{\bar{c}, \bar{a}\}, \{c, a\}, \{\bar{c}, a\}) = m[\bar{a}](\bar{c}) = 0.3$ $m(\{C_t \times A_t\}) = m[\bar{a}](\{c, \bar{c}\}) = 0.2$	

Table 8. Joint Form of $M[A_t](C_t)$

	$m(\{c, a\}, \{c, \bar{a}\}, \{\bar{c}, \bar{a}\})$ = 0.7	$m(\{\bar{c}, a\}, \{c, \bar{a}\}, \{\bar{c}, \bar{a}\})$ = 0.2	$m(\{C_t \times A_t\})$ = 0.1
$m(\{c, \bar{a}\}, \{c, a\}, \{\bar{c}, a\})$ = 0.5	$m(\{c, \bar{a}\}, \{c, a\})$ = 0.35	$m(\{c, \bar{a}\}, \{\bar{c}, a\})$ = 0.1	$m(\{c, \bar{a}\}, \{c, a\}, \{\bar{c}, a\})$ = 0.05
$m(\{\bar{c}, \bar{a}\}, \{c, a\}, \{\bar{c}, a\})$ = 0.3	$m(\{\bar{c}, \bar{a}\}, \{c, a\})$ = 0.21	$m(\{\bar{c}, \bar{a}\}, \{\bar{c}, a\})$ = 0.06	$m(\{\bar{c}, \bar{a}\}, \{c, a\}, \{\bar{c}, a\})$ = 0.03
$m(\{C_t \times A_t\})$ = 0.2	$m(\{c, a\}, \{c, \bar{a}\}, \{\bar{c}, \bar{a}\})$ = 0.14	$m(\{\bar{c}, a\}, \{c, \bar{a}\}, \{\bar{c}, \bar{a}\})$ = 0.04	$m(\{C_t \times A_t\})$ = 0.02

Once Mix_0 and Mix_t are constructed and initialized, we can compute the marginals of variables at any time step $t = T$ by recursively performing the bidirectional message-passing scheme in Mix_t .

Performing the Propagation Process in the DDEVN. Suppose now that we wish to compute the marginal of the variable E_t at time step $t = 1000$. We first perform the inference process in Mix_0 and we compute the joint distribution over nodes in the outgoing interface $I_0 = \{A_0, C_0\}$ (the first and the second steps of algorithm 3). The joint distribution over the outgoing interface I_0 's nodes will be used when performing the second propagation in the MixBJT Mix_t in the next time slice ($t = 1$). It will be introduced in the in- I_0 of Mix_t . After performing the inference algorithm, Mix_t yields the joint distribution of the out- I_1 (forming the outgoing interface out- I_1) which is the sufficient information needed to continue the propagation in the following time slice $t = 2$.

After carrying out the inference process in the MixBJT Mix_t recursively for 1000 time slices, we obtain the following distribution for node E_{1000} : $M(E_{1000}) = [m(\emptyset) = 0 \ m(\{e\}) = 0.00218 \ m(\{\bar{e}\}) = 0.8525 \ m(\{e, \bar{e}\}) = 0.14525]$.

7 Conclusion

We have presented in this paper an algorithm for inference in DDEVN based on a computational structure, called the mixed binary join tree, which is proposed for making the exact online inference in these networks. In future work, the development of new algorithms to perform faster propagation in the DDEVN will be of a great interest, since exact propagation still a NP-Hard problem when we manipulate huge amount of data.

References

1. Ben Yaghlane, B., Mellouli, K.: Inference in Directed Evidential Networks Based on the Transferable Belief Model. *Int. J. Approx. Reasoning* 48(2), 399–418 (2008)
2. Heni, A., Ben Amor, N., Benferhat, S., Alimi, A.: Dynamic Possibilistic Networks: Representation and Exact Inference. In: IEEE International Conference on Computational Intelligence for Measurement Systems and Applications (CIMSA 2007), Ostuni, Italy, pp. 1–8 (2007)
3. Laâmari, W., Ben Yaghlane, B., Simon, C.: Dynamic Directed Evidential Networks with Conditional Belief Functions: Application to System Reliability. In: Greco, S., Bouchon-Meunier, B., Coletti, G., Fedrizzi, M., Matarazzo, B., Yager, R.R. (eds.) IPMU 2012, Part III. CCIS, vol. 299, pp. 481–490. Springer, Heidelberg (2012)
4. Laâmari, W., Ben Yaghlane, B., Simon, C.: On the Complexity of the Graphical Representation and the Belief Inference in the Dynamic Directed Evidential Networks with Conditional Belief Functions. In: Hüllermeier, E., Link, S., Fober, T., Seeger, B. (eds.) SUM 2012. LNCS, vol. 7520, pp. 206–218. Springer, Heidelberg (2012)
5. Murphy, K.: Dynamic Bayesian Networks: Representation, Inference and Learning. PhD thesis. Dept. Computer Science, UC, Berkeley (2002)
6. Pearl, J.: Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann (1988)
7. Shafer, G.: A Mathematical Theory of Evidence. Princeton University Press, Princeton (1976)
8. Shenoy, P.P.: Binary Join Trees for Computing Marginals in the Shenoy-Shafer Architecture. *Int. J. Approx. Reasoning* 17(2-3), 239–263 (1997)
9. Shenoy, P.P., Shafer, G.: Axioms for probability and belief functions propagation. *Uncertainty in Artificial Intelligence* 4, 159–198 (1990)
10. Simon, C., Weber, P., Evsukoff, A.: Bayesian networks inference algorithm to implement Dempster Shafer theory in reliability analysis. *Reliability Engineering and System Safety* 93, 950–963 (2008)
11. Smets, P.: Belief Function: The Disjunctive Rule of Combination and the Generalized Bayesian Theorem. *Int. J. Approx. Reasoning* 9, 1–35 (1993)
12. Weber, P., Simon, C.: Dynamic Evidential Networks in System Reliability Analysis: A Dempster Shafer Approach. In: 16th Mediterranean Conference on Control and Automation, France, vol. 93, pp. 262–267 (2008)

Evidential Grammars for Image Interpretation – Application to Multimodal Traffic Scene Understanding

Jean-Baptiste Bordes¹, Franck Davoine², Philippe Xu¹, and Thierry Denœux¹

¹ UMR CNRS 7253, Heudiasyc, Université de Technologie de Compiègne, BP 20529,
60205 Compiègne Cedex France

² LIAMA, CNRS, Peking University, Beijing, P.R. China

Abstract. In this paper, an original framework for grammar-based image understanding handling uncertainty is presented. The method takes as input an over-segmented image, every segment of which has been annotated during a first stage of image classification. Moreover, we assume that for every segment, the output class may be uncertain and represented by a belief function over all the possible classes. Production rules are also supposed to be provided by experts to define the decomposition of a scene into objects, as well as the decomposition of every object into its components. The originality of our framework is to make it possible to deal with uncertainty in the decomposition, which is particularly useful when the relative frequencies of the production rules cannot be estimated properly. As in traditional visual grammar approaches, the goal is to build the “parse graph” of a test image, which is its hierarchical decomposition from the scene, to objects and parts of objects while taking into account the spatial layout. In this paper, we show that the parse graph of an image can be modelled as an evidential network, and we detail a method to apply a bottom-up inference in this network. A consistency criterion is defined for any parse tree, and the search of the optimal interpretation of an image formulated as an optimization problem. The work was validated on real and publicly available urban driving scene data.

Keywords: Visual grammars, Belief functions, Image understanding.

1 Introduction

Automatic understanding of the traffic scene in front of a car is an essential task for autonomous driving, as well as for safety systems. Generally speaking, by “understanding” we mean detecting the objects in the image and eventually describing some spatial or functional relationships between them. However, detecting even a single kind of object can be very challenging since the varying lighting conditions, the highly cluttered environment, the dynamically changing backgrounds among many others contribute to the difficulty of the task. Many algorithms have been developed which are specialized for one of these detection

subtasks and each of them can use different kinds of sensors. However, even the most competitive state-of-the-art detection modules are sources of false detections, as well as misdetections (pedestrian detection for example). By modelling the consistency of the relative positions of the objects with respect to the scene, it is possible to strengthen weak detection while pruning some false detections. For better results, these post-processing methods should take into account the uncertainty of classification outputs provided by different independent modules, which is possible using the Dempster-Shafer theory [9]. Indeed, this theory provides an elegant formalization of uncertainty and the Dempster's combination rule is an efficient way to fuse information in a multi-sensor context. The critical goal which is addressed in this paper is thus to define a precise framework to model the consistency of a scene while handling the uncertainty on the class of detected objects.

1.1 Related Work

Scene understanding is one of the main goals in computer vision and robotics. In the last decade, the accuracy of object detection methods has increased substantially. On urban scenes, Ess et al. [3] and Gavrila et al. [5] have successfully combined a set of visual cues extracted from a stereo camera for pedestrian detection in real time. More recent approaches try to annotate every pixels of the image with semantic categories: Ess et al [4] use textural features as well as some geometric information to train a set of classifiers in one versus all on small patches of the image. Some temporal smoothing is then applied as a post-processing but the annotation is purely local. Wojek et al [10] exploit the scene layout information at a further extent using joint object detection to perform the segmentation; the model includes some consistency clues about the spatial relationships between the objects (the distance between pedestrians and the ground for example).

More generally, an efficient way to take into account spatial and functional relationships between the objects is the use of ontologies [1], which is the formal representation of concepts for a particular domain and their semantic relationships. The literature on computer vision provides several references about ontology application on various types of databases, e.g., on the PASCAL challenge database using WordNet semantic network [8] or on traffic scenes [1] where Brehar uses the openCyc ontology. The semantic hierarchy provided by ontologies like WordNet or openCyc is useful to shed light on inter-class relations between visual elements. However, we believe it is not sufficient for modelling symbolic representation of a visual system since it does not account for some important aspects of semantic relations such as spatial and functional relationships. In [13], *and/or* graphs are used to represent the decomposition of scenes into objects, parts and low-level patterns called “primitives”. This graph is set in relation with a visual grammar, which is a set of derivation rules. By augmenting these rules with probabilities, a stochastic model is defined which is trained and then used to compute the parsing graph of a test image under the Bayesian posterior. Impressive results have been achieved on various datasets containing a large

number of object categories [12]. The strong advantage of visual grammars over discriminative machine learning methods is their generalization capability: parts of objects can be learnt with a relatively small amount of training data and can be used to recognize a large number of configurations, they also provide elegant and efficient ways to solve problems like occlusions and scale.

1.2 Contributions

Visual grammars take as input a set of visual primitives (texture, corners, color histograms ...) extracted from the image during a first step of image processing. Each one of these primitives provides by itself rather poor information about the class of the area it covers, but there is no doubt about its value. In this paper, we will show how it is possible to extend the visual grammar approach to the case where the input information is uncertain. The belief function theory makes it possible to deal with this situation: every input of information associated to a part of the image which will be defined as a belief function. This approach can be especially interesting in the multimodal case where modules using information incoming from several sensors can extract “higher-level” information than traditional visual primitives: parts of objects can indeed be detected, or even objects themselves. This classification result can be taken as input to our visual grammar method, but the uncertainty of this information has to be taken into account as a module may detect an object with a given confidence degree, or may not be able to discriminate between different classes (a general obstacle detector cannot distinguish a vehicle from a pedestrian). We also propose a way to deal with another kind of uncertainty, which is uncertainty in the production rules. When the relative frequencies of occurrence of different production rules are unknown, rewriting the production rules as conditional belief function allows us not to add artificial information in the model. In brief, our work is at the intersection of two promising approaches: visual grammars and belief functions. To our knowledge, no work has yet focused on the possibility to match these two theories.

1.3 Overview

The system considered here consists of several sensors, like mono or stereo cameras and laser sensors, observing an urban driving scene. Images are over-segmented and then processed at the level of entire segments rather than single pixels. Each sensor provides data to one or more modules which will run totally or partially in parallel and assign a belief function about the class label of the image segments. The output belief functions from different modules are then fused on every segment using the Dempster’s rule of combination [11] (cf Fig 1). In this article, we augment this “local” fusion process with a “global” fusion step by merging the segments into larger regions using predefined combination rules. This can achieve two main goals: disambiguate the belief one has on the segments by adding context information, and infer complex objects by combining small regions corresponding to parts of these objects.

We will show in Section 4 how to propagate the belief functions from the segment level to the region and scene levels. We will then define a criterion which makes it possible to rank all the possible parse trees for a given input image. The understanding process is thus defined as the search for the parse tree minimizing this criterion. We will then present a fast optimization algorithm implementation of this research. We call this approach a “framework” since it can be used for any set of derivation rules, spatial relationships and labels. To apply the evidential grammar, they have to be instantiated. Experimental validation is presented in Section 6, using the publicly available KITTI Vision Benchmark Suite [6].

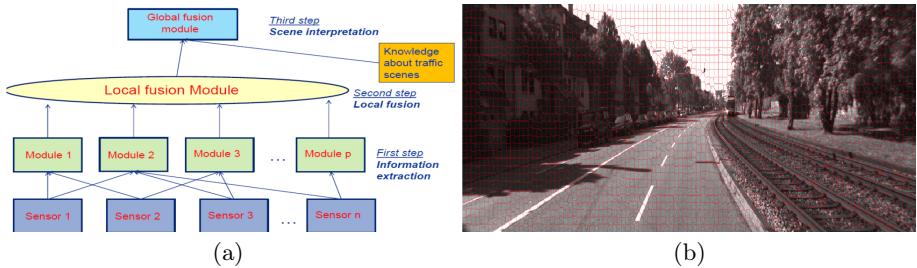


Fig. 1. (a) Global architecture of the system. The approach presented in this article corresponds to the “global fusion module” at the top of this structure. (b) Example of an oversegmented image

2 Evidential Grammars

2.1 Formal Grammars and Stochastic Grammars

The modern formalization of grammars can be attributed to Chomsky [2]. A grammar is defined as a 4-tuple $\{V_N, V_T, S, \Gamma\}$ where V_N is a finite set of non-terminal nodes, V_T a finite set of terminal nodes, S a start symbol at the root, and Γ is a set of production (or derivation) rules. A production rule $\gamma \in \Gamma$ changes a string of symbols (containing at least one non-terminal symbol) into another string of symbols. The production process starts with the S symbol and stops when the string is composed only of terminal symbols. The set of all the possible strings which can be produced by a grammar is called a *language*. The strength of language grammars lies in the fact the language generated by a grammar can be large even when the *vocabulary*, that is to say V_T and V_N contain few elements.

To deal with real data, which may include a large amount of irregular patterns, a grammar should contain a substantial amount of derivation rules to ensure that at least one sequence generates the terminal symbols which are extracted in the data. When several possible sequences of rules can generate the terminal symbols, the grammar is said to be *ambiguous*. To rank alternative

interpretations of a given image, the grammar is augmented with a set of probabilities P as a fifth component to form a stochastic grammar $\{V_N, V_T, S, \Gamma, P\}$. A set of nodes can then be derived in other sets of nodes with given probabilities, taking into account the spatial relationships between the nodes. They are an essential component of the model and are estimated during the training process.

2.2 Visual Grammars

The most important problem when dealing with image grammars is the loss of the natural left-to-right ordering in textual data. In language, every production rule is assumed to generate a linearly ordered sequence of nodes and, following this down to the leaves, a linearly ordered sequence of terminal symbols is obtained. In images, the implicit links between neighbouring words have to be replaced by more complex links between the nodes. Indeed, the spatial links that can combine parts of objects into an object are numerous such as “hinge”, “border”, “butting”, “surround” as well as various alignments. 3D information can also be used to determine that an object is “occluding” another one or that an object is “supporting” another object. In image grammars, pixels seem natural candidates to be terminal symbols, but a single pixel carries very little information and usually some sets of local structures are preferred. Several works have focused on proposing *visual primitives*, sometimes called also *textons* [7] as terminal nodes for visual grammars. These terminal nodes usually correspond to corners or patches of texture; they are computed deterministically during a step of feature extraction.

In this article, a framework is proposed to deal with terminal nodes containing uncertainty. This can be used to combine outputs of a set of modules detecting various objects, parts of objects and areas but which are not perfectly reliable such as, for example in driving scenes, road extraction, wheel or pedestrian detectors, etc. The main idea is to use as terminal nodes a higher-level vocabulary taking into account uncertainty instead of low-level primitives with no uncertainty. We believe this to be especially a good choice in multimodal situation like ours: the extraction of information incoming from various sensors makes it relevant to reason from a higher level than image primitives. However, pedestrian detectors, for example, are well-known to provide many false positives and handling precisely the uncertainty is a matter of highest importance. To this end, the Dempster-Shafer theory will be used and the set of probabilities augmenting the production rules will be replaced by a set of belief functions to constitute what we will call here *evidential grammars*.

2.3 Formalization of Evidential Grammars

An evidential grammar is defined as a 5-tuple $\{V_N, V_T, S, \Gamma, \mu\}$ where the first four elements correspond to the traditional 4-tuple defining a formal grammar. The fifth component μ replaces the set of probabilities of a stochastic grammar. The set of non-terminal elements V_N and the set of terminal elements V_T contain the classes that are used to label the images. The elements of V_N correspond

to the classes of objects which can be decomposed into a set of other elements belonging to $V_N \cup V_T$. The starting element S of the grammar will be interpreted here as the “Scene” annotating every image. Γ contains the production rules of the grammar. Every rule defines a decomposition of a class of object into a set of classes of objects with a defined spatial layout, and every element of V_N has at least one production rule.

More formally, let $A \in V_N$ be a class with $n_{k,\xi}(A)$ production rules into k components under spatial relation ξ (A may have production rules in more or less than k components and these components may have other spatial layouts than ξ):

- $\Gamma_1: A \rightarrow A_1^1 A_2^1 \dots A_k^1$
- \dots
- $\Gamma_{n_{k,\xi}(A)}: A \rightarrow A_1^{n_{k,\xi}(A)} A_2^{n_{k,\xi}(A)} \dots A_k^{n_{k,\xi}(A)}$

where the A_j^i stand for classes of objects belonging to $V_N \cup V_T$. Let us define the precise meaning of these production rules.

Let R be a region of an image and Y an evidential variable describing the class contained inside this region. R is supposed to be segmented in k regions R_1, R_2, \dots, R_k and the class contained in each region R_i is defined by the evidential variable X_i (see Fig 2). The use of evidential variables makes it possible to deal with the uncertainty about the class of the regions. However, even though one might have uncertainty on the value of the class inside these regions, the (strong) assumption is made that every region contains one and only one instance of a class. For example, if it is certain that $Y \in \{\text{car}, \text{pedestrian}\}$ (that is to say, $m_Y(\{\text{car}, \text{pedestrian}\}) = 1$), this means that the region contains either one (full) car or one (full) pedestrian, but this does not mean the region can contain a pedestrian **and** a car.

The spatial relationship between the regions R_1, R_2, \dots, R_k is described by a deterministic variable Ξ (since this spatial layout in an image between a set of regions is always observable). Ξ can describe a global layout between the regions (the regions are “aligned” or “radial”) as well as a set of pairwise relationships.

The production process defined by the production rules $\Gamma_1, \dots, \Gamma_{n_{k,\xi}(A)}$ is associated to a conditional mass function $m_{X_1, \dots, X_k | Y=A, \Xi=\xi}$. To be consistent with the production rules, the only possible focal sets of the latter mass function are the $\Omega_i = \{A_1^i, A_2^i, \dots, A_k^i\}$ and the sets defined as unions of Ω_i . Let us emphasize two particular cases:

1. The case when the focal elements are the Ω_i is equivalent to the production process of a stochastic grammar.
2. The case when $m_{X_1, \dots, X_k | Y=A, \Xi=\xi}(\bigcup_{i=1}^{n_{k,\xi}(A)} \Omega_i) = 1$ is an assumption of complete ignorance about the relative frequencies of output of the production process.

The conditional mass $m_{X_1, \dots, X_k | Y=A, \Xi=\xi}$ describes the derivation process of A into k components with spatial relation ξ . Consequently, conditional mass has to be defined for every possible number of components and every spatial layout in which the class of object A can be derived.

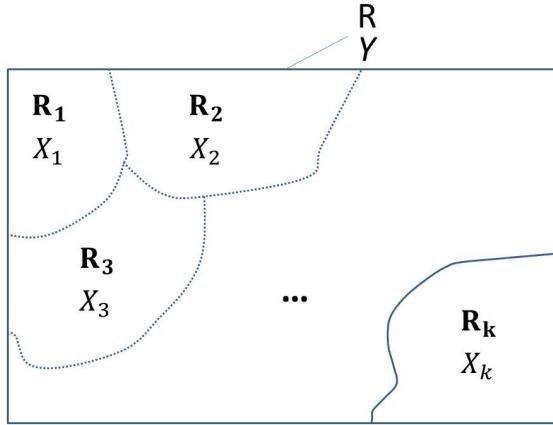


Fig. 2. Illustration of the derivation process: the class of the region R (the whole rectangular area) is described by an evidential variable Y . The region R is then segmented in k regions R_1, R_2, \dots, R_k the classes of which are also described by evidential variables.

3 Model of an Image Interpretation

Parse hypergraphs will be introduced in this section as the structure for image interpretation. Given an evidential grammar $\{V_N, V_T, S, \Gamma, \mu\}$, we define $\Omega = V_N \cup V_T$ as the set of labels which will be used to annotate the image.

3.1 Parse Hypergraphs

Let us consider a test image I which is supposed to be parsed hierarchically in l layers ($l \geq 2$) of a parse hypergraph. A given layer $i \in \{1, \dots, l\}$ is composed of a set of n_i regions which are denoted $\{R_{i1}, \dots, R_{in_i}\}$. The layer l , at the top of the hierarchy, contains a single region which is the image itself I , we can thus write $R_{l1} = I$. If the layer is not the bottom one (ie, $i > 1$) a region R_{ij} can be partitioned in a set of regions of the lower layer (see Fig 3).

Moreover, an evidential variable denoted X_{ij} taking its value in Ω is introduced to describe the content of every region R_{ij} and is associated to the belief function $m_{X_{ij}}$.

The evidential variable X_l is constrained to a single value which is the “Scene” label. “Scene” corresponds to the S symbol of any formal grammar, which is the starting symbol from which all the sentence is derived up to the leaves. The higher-level labels are expected to correspond to more complex objects, like “car” or “pedestrian”, than the lower-level labels, which are expected to correspond to parts of objects like “wheel”, or “pedestrian head”. However, the scale issue implies that it is possible for any object to be located in a small region of an image, making it impossible to parse into its components. Consequently, it must be possible for any symbol (except S) to terminate at any level of the

labels hierarchy. Thus, the evidential grammar $\{V_N, V_T, S, \Gamma, \mu\}$ is defined with $V_N = V_T$.

A *Node* N_{ij} is defined as the pair (R_{ij}, X_{ij}) , that is to say a region and the description of its content. The nodes will be used to represent the oriented hypergraph structure of the image interpretation with two kinds of edges being considered:

1. An edge “is part-of/is composed-of”: $\{N_1, \dots, N_p\}$ are part-of N , or reciprocally N is composed-of $\{N_1, \dots, N_p\}$ if the regions covered by $\{N_1, \dots, N_p\}$ define a partition of the region covered by N .
2. An edge describing the spatial relationship between a set of nodes at the same level.

The first type of edge defines a parse tree since it represents the decomposition of the scene into objects and parts of objects. By adding the second type of edge between nodes of the same layer, the parse tree is augmented to a parse hypergraph. Indeed, to describe complex relations between a set of nodes (radial rays of a bicycle), edges linking several nodes should be considered. The set of nodes and edges defining an image interpretation is called a parse hypergraph of the image I and is denoted $P_h(I)$

3.2 Evidential Network Corresponding to an Image Interpretation

Let $U = \{X_{11}, X_{12}, \dots, X_l\}$ be the set of all the variables used for the image interpretation and p the number of edges in a given parse hypergraph $P_h(I)$. A set of masses $\mathbb{M} = \{m_1, \dots, m_p\}$ is introduced where every mass m_i is defined on the discernment frame $U_i \subset U$ where U_i is a set of variables whose corresponding nodes consist in one node which is linked to several nodes with the “part-of” link (cf Fig 3). Let $O = \{U_1, \dots, U_p\}$ be the set of all the discernment frames. The 3-tuple $\mathbb{E} = \{U, O, \mathbb{M}\}$ defines an evidential network and matches the structure of the image interpretation $P_h(I)$.

As for the Bayesian network where the graphical model corresponds to a set of conditional independences between random variables, the evidential network facilitates the inference of the unknown belief functions since the masses included in \mathbb{M} provide an expression of the joint mass of the subsets of variables U_i independently of the rest of the evidential variables of U . The global joint mass function can be computed as:

$$m_U = \bigoplus_{i=1}^p m_{U_i \uparrow U},$$

where $m_{U_i \uparrow U}$ stands for the vacuous extension in the product space of U .

The belief function of the leave nodes of the parse tree are assumed to be known as an input data and we wish to marginalize U to compute the belief functions on all the other variables. At this point, two questions clearly stand out:

1. Given the hierarchy of regions, how to propagate the belief function from the leaves to the upper nodes?
2. What could be a criterion which could lead us the most realistic interpretation of an image?

These questions will be addressed in the two next sections.

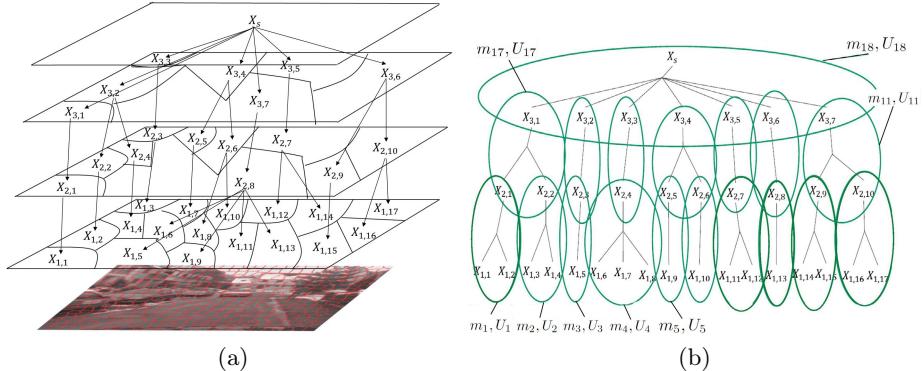


Fig. 3. (a) Hierarchy of regions for the interpretation of a test image (b) Evidential network corresponding to this hierarchy. A joint mass is defined for all the sets of nodes linked by a “part-of” edge (the sets are surrounded by green circles.).

4 Belief Propagation

4.1 Bottom-Up Inference

Given an interpretation $P_h(I)$ of a test image I and the corresponding evidential network $\mathbb{E} = \{U, O, \mathbb{M}\}$, every mass m_i is defined on a set of variables U_i , the corresponding nodes of which are composed k nodes of a layer $j \in \{1, \dots, l-1\}$ linked together by a spatial relationship ξ , and linked to one node in the layer $j+1$ by a “part-of” link. The belief functions of the leaves are assumed to be known as the input of the present algorithm. The scheme which is proposed here consists in computing the belief functions iteratively from layer 2 to layer l : the nodes of layer 1 are used to compute the belief functions of the nodes of layer 2, and so on up to the root.

In a first step, the vacuous extension is applied to the functions $m_{X_1}, m_{X_2}, \dots, m_{X_p}$. The resulting functions are denoted $m_{X_1 \uparrow X_1, X_2, \dots, X_p, Y}, m_{X_2 \uparrow X_1, X_2, \dots, X_p, Y}, \dots, m_{X_p \uparrow X_1, X_2, \dots, X_p, Y}$. These belief functions characterize the contents of disjoint regions and are thus supposed to be independent pieces of evidence. These belief functions are then combined using Dempster’s rule:

$$m_{X_1, X_2, \dots, X_p, Y}^1 = \bigoplus_{i=1}^p m_{X_i \uparrow X_1, X_2, \dots, X_p, Y}.$$

In a second step, all the N conditional belief functions corresponding to grammar rules involving the rewriting of a symbol into p symbols under relation ξ are deconditioned and a set of N functions defined on the product space $\{X_1, \dots, X_p, Y\}$. These belief functions correspond to distinct production rules which themselves encode different semantic information about the decomposition of the objects and the scene. They are thus supposed to be independent pieces of information and Dempster's rule of combination is consequently applied. We have:

$$m_{X_1, X_2, \dots, X_p, Y}^2 = \bigoplus_{k=1}^N m_{X_1, X_2, \dots, X_p, Y | \Xi = \xi}^k,$$

where Ξ is the observable variable defining the spatial relation between the regions. This function is then combined with the previous one, to get a belief function taking into account all the available information:

$$m_{X_1, X_2, \dots, X_p, Y} = m_{X_1, X_2, \dots, X_p, Y}^1 \oplus m_{X_1, X_2, \dots, X_p, Y}^2.$$

The joint mass $m_{X_1, X_2, \dots, X_p, Y}$ is finally marginalized to extract m_Y .

4.2 Top-Down Propagation

Once all the mass functions have been computed through the previous inference mechanism, a top-down scheme can be performed to disambiguate the mass functions at the lower layers of the hierarchy. For every subset of variables $U_i = \{X_1, X_2, \dots, X_p, Y\}$ and after the joint mass has been computed, k marginalizations are applied to compute $m_{X_1}, m_{X_2}, \dots, m_{X_k}$. This process is performed from the root down to the leaves for every subset of variables. This scheme results in a disambiguation of the variables by re-injecting the information gathered through the combinations performed during the inference down to the variables corresponding to the segment level of the image.

5 Search for the Optimal Interpretation

The groupings of nodes with the “part-of” links define a hierarchy of regions and this hierarchy define an evidential network. By using the scheme detailed in the previous section, the belief functions are computed from the lower levels up to the root to get an interpretation of an image. However, a large number of possible parse trees can be considered and consequently as many possible interpretations of a same image. Using traditional stochastic visual grammars, we build the parse tree of a test image with maximal posterior probability. For evidential visual grammars, we propose to use the minimal conflict accumulated at the root node as the optimization criterion to select the parse hypergraph.

5.1 Optimization Criterion

The key value that will be used to measure how relevant it is to group a set of nodes is the conflict located in the newly constructed node. By conflict, we mean here the value of the mass function of this node on the empty set.

Consistency Measure of an Interpretation. The dependencies between the variables induced by the grouping of nodes with the “part-of” link imply two remarks:

1. If several nodes N_1, N_2, \dots, N_p are linked to an upper node M by a “part-of” edge in the parse tree, and if their mass functions don’t match any derivation rule, the upper node M will contain a conflict equals to 1. More generally, a node containing a high conflict indicates high inconsistency in the subtree.
2. When several nodes N_1, N_2, \dots, N_p are linked to an upper node M by a “part-of” edge in the parse tree, the combination formula implies that the conflict of M is bigger than the maximum conflict of N_1, N_2, \dots, N_p . Thus, any conflict appearing during the bottom-up belief propagation process in the evidential network will appear on the root node.

These two remarks drive us to consider the conflict in the node N_{l1} as the main optimization criterion, since as the root node of the evidential network, it aggregates all the conflict contained in the evidential network and it gives a measure of the quality of the hierarchy. The lower is $m_{X_{l1}}(\emptyset)$ (or the higher is $m_{X_{l1}}(S)$ since $m_{X_{l1}}(S) = 1 - m_{X_{l1}}(\emptyset)$), the more consistent is the hierarchy.

Remarks on the Number of Optimal Interpretations. The principle of consistency maximization does not ensure that only one optimal interpretation exists. In some cases, many alternative interpretations of an image can be found which have no conflict at the root node. Actually, two factors impact the number of interpretations that can be expected:

1. The quantity of information carried by the belief functions of the input nodes.
2. The number of production rules in the model.

The less informative the input belief functions are and the more numerous are the production rules, the more possible combinations there are to form different consistent interpretations. The production rules being a part of the model, they have to fit correctly the database: too restrictive rules will lead to no available interpretation since any interpretation will have a high level of conflict. On the contrary, defining too many configurations or allowing too many possible decompositions of the objects will lead to an output containing many unsatisfactory interpretations.

5.2 Optimization Algorithm

The number of possible parse hypergraphs is very large and exploring the whole space is untractable. A greedy algorithm is thus introduced here to obtain a relevant interpretation of a test image in reasonable computation time. The main idea of this algorithm is to initiate a complex configuration which is simplified step by step as long as the consistency measure of the parse tree decreases:

- A parse tree is first initialized by linking all the nodes corresponding to the segments of the image directly to the root node. This is equivalent to considering that every segment is interpreted as one object.
- As long as the consistency measure of the parse tree decreases:
 - The consistency measure is computed for a set of alternative hypergraphs, each one being obtained by applying one single elementary modification to the current parse hypergraph. The elementary modifications that we consider are the merging of every pair of nodes of the same level of the hierarchy of the parse graph. If the nodes are terminal nodes, a new node is created which is linked by the “part of” relationship with this pair of nodes. If the nodes are not terminal nodes, a new node is created which is composed of all the children of this pair of nodes.
 - The parse hypergraph minimizing the consistency measure is kept for the next iteration.
- The last parse hypergraph is kept as the output of the method.

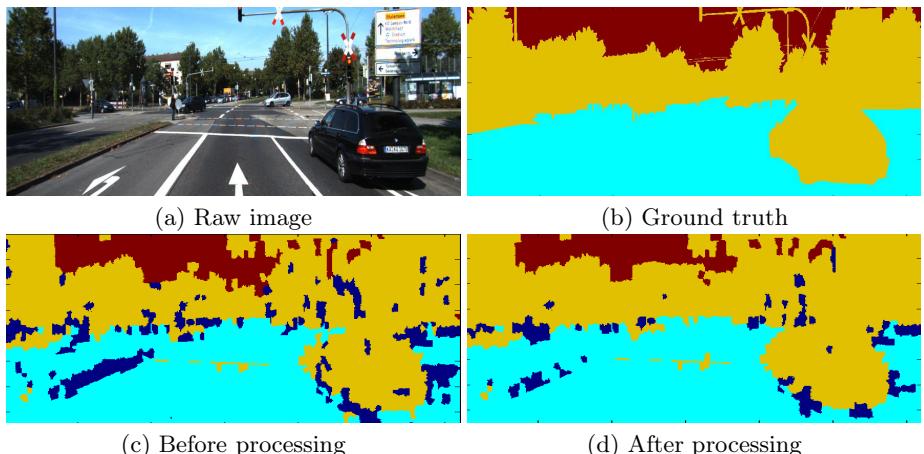


Fig. 4. Three classes $\Omega = \{\text{Ground}, \text{Sky}, \text{Obstacle}\}$ classification result. The deep blue color indicates that the image segment has not been classified by the system.

6 Experiments

The KITTI dataset [6] was used to validate our approach. The stereo color camera system and Velodyne LIDAR were used as sensors. Using the algorithms presented in [11], the images of the dataset were first oversegmented and a combination of modules was used and merged locally at the segment level to provide a belief function for every segment of the image.

Three classes of objects were considered for the definition of the grammar: “Ground”, “Obstacle” and “Sky”. The pairwise links “occlude”, “is occluded by”, “bordering” and “disjoint” were used to describe the spatial relationships between the regions. The starting node S can produce an arbitrary combination of these objects under a few basic spatial constraints (for example, the sky cannot occlude the other types of objects). At the part of object level, three classes were considered: “Patch of ground”, “Patch of obstacle”, and “Patch of sky”. The pairwise links “neighbouring” and “disjoint” were considered. Each object can be decomposed in an arbitrary combination of the corresponding patches of object under a spatial constraint of neighbourhood. It should be noticed that this model takes only little advantage of the potential of the grammars to decompose complex objects in structured and reusable components. Indeed, no input data for parts-of-objects were available for that purpose. We plan to apply a model at the object part level in future work.

Fig 5 shows the classification results before and after processing of the evidential grammar. Since our system allows us to represent ignorance, it can happen that no decision can be made about some segments, which explains why the recall rate is different from the diagonal of the confusion matrix. A result image is shown in Fig 4. As it can be seen on these results, there is little difference in precision but the evidential grammar provides an important improvement in recall. This comes from the disambiguation provided by the top-down propagation of the belief, which transfers the fused information at the region level back to the segment level.

	Ground	Obstacle	Sky	Recall
Ground	96,4	3,6	0	86,1
Obstacle	5,9	94,0	0,1	81,3
Sky	0	33,2	66,8	66,5

	Ground	Obstacle	Sky	Recall
Ground	97,0	3,0	0	95,7
Obstacle	6,2	93,7	0,1	85,8
Sky	0	33,2	66,8	66,5

(a)

(b)

Fig. 5. Confusion matrices, values are given in percentage. (a) Results without processing of evidential grammar (b) Results after processing of evidential grammar.

7 Conclusion

We have introduced an original framework for image understanding based on visual grammars and Dempster-Shafer theory. Our method makes it possible

to introduce uncertainty in the production rules of the grammar, which allows us to bypass artificial knowledge when the relative frequencies of the output of some derivation rules cannot be estimated reliably. Moreover, our method can handle input data tainted with uncertainty. We demonstrated the efficiency of our method by post-processing the classification result of an oversegmented image with uncertainty on the class of the segments. We showed that our method provides significant improvement on the recall rate. Future work will take into account information at the part-of-object level in order to fully exploit the strength of the grammars, which lies in their ability to combine visual elements to detect complex structured objects even in highly cluttered environment.

References

1. Brehar, R., Fortuna, C., Bota, S., Mladenic, D., Nedevschi, S.: Spatio-temporal reasoning for traffic scene understanding. In: Proc. of ICCP, Pittsburgh, pp. 377–384 (2011)
2. Chomsky, N.: Syntactic Structures. The Hague, Mouton (1957)
3. Ess, A., Leibe, B., Schindler, K., Gool, L.V.: Moving obstacle detection in highly dynamic scenes. In: Proc. of ICRA, Kobe, pp. 4451–4458 (2009)
4. Ess, A., Mueller, T., Grabner, H., Van Gool, L.J.: Segmentation-based urban traffic scene understanding. In: Proc. of BMVC, London, pp. 84.1–84.11 (2009)
5. Gavrila, D.M., Munder, S.: Multi-cue pedestrian detection and tracking from a moving vehicle. IJCV 73(1), 41–59 (2007)
6. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? In: Proc. of CVPR, Providence, USA, pp. 3354–3361 (June 2012)
7. Julesz, B.: Textons, the elements of texture perception, and their interaction. Nature 290 (1981)
8. Marszałek, M., Schmid, C.: Semantic hierarchies for visual object recognition. In: Proc. of CVPR, Minneapolis, pp. 1–7 (2007)
9. Shafer, G.: A mathematical theory of evidence. Princeton University Press (1976)
10. Wojek, C., Schiele, B.: A dynamic conditional random field model for joint labeling of object and scene classes. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part IV. LNCS, vol. 5305, pp. 733–747. Springer, Heidelberg (2008)
11. Xu, P., Davoine, F., Bordes, J.-B., Zhao, H., Denœux, T.: Information fusion on oversegmented images: An application for urban scene understanding. In: Proc. of MVA, Kyoto (2013)
12. Yao, B., Yang, X., Zhu, S.-C.: Introduction to a large-scale general purpose ground truth database: Methodology, annotation tool and benchmarks. In: Yuille, A.L., Zhu, S.-C., Cremers, D., Wang, Y. (eds.) EMMCVPR 2007. LNCS, vol. 4679, pp. 169–183. Springer, Heidelberg (2007)
13. Zhu, S.-C., Mumford, D.: A stochastic grammar of images. Found. Trends. Comput. Graph. Vis. 2(4), 259–362 (2006)

Generic Discounting Evaluation Approach for Urban Image Classification

Ahmed Samet¹, Imen Hammami¹, Eric Lefèvre², and Atef Hamouda¹

¹ Laboratory of Research in Programming, Algorithmic and Heuristic Faculty
of Science of Tunis, Tunisia

{ahmed.samet,imen.hammami,atef.hamouda}@fst.rnu.tn

² Univ. Lille Nord de France UArtois, EA 3926 LGI2A, 62400, Béthune, France
eric.lefeuvre@univ-artois.fr

Abstract. Belief function theory provides a robust framework for uncertain information modeling. It also offers several fusion tools in order to profit from multi-source context. Nevertheless, fusion is a sensible task where conflictual information may appear especially when sources are unreliable. Therefore, measuring source's reliability has been the center of many research and development. Existing solutions for source's reliability estimation are based on the assumption that distance is the only factor for conflictual situations. Indeed, integrating only distance measures to estimate source's reliability is not sufficient where source's confusion may be also considered as conflict origin. In this paper, we tackle reliability estimation and we introduce a new discounting operator that considers those two possible conflict origins. We propose an automatic method for discounting factor calculation. Those factors are integrated on belief classifier and tested on high-resolution image classification problem.

Keywords: Belief function theory, Discounting, Classification, Conflict management, Source confusion.

1 Introduction

The improvement of image acquisition techniques have led to the treatment of more complex images in terms of details. This detail complexity comes generally from the multi-band nature of the image. The multi-source information is too valuable for decision process but needs an adequate formalism able to manipulate it. In order to synthesize more useful information related to the observed scene, many formalisms were proposed manipulating those multi-source information and allowing to formalize mathematically uncertain and imprecise data as the Bayesian theory, fuzzy set theory,...

The belief function theory, introduced by Dempster [2] and formalized by Shafer [14], presents a powerful mathematical background in information fusion domain. It not only allows modelizing mathematically uncertainty and imprecision information but it also integrates many combination tools allowing source fusion. The fusion ability of this formalism is granted by several combination

rules; the oldest is the Dempster's rule of combination. However, Zadeh in [20] highlighted its counter-intuitive behavior. Additionally, a lot of studies have been interested on fusing information in the context of image classification [12].

Despite, the fact that belief function theory excels in extracting the most truthful proposition from a multi-source context, it nevertheless presents a major inconvenient that is conflict. The conflict is weight accorded to the empty set proposition and generally appears after source combination. Many works have been done in this domain allowing conflict elimination. In literature, we distinguish two main conflict management family approaches. The first family consists of managing the conflict meaning the combination operator allowing fit conflict redistribution. From those works we can cite [17, 19]. Some authors tried to unify these combination rules [4, 11].

Another method to reduce conflict is the use of discounting factors [14] before combining sources. Indeed, those kinds of approaches rely on the fact that conflict is induced and generated by the unreliability of at least one source. Many works have been done finding those discounting factors such that [6, 8, 9].

Shafer in [14] has proven that the resulting conflict may not only come from source's contradiction in combination phase. Indeed, the confusion rate of a source may generate conflict. This assumption means that the more the source is less informative, the higher the conflict is. To the best of our knowledge, rare are the discounting based approaches that addressed conflict taking those two conflict origins into consideration.

In this paper, we consider two possible factors for conflict and should be taken into consideration. The *intrinsic conflict* caused by the unreliability of a source to determine certain classes. The second considered conflict origin is the *extrinsic conflict* which indicates to what extent the obtained sources are in contradiction. In this work, we propose a new conflict management approach denoted *Generic Discounting Approach* (GDA) based on discounting factors determination. Those discounting factors are found by studying not only the confusion of the source but also its contradiction with the other ones. The GDA discounting approach was experimented on a high-resolution urban image classification problem. We take advantage of the belief function theory in order to modelize imperfect data extracted from the image and benefit from combination rules. Our classification approach is based on two main stages where, in the first phase, we combine multi-source information in order to obtain a reference classification. In the second stage, we try to improve the classification result by adding the GDA discounting approach.

This paper is organized as follows: in the second section we briefly introduce the basics of the belief function theory. In section 3, we detail several extrinsic and intrinsic reliability measures developed in the framework of belief function theory. In the following, we introduce our Generic Discounting Approach (GDA) capable of estimating the reliability of a source regarding its two sided conflict measures. In section 5, we experiment our approach on a high resolution urban image providing comparative results with other notable works. Finally, we conclude and we sketch issues of future work.

2 Belief Function Theory

The belief function theory or the evidence theory was introduced by Dempster [2] in order to represent some imprecise probabilities with upper and lower probabilities. Then, it was mathematically formalized by Shafer [14]. The belief function theory is used for representing imperfect (uncertain, imprecise and/or incomplete) information. In this section, we present the main concepts of this theory.

2.1 Frame of Discernment

The frame of discernment is the set of possible answers for a treated problem and generally noted θ . It is composed of exhaustive and exclusive hypotheses:

$$\theta = \{H_1, H_2, \dots, H_N\}.$$

These elements are assumed to be mutually exclusive and exhaustive. From the frame of discernment θ , we deduce the set 2^θ containing all the 2^N subsets A of θ :

$$2^\theta = \{A, A \subseteq \theta\} = \{H_1, H_2, \dots, H_N, H_1 \cup H_2, \dots, \theta\}$$

This set constitutes a reference to assess the veracity of any proposal.

2.2 Basic Belief Assignment

A Basic Belief Assignment (BBA) m is the mapping from elements of the power set 2^θ onto $[0, 1]$ such that:

$$m : 2^\theta \longrightarrow [0, 1]$$

having as constraints:

$$\begin{cases} \sum_{A \subseteq \theta} m(A) = 1 \\ m(\emptyset) = 0. \end{cases} \quad (1)$$

Each subsets X of 2^θ verifying $m(X) > 0$ is called focal elements. Constraining $m(\emptyset) = 0$ is the normalized form of a BBA and this corresponds to a closed-world assumption [16], while allowing $m(\emptyset) > 0$ corresponds to an open world assumption [17].

2.3 Combination Rules

The belief function offers many advantages. One of its proposed asset is the information fusion allowing extracting the more veracious proposition from a multi-source context. This benefit is granted by the combination rules.

Several operators were defined such the conjunctive rule allowing fusion without any normalization (conflict management). For two sources S_1 and S_2 having respectively m_1 and m_2 as BBA, the conjunctive rule is defined by:

$$m_{\odot}(A) = \sum_{B \cap C = A} m_1(B) \times m_2(C) \quad \forall A \subseteq \theta. \quad (2)$$

A normalized version of conjunctive rule proposed by Dempster [2] integrates a conflict management approach that redistributes the generated conflictual mass. The Dempster's rule is defined as follows:

$$m_{\oplus}(A) = \frac{1}{1 - K} \sum_{B \cap C = A} m_1(B) \times m_2(C) = \frac{1}{1 - K} m_{\odot}(A) \quad \forall A \subseteq \theta, A \neq \emptyset \quad (3)$$

where K is defined as:

$$K = \sum_{B \cap C = \emptyset} m_1(B) \times m_2(C) = m_{\odot}(\emptyset). \quad (4)$$

K represents the conflict mass between m_1 and m_2 .

2.4 Decision Operators

In literature, among several functions that were proposed, we distinguish the pignistic probability. The pignistic probability noted BetP was proposed by Smets [17] within his Transferable Belief Model (TBM) approach. TBM is based on the differentiation between the knowledge representation and decision-making level. In the decision phase, the pignistic transformation consists in distributing equiprobably the mass of a proposition A on its contained hypotheses, formally:

$$BetP(H_n) = \sum_{A \subseteq \theta} \frac{|H_n \cap A|}{|A|} \times m(A) \quad \forall H_n \in \theta. \quad (5)$$

2.5 Discounting

Assuming that an information source has a reliability rate equal to $(1 - \alpha)$ where $(0 \leq \alpha \leq 1)$, such meta-knowledge can be taken into account using the discounting operation introduced by Shafer [14], and defined by:

$$\begin{cases} m^{\alpha}(B) = (1 - \alpha) \times m(B) & \forall B \subseteq \theta \\ m^{\alpha}(\theta) = (1 - \alpha) \times m(\theta) + \alpha. \end{cases} \quad (6)$$

A discount rate α equal to 1 means that the source is not reliable and the piece of information it provides cannot be taken into account. On the contrary, a null discount rate indicates that the source is fully reliable and the piece of information it provides is entirely accepted. Thanks to discounting, an unreliable source's BBA is transformed into a function assigning a larger mass to θ .

3 Reliability Measure and Discounting

Empty set mass resulting from the conjunctive rule can be considered as the most obvious conflict measure in the belief function theory. The conflictual mass could be seen as a measure of contradiction between BBA. Shafer in [14], has defined the confusion of a source as a possible cause for conflict appearance after combination. The conflictual mass resulting from fusion could be seen as the result of the source confusion and the contradiction between fused BBA. In this part, we distinguish between two possible causes for the conflict. At first, the intrinsic conflict which is the conflict generated by the confusion rate of a source. The extrinsic conflict is the conflict resulting from source contradiction. In this section, we present notable state of art of metrics used to measure those two types of conflict.

3.1 Intrinsic Conflict

The intrinsic conflict measures the consistency between the different focal elements inside the BBA. Several measures have been proposed in literature. These measures take into account the inclusion relations between the focal elements present in the BBA. Several measures were proposed such that the auto-conflict [10]. Nevertheless, auto-conflict is a kind of contradiction measure that depends on order, it was therefore necessarily to define an independent measure that get rid of this constraint. Smarandache et al. [15] proposed the contradiction measure, defined by:

$$contr(m) = c \sum_{X \subseteq 2^\theta} m(X) \cdot D(m, m_X) \quad (7)$$

where $m_X(X) = 1$, $X \in 2^\theta$ is the categorical BBA, c is normalization constant and D is the Jousselme's distance [7] defined for two mass function m_1 and m_2 by:

$$D(m_1, m_2) = \sqrt{\frac{1}{2} \cdot (m_1 - m_2)^t \cdot \mathcal{D} \cdot (m_1 - m_2)} \quad (8)$$

$$\mathcal{D}(A, B) = \begin{cases} 1 & \text{if } A = B = \emptyset \\ \frac{|A \cap B|}{|A \cup B|} & \text{if } A, B \subseteq 2^\theta. \end{cases} \quad (9)$$

From the other intrinsic distance, we can cite the confusion measure [5], the auto-conflict [10], ...

3.2 Extrinsic Conflict

Several measures of extrinsic conflict have been studied in order to model the disagreement between sources. Indeed, if one source opinion disagree the other, their fusion will lead to an important conflictual mass. Some authors have defined distance between the mass functions directly, such as Jousselme's distance [7]

(equation 8) which has the advantage of taking into account the cardinality of focal elements. From other distances, we can cite Euclidean distance [11], Tessem's distance [18], Milan's distance [1]. Martin et al. [9] proposes using a function that quantifies the conflict between BBA. This function, called $Conf(.,.)$, is defined as:

$$Conf(i, E) = \frac{1}{M-1} \sum_{k=1; i \neq k}^M Conf(i, k) \quad (10)$$

with M is the number of belief functions produced respectively by M sources called S_1, \dots, S_M and E is the set of BBA such that $\{m_k | k = 1, \dots, M \text{ and } k \neq i\}$. The function $Conf(i, k)$ is obtained using a BBA distance introduced by Jousselme et al. [7]:

$$Conf(i, k) = D(m_i, m_k). \quad (11)$$

The value $Conf(i, E)$ quantifies the average conflict between the BBA m_i and the BBAs of the set E . Once the conflict measure is obtained, the authors have proposed to compute discounting rates as follows:

$$\alpha_i = f(Conf(i, M)) \quad (12)$$

where f is a decreasing function. The authors propose to choose the function f as follows:

$$\alpha_i = (1 - Conf(i, M)^\lambda)^{1/\lambda} \quad (13)$$

with $\lambda > 0$. The authors recommend setting λ to 1.5. Extensions of this work use the idea of sequential discount to manage the conflict when combining belief functions [8].

4 Generic Discounting Approach (GDA) for Conflict Management

Many works have been proposed for finding discounting factors in order to eliminate conflict. Most of proposed discounting approaches rely on extrinsic measures rather than intrinsic measures. Rare were works that tried to associate those two conflict origins to estimate source reliability. In this section, we propose an automatic method to find those discounting factors depending on the two conflict measures. The GDA is a discounting approach that estimates source's reliability based on the two conflict origins. It is a function f satisfying several constraints:

- f is an increasing function from $[0, 1]^2 \rightarrow [0, 1]$
- $f(1, 1) = 1$ and $f(0, 0) = 0$.

$$\begin{cases} f := [0, 1]^2 \rightarrow [0, 1] \\ (\delta, \beta) \rightarrow \delta^{(1-\beta)} \end{cases} \quad (14)$$

where δ is an extrinsic measure and β is an intrinsic measure that can be chosen as follows:

$$\delta = Dist(m) \quad (15)$$

$$\beta = contr(m). \quad (16)$$

Thus, the GDA is a function f that can be written as follows:

$$\begin{cases} f := [0, 1]^2 \rightarrow [0, 1] \\ (\delta, \beta) \rightarrow (Dist(m))^{(1-contr(m))}. \end{cases} \quad (17)$$

where $Dist(m)$ designates the distance between m and other BBA and can be written:

$$Dist(m) = \frac{\sum_{m_i \setminus m, i \in [1..I]} D(m, m_i)}{I - 1}. \quad (18)$$

D is the Jousselme's distance and $contr(m)$ is the contradiction value indicating the confusion rate of the source itself. The discounting formula (equation 6) can be written as follows:

$$\begin{cases} m^{GDA}(B) = (1 - f(\delta, \beta)) \times m(B) & \forall B \subseteq \theta \\ m^{GDA}(\theta) = (1 - f(\delta, \beta)) \times m(\theta) + f(\delta, \beta) \end{cases} \quad (19)$$

In [13], we proposed another version for source discounting regrouping both intrinsic and extrinsic measures. The Table 1 shows the discounting value that could be associated to a BBA depending on the confusion and distance rates.

Table 1. The GDA discounting value for the extremum cases

Source	With Confusion	Without Confusion
Distant	$f(\delta, \beta) = 1$	$f(\delta, \beta) = D(m, \bar{m})$
Near	$f(\delta, \beta) = 1$	$f(\delta, \beta) = 0$

Example 1. Let's consider the frame of discernment $\theta = \{H_1, H_2\}$ and three sources S_1 , S_2 and S_3 . The belief function values associated to those sources and their discounting values are calculated in Table 2.

We can remark that the distance between a source and the other directly affects the GDA discounting coefficient. The BBA's distance (extrinsic measure) is powered by the confusion (intrinsic measure) that is why the GDA factor is equal to the extrinsic measure when the source is not confused. However, it increases the more confused the source is. As it is shown in Table 2, GDA decreases drastically conflict where comparatively to the conjunctive sum, it fell from 0.717 to 0.0748. GDA also improved $\{H_1\}$ and $\{H_2\}$ hypothesis credibilities by considering S_1 and S_3 unconfused nature.

Table 2. Evaluation of discounting approach on an example

	S_1	S_2	S_3	$m \odot$	m_{\odot}^{GPA}
$\{H_1\}$	0.7	0.6	0.2	0.187	0.3061
$\{H_2\}$	0.1	0.3	0.7	0.0.094	0.1613
θ	0.2	0.1	0.1	0.002	0.4578
\emptyset	0	0	0	0.717	0.0748
δ	0.5845	0.7891	0.6904	-	-
β	0.3452	0.2691	0.4562	-	-
$f(\delta, \beta)$	0.6428	0.7581	0.7842	-	-

5 Classification Optimization by Discounting Factors Determination

In this part, we detail our image classifier and how we managed to integrate the GDA discounting to optimize the result of high-resolution urban image first classification. We experimented our approach on a Quickbird image covering urban areas of Strasbourg, taken in 2008, having four bands, each band with 2.44-2.88m/px. From the variety of object constituting this image, we are interested in finding roads, buildings and vegetation (see figure 1). Those three classes will constitute our frame of discernment $\theta = \{Roads, Building, Vegetation\}$. In order to extract correctly those classes, we used five different sources. Each source corresponds to a band from the image. The five considered sources are: the R/G/B bands, the NDVI band and the PIR band.

5.1 First Classification

In belief function estimation and classification, we distinguish two main family approaches. Likelihood based approaches [14], rely on density estimation where they assume known the class-conditional probability densities for each class. The second family, is the distance based approaches introduced by Zouhal and Denoeux [21]. Both methods are applicable in our image classification problem, but we have chosen to work with the distance based model for its simplicity of its generated BBA. The distance classifier relies on training base constituted by I vectors x_i . Each training vector, belonging to H_n^i , sufficiently close to the vector to classify x constitutes a piece of evidence and should be taken into consideration. Indeed, this piece of evidence influences our belief concerning the class membership of the entity under consideration. A fraction of the unit mass is assigned by m to the singleton $\{H_n^i\}$, and the rest is assigned to the whole frame of discernment. Denceux in [3], proposed a new alternative to the training base by characterizing each studied class by a prototype (a value that represents the class) rather than using training vectors x_i . Two strategies for distance modeling can be differentiated which are the mono-dimensional and multi-dimensional variants. The difference consists in fusion's level where the multi-dimensional strategy considers the vector x as a single information leading

to a unique fusion level. In the other hand, the mono-dimensional strategy apply the described distance estimation for each vector x component that lead to a two fusion level. In our case, we studied the mono-dimensional rather than the multi-dimensional variant which is also applicable. As training set we associated the distance estimation to a prototype base. For every x component x_j (with $j \in [1,..,J]$), we estimate our BBA following this expression:

$$\begin{cases} m_{sj}(\{H_n\}) = \alpha_j^s \phi_j^s(d_j^s) \\ m_{sj}(\theta) = 1 - \alpha_j^s \phi_j^s(d_j^s) \end{cases} \quad (20)$$

where $0 < \alpha_j^s < 1$ is a constant. ϕ_j^s is a decreasing function verifying $\phi_j^s(0) = 1$ and $\lim_{d \rightarrow \infty} \phi_j^s(d) = 0$, d_j^s represent the distance between the constituent x_j of the vector x and the component j -th prototype of source s . The ϕ_j^s function might be an exponential function following this form:

$$\phi_j^s(d_j^s) = \exp(-\gamma_j^s(d_j^s)^2). \quad (21)$$

The use of Dempster combination operator allows merging those J belief functions. m_s is the resulting belief function:

$$m_s = \oplus_{j \in [1,J]} m_{sj}. \quad (22)$$

A unique belief function m is obtained by the application of the same fusion principle on those resulting S BBA:

$$m = \oplus_{s \in [1,S]} m_s \quad (23)$$

with S the number of source. The described method constitutes the *Distance Classifier* (DC). However, for our first classification, we replace the Dempster's combination rule (equation 23) by the conjunctive sum (equation 2) in order to generate conflict and analyze source's discordance. Those two methods are applied on each pixel of the image following to the studied sources.

5.2 Classification Optimization

After a first classification, a high amount of conflict is generated making the results more or less acceptable. As it is shown in Table 3, more than 70 percent of our BBA present a conflict rate exceeding 0.6. This result means that 70 percent of the obtained BBA can be potentially attributed to another suited class. A conflict management approach is needed in order to improve results. In this section, we present how we managed to optimize and improve our classification results by adding an extra process which is conflict management.

The pixels (BBA) that present a conflict rate superior to a threshold are reanalyzed by a new combination phase. In this new fusion step, we aim to discard unreliable sources following GDA discounting factors (equation 19).

Table 3. Conflict rate after the first classification

Conflict rate < 0.4	< 0.6	> 0.6
5.006%	24.195%	70.779%

5.3 Application and Results

In order to test our approach we compared it to the DC approach (described in section 5.1) based on distance estimation where conflict management is operated thanks to the Dempster’s combination rule. The comparison were also conducted to Martin discounting approach [9]. The test has been done 8458 images pixel where 1825 represent building points, 1666 road points and 4967 vegetation pixels.

Table 4. Comparative classification results

	Building			Road			Vegetation		
	DC	Mart	GDA	DC	Mart	GDA	DC	Mart	GDA
Building	68.25%	72.79%	71.52%	28.19%	25.90%	27.46%	3.56%	1.31%	1.02%
Road	18.03%	15.21%	13.63%	81.69%	84.51%	86.12%	0.28%	0.28%	0.25%
Vegetation	0.21%	0%	0%	1.53%	0.36%	0.19%	98.26%	99.64%	99.81%

As it is shown in the Table 4, the proposed discounting factor approach (GDA) presents quite satisfying results. The versatility of the proposed discounting factors associated to the confused nature of the BBA have improved results. By comparing our approach to the DC approach, we can notice that we did improve all class detection. We can conclude that we did optimize the first classification.

The Figure 4 represents the classification of the original image (Figure 1) with DC approach. A first classification is applied as announced in section and instead of using the conjunctive combination rule, we apply Dempster’s rule that integrates a conflict manager. The Figure 3 represents the initial image classification with Martin discounting approach.

The Figure 2 represents the same urban image classified using the proposed conflict manager. For each pixel in this image, we apply the first classification. For each high conflict pixel (BBA with high conflict rate), we calculate the discounting factor using the BBA’s extrinsic and intrinsic rate.

6 Conclusion

In this paper, we presented a classification approach for urban high-resolution image. This method is based on two stages, a first classification based on the belief function framework. This classification is improved and optimized with a new conflict management approach. The conflict management is based on an



Fig. 1. Strasbourg high-resolution urban image



Fig. 2. GDA classification results



Fig. 3. Martin discounting approach classification results



Fig. 4. CCA classification results

automatic discounting factor calculation. The discounting factors are found using not only the BBA distance measure but also the confusion rate of belief function. The first classification result has improved thanks to the discounting factors determination. In future work, we will try to propose a complete classification approach based on fusion discounted pieces of evidence. This discounting will also associate an intrinsic and extrinsic measures for more adequate combination. Even if the results are satisfying on image classification, tests can be extended to UCI benchmarks to verify GDA contribution.

References

1. Daniel, M.: Conflicts within and between belief functions. In: Hüllermeier, E., Kruse, R., Hoffmann, F. (eds.) IPMU 2010. LNCS, vol. 6178, pp. 696–705. Springer, Heidelberg (2010)
2. Dempster, A.P.: Upper and lower probabilities induced by a multivalued mapping. *Annals of Mathematical Statistics* 38, 325–339 (1967)
3. Denoeux, T.: A neural network classifier based on Dempster-Shafer theory. *IEEE Transactions on Systems, Man and Cybernetics, Part A* 30(2), 131–150 (2000)

4. Florea, M.C., Bossé, E.: Crisis management using Dempster Shafer theory: Using dissimilarity measures to characterize sources' reliability. C3I for Crisis, Emergency and Consequence Management, Bucharest, Roumania (2009)
5. George, T., Pal, N.R.: Quantification of conflict in Dempster-Shafer framework: A new approach. *International Journal of General Systems* 24(4), 407–423 (1996)
6. Guo, H., Shi, W., Deng, Y.: Evaluating sensor reliability in classification problems based on evidence theory. *IEEE Transactions on Systems, Man, and Cybernetics, Part B* 36(5), 970–987 (2006)
7. Jousselme, A.-L., Grenier, D., Bossé, E.: A new distance between two bodies of evidence. *Information Fusion* 2, 91–101 (2001)
8. Klein, J., Colot, O.: Automatic discounting rate computation using a dissent criterion. In: Workshop on the Theory of Belief Functions, Brest, France, pp. 1–6 (2010)
9. Martin, A., Jousselme, A.-L., Osswald, C.: Conflict measure for the discounting operation on belief functions. In: International Conference on Information Fusion, Brest, France, pp. 1003–1010 (2008)
10. Osswald, C., Martin, A.: Understanding the large family of Dempster-Shafer theory's fusion operators - a decision-based measure. In: 9th International Conference on Information Fusion, pp. 1–7 (July 2006)
11. Ristic, B., Smets, P.: The TBM global distance measure for the association of uncertain combat id declarations. *Information Fusion* 7, 276–284 (2006)
12. Samet, A., Ben Dhiaf, Z., Hamouda, A., Lefevre, E.: Classification of high-resolution remote sensing image by adapting the distance belief function estimation model. In: International Conference on Communications, Computing and Control Applications (CCCA), pp. 1–6 (2011)
13. Samet, A., Lefevre, E., Yahia, S.B.: Reliability estimation with extrinsic and intrinsic measure in belief function theory. To appear in 5th International Conference on Modeling, Simulation and Applied Optimization (ICMSAO 2013), April 28-30, Hammamet, Tunisia (2013)
14. Shafer, G.: A Mathematical Theory of Evidence. Princeton University Press, Princeton (1976)
15. Smarandache, F., Martin, A., Osswald, C.: Contradiction measures and specificity degrees of basic belief assignments. In: International Conference on Information Fusion, pp. 1–8 (2011)
16. Smets, P.: Belief functions. In: Non Standard Logics for Automated Reasoning, pp. 253–286 (1988)
17. Smets, P.: The Transferable Belief Model and other interpretations of Dempster-Shafer's model. In: Uncertainty in Artificial Intelligence 6 Annual Conference on Uncertainty in Artificial Intelligence (UAI 1990), pp. 375–383. Elsevier Science, Amsterdam (1990)
18. Tessem, B.: Approximations for efficient computation in the theory of evidence. *Artificial Intelligence* 61(2), 315–329 (1993)
19. Yager, R.R.: On considerations of credibility of evidence. *International Journal of Approximate Reasoning* 7, 45–72 (1992)
20. Zadeh, L.: Review of Shafer's mathematical theory of evidence. *AI Magazine* 5, 81–83 (1984)
21. Zouhal, L., Denoeux, T.: An evidence-theoretic K-NN rule with parameter optimization. *IEEE Transactions on Systems, Man and Cybernetics, Part C* 28(2), 263–271 (1998)

Spatial Weighting for Bag-of-Features Based Image Retrieval

Chuanqian Wang¹, Baochang Zhang¹, Zengchang Qin^{1,2}, and Junyi Xiong^{1,*}

¹ Science and Technology on Aircraft Control Lab
School of Automation Science and Electrical Engineering,
Beihang University, Beijing, China
² Intelligent Computing and Machine Learning Lab
School of Automation Science and Electrical Engineering,
Beihang University, Beijing, China
xiong.junyi@hotmail.com

Abstract. Visual features extraction for large-scale image retrieval is a challenging task. Bag-of-Features (BOF) is one of most popular models and gains attractive performance. However, BOF intrinsically represents an image as an unordered collection of local descriptors based on the intensity information, which provides little insight into the spatial structure of the image. This paper proposes a Spatial Weighting BOF (SWBOF) model to extract a new kind of bag-of-features by using spatial information, which is inspired by the idea that different parts of an image object play different roles on its categorization. Three approaches to measure the spatial information, local variance, local entropy and adjacent blocks distance are extensively studied, respectively. Experimental results show that SWBOF significantly improves the performance of the traditional BOF method, and achieves the best performance on the Corel database to our best knowledge.

Keywords: image retrieval, Bag-of-Features (BOF), spatial weighting.

1 Introduction

Content-based image retrieval (CBIR) is used for retrieving similar images from an image database. It has gained increasing attentions in recent years [1, 2]. The most challenging aspect of CBIR is to bridge the gap between low-level feature layout and high-level semantic concepts. The most popular approach recently relies on a bag-of-features (BOF) representation of the image [3]. The idea is to quantize local invariant descriptors such as scale-invariant feature transform (SIFT) [4], into a set of visual words. The images are represented with the frequency vector of the visual words and a histogram intersection is used for efficient comparison of such BOFs.

Despite the success of the bag-of-features model in recent studies, one problem needs to be concerned. Since the bag-of-features approach represents an image as an orderless collection of local descriptors, the resulting vector-space model provides

* Corresponding author.

little spatial constitution of the image. We have seen some works aiming at incorporating spatial information and color information in the “bag-of-features” model, such as dividing an image into equal sized rectangular regions and computing visual word frequency from each region [5], combining different descriptors into BOF model, using Gaussian mixture model to provide spatial weighting for visual words, using multi-scale spatial grids for locally orderless description of features [6] and using color SIFT descriptors [7].

In this paper, we develop a new image representation method, which provides spatial weighting scheme for visual words by using various texture related measures contained in each sub-block. The idea is that the more dissimilar between pixels for a specific region is, more useful information is contained. Intrinsically, such regions usually contain key points that are on edges and corners, and these regions should be considered more discriminative. There are several ways to measure the amount of information of an image sub-block. In this paper, we calculate local variance and entropy in each block to assign weights for further visual words counting [8]. We compute the distance between adjacent blocks as the weight, which is also chosen to be a baseline method. The benchmark is exploited to evaluate the performance of the new weighting methods. In the proposed method, each image is first divided into equal sized blocks, then local variance, local entropy and adjacent blocks distance for each block is computed. The codebook was computed with traditional unordered bag-of-features model. The spatial weighting scheme is achieved by weighting corresponding visual words according to texture information. Experimental results demonstrate that the proposed method can filter the background part of the image object to some extend and so improve the performance of content-based image retrieval system.

The remainder of this paper is organized as follows. In Section 2, we describe the procedure to generate a lexicon of the features for an image data set. In Section 3, we introduce the spatial weighting scheme. Section 4 reports the experimental results of the proposed method and compares our approach to the traditional “bag-of-features” approach. We conclude the paper in Section 5.

2 Codebook of Features

The bag-of-features (BOF) method is mainly inspired by the bag-of-words (BOW) [9] concept, which has been used in text categorization. In the BOF model, each image is described by a set of orderless local descriptors such as SIFT (Scale-invariant Feature Transform) and SURF (Speeded Up Robust Features). recent research has demonstrated its effectiveness in image analysis [10, 11]. In our approach, as shown in Figure 1, salient points are equally chosen from an image, then salient point is described by a 128-dimensional SIFT descriptor. Although we have extracted visual descriptors from images, these visual tokens are still too diverse to be analyzed. It is necessary to quantize those variant visual tokens into uniform forms (visual words) to facilitate our analysis. Thus, we adopt the K-Means algorithm with the Euclidian distance as the distance measurement to cluster local features into groups, and take centers of each group as “visual words” and take the collection of these “visual words” as an codebook. After obtaining the codebook, each image can be represented by the BoF frequency histograms of the visual words of the codebook.

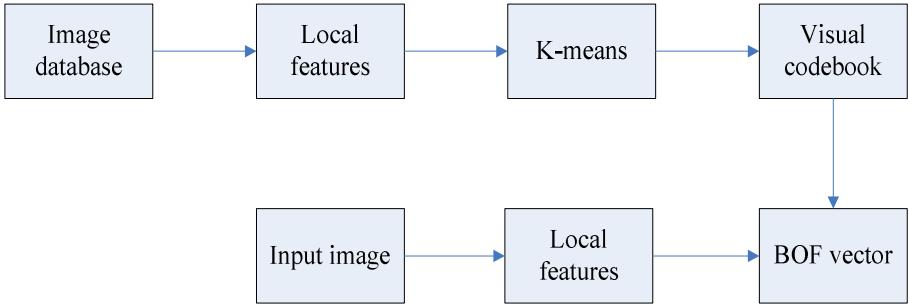


Fig. 1. Basic framework of unordered bag-of-features

3 Spatial Weighted Bag-of-Features

Unordered bag-of-features provides little information about the spatial constitution of the image; it takes background and foreground of an image as same. This may result in the decrease of the discrimination of the foreground. A batch of spatial weighting approaches has been proposed such as Gaussian mixture model (GMM) [12] and block based bag-of-features [13]. The spatial weighting schemes in this paper are aiming at diluting the discrimination of background and enhancing the foreground, insuring the resulting feature vector according to human eyes' perception. The basic framework is shown in Figure 2.

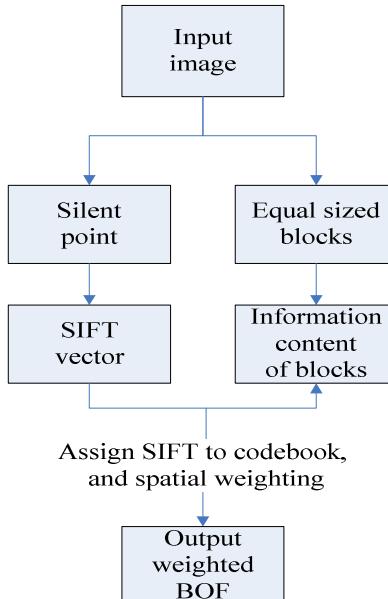


Fig. 2. Framework of spatial weighted BOF

3.1 Spatial Weighting Based on Local Variance

First, images are dividing into $n \times m$ equal sized rectangular regions and variances of gray value of pixels in each region are then counted. A matrix with $n \times m$ inputs for each image is constructed with these variances gained above. Then local descriptors of an image are assigned to each visual word in codebook with nearest neighbour scheme. And the spatial weighting scheme is achieved by weighting the visual words according to the local variance matrix.

$$W(i, j) = Var(G(i, j)) \quad (1)$$

Where $G(i, j)$ and $Var(G(i, j))$ are the gray-level histogram and variance of image blocks.

3.2 Image Sub-block Information Entropy

Shannon introduced a measure of information content [14]. The idea is that in constructing content of longer records, these messages will appear with different frequencies and according to patterns. These patterns dominate how much information each message actually carries. Shannon's entropy is defined as Equation (2).

$$H_n(p_1, p_2, \dots, p_n) = -\sum_{i=1}^n p_i \log_2 p_i \quad (2)$$

Where n is the number of separate symbols, in image n is the number of grayscales. p_i is the probability of symbol i , in image it contains the histogram counts from image grayscale histogram.

In our approach, we divide each image into $n \times m$ equal sized sub-regions, for each region, its entropy can be obtained by Equation (2) and an $n \times m$ matrix of entropy is constructed. Then the spatial weighting scheme can be achieved as local variance based spatial weighting.

3.3 Distance between Adjacent Sub-blocks Based Spatial Weighting

As above, images are divided into equal sized sub-blocks, and grayscale histogram of each block $h(i, j)$ is constructed. We define the distance between blocks is histogram intersection distance between the grayscale histogram of each block, shown as Equation (3).

$$D(h(i, j), h(l, k)) = 1 - \sum_{x=1}^d \min(a_x, b_x) \quad (3)$$

where a_x and b_x represents the frequencies of visual words of two blocks $b(i, j)$ and $b(l, k)$ respectively. For each block in a image, compute the distance between the block and each of its 8 neighbors (on its left-top, left-middle, left-bottom, right-top, etc.). Use the average of these eight distances as the weight of the block, i.e. as Equation (4). Then the spatial weighting scheme is the same as above.

$$W(i, j) = \frac{1}{8} \sum_{l=-1}^1 \sum_{k=-1}^1 D(i, j; i+l, j+k) \quad (4)$$

$$i = 0, \dots, n, j = 0, \dots, m$$

4 Experimental Results

In this section, we conduct an experiment of content-based image retrieval based on the proposed spatial weighting method on Corel database [15] and compare the results with the unordered “bag-of-features” approaches and the previously best reported results. Corel contains 1000 images in 10 categories. The images are with the size of either 256×384 or 384×256 . We choose codebooks with size of 200 and 1000, respectively, and divide each image into 16×24 or 24×16 blocks with the size of 16×16 , as shown in Figure 3. Table 1 shows the experimental results. We compare the spatial weighted BOF vector of dinosaurs with the unweighted BOF, as shown in Figure 4.

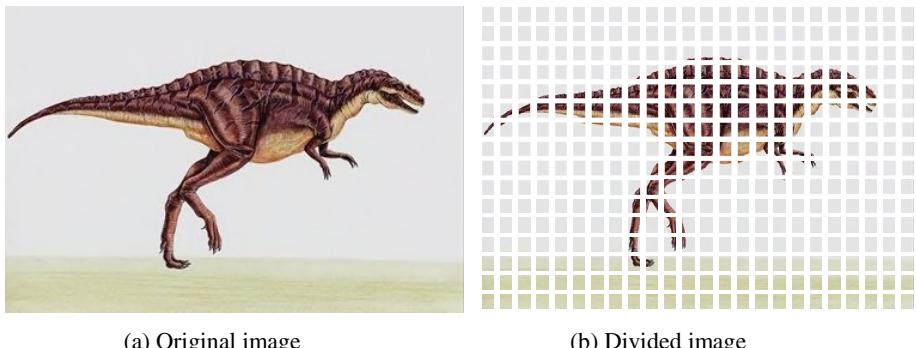


Fig. 3. Division of a sample image

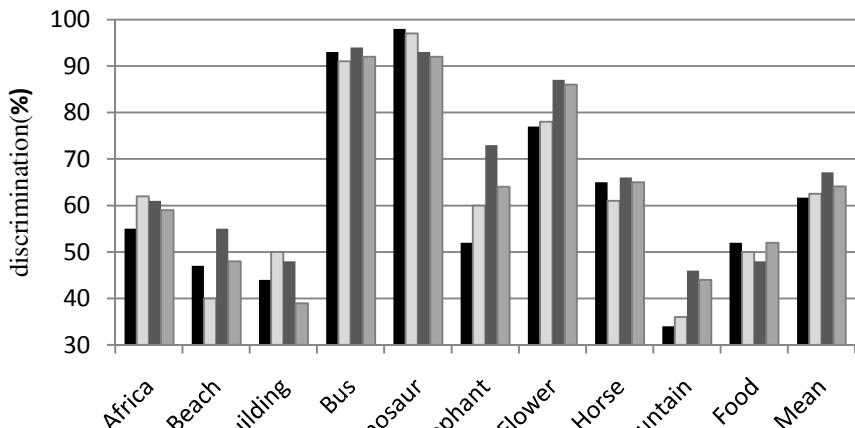
Table 1. Comparisons of average retrieval precision (ARP) obtained by proposed method to existing BoF systems [10]

(1) Codebook with size of 200

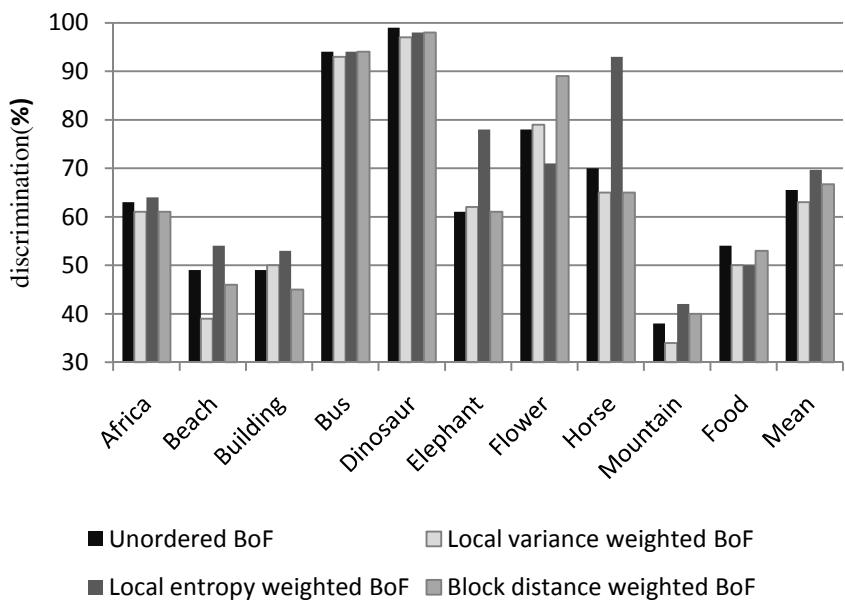
Method category \	Un-ordered BoF (%)	Local Variance Weighted BoF (%)	Local Entropy Weighted BoF (%)	Block Distance weighted BoF (%)
Africa	55	62	61	59
Beach	47	40	55	48
Building	44	50	48	39
Bus	93	91	94	92
Dinosaur	98	97	93	92
Elephant	52	60	73	64
Flower	77	78	87	86
Horse	65	61	66	65
Mountain	34	36	46	44
Food	52	50	48	52
Mean	61.7	62.5	67.1	64.1

(2) Codebook with size of 1000

Method category \	Un-ordered BoF (%)	Local Variance Weighted BoF (%)	Local Entropy Weighted BoF (%)	Block Distance weighted BoF (%)
Africa	63	61	64	61
Beach	49	39	54	46
Building	49	50	53	45
Bus	94	93	94	94
Dinosaur	99	97	98	98
Elephant	61	62	78	61
Flower	78	79	71	89
Horse	70	65	93	65
Mountain	38	34	42	40
Food	54	50	50	53
Mean	65.5	63	69.7	66.7

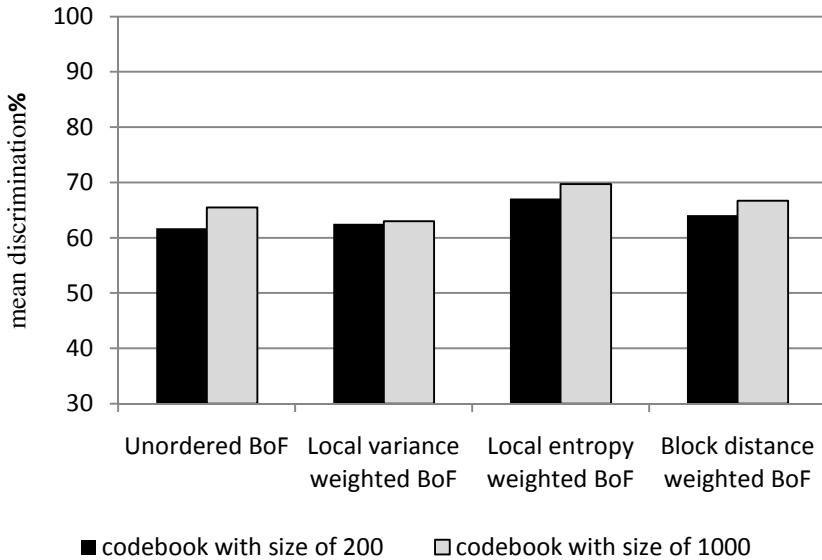


(1) Codebook with size of 200



(2) Codebook with size of 1000

Fig. 4. Comparisons of average retrieval precision (ARP) obtained by proposed method to existing BOF systems



(3) Comparison of mean discrimination with different codebook sizes

Fig. 4. (continued)

The experiment results show that, our spatial weighting scheme generally improves the performance of BOF. However, we can also find that our approach can not get a consistent better performance on the set of dinosaurs. We can find that the images in category of dinosaurs are typically with simple background, while images in other categories are with complex background. The original BOF have biggest output on the word generally representing the flat area as shown in Figure 5, which is obviously the background part of the Figure 3 (a). This means that the original BOF relies on the background part to be classified. It is not as effective as our proposed method, which can filter the background from the foreground, and so get a much better performance on other test sets, considering that natural images are generally with complex background.

In practical applications, there could be more than one category images with simple background and the discrimination of original BOF mentioned above will not exist. Therefore, it is not reasonable to rely on particularity of background for retrieval. But it is also unwise to discard the information of background. In order to solve this problem, a cascade retrieval scheme can be used. Simply, the variance of simple background is generally small, so we can categorize the images into two subclasses: simple or complex background, by local variance. And then in each group, we use proposed spatial weighed BOF for retrieval respectively.

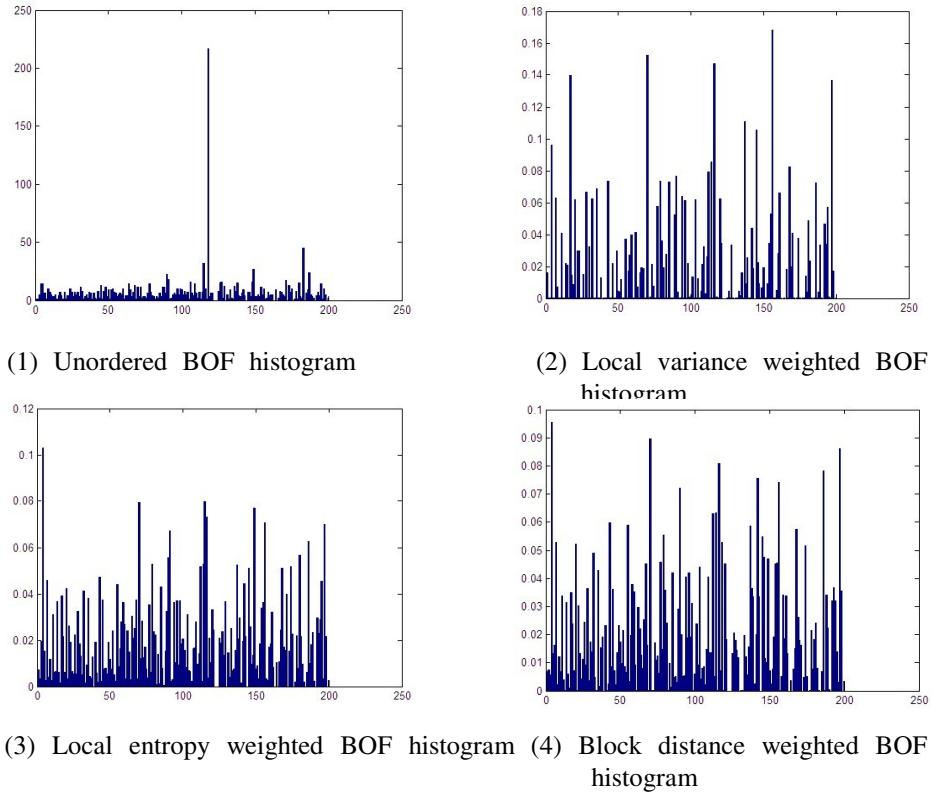


Fig. 5. BoF histogram of the dinosaur image

5 Conclusion

In this paper, a novel but simple approach of spatial weighting for Bag-of-Features based image retrieval is proposed, i.e. using the information content in sub-blocks of an image as the weight of each block. Three ways to measure the information content named local variance, local entropy and adjacent blocks distance respectively are studied in this paper. Our scheme to certain extend offsets the disadvantage of orderless BOF for discarding the spatial information, and significantly improve the performance of BOF in image retrieval. After comparing the result, a cascade retrieval approach is proposed; conducting experiment with the method on a proper image database will be our future work. And from the result, we can find that three ways of measuring the information content have their advantages and disadvantages, how to combine them to improve the performance will be our future work.

Acknowledgments. This work was supported in part by the Natural Science Foundation of China, under Contracts 60903065, 61039003 and 61272052, in part by

the Ph.D. Programs Foundation of Ministry of Education of China, under Grant 20091102120001, in part by the Fundamental Research Funds for the Central Universities.

References

1. Joshi, D., Li, J., Wang, J., Datta, R.: Image retrieval: ideas, influences, and trends of the new age. In: Proceedings of the 7th ACM SIGMM on Multimedia Information Retrieval (2005)
2. Pujari, J., Hiremath, P.S.: Content based image retrieval using color, texture and shape features. In: Proceedings of the International Conference on Advanced Computing and Communications, pp. 780–784 (2007)
3. Sivic, J., Zisserman, A.: Video google: A text retrieval approach to object matching in videos. In: Proc. of 9th IEEE Int'l Conf. on Computer Vision, vol. 2, pp. 1470–1477 (2003)
4. Lowe, D.G.: Object recognition from local scale-invariant features. In: ICCV, pp. 1150–1157 (1999)
5. Yang, J., Jiang, Y.G., Hauptmann, A.G., Ngo, C.W.: Evaluating bag-of-visual-words representations in scene classification. In: ACM SIGMM Int'l Workshop on Multimedia Information Retrieval (MIR 2007), Augsburg, Germany (September 2007)
6. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: Proc. CVPR (2006)
7. van de Weijer, J., Schmid, C.: Coloring local feature extraction. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3952, pp. 334–348. Springer, Heidelberg (2006)
8. Zhang, H., Fritts, J.E., Goldman, S.A.: An entropy-based objective evaluation method for image segmentation. In: Storage and Retrieval Methods and Applications for Multimedia, pp. 38–49 (2004)
9. Tian, Q., Zhang, S.: Descriptive visual words and visual phrases for image applications. In: ACM Multimedia, pp. 19–24 (2009)
10. Yuan, X., et al.: A SIFT-LBP retrieval model based on bag of features. In: 18th IEEE International Conference on Image Processing (2011)
11. Liu, C., Yuen, J., Torralba, A.: SIFT flow: Dense correspondence across scenes and its applications. IEEE Transactions on Pattern Analysis and Machine Intelligence 33(5) (2011)
12. Chen, X., Hu, X., Shen, X.: Spatial weighting for bag-of-visual-words and its application in content-based image retrieval. In: Theeramunkong, T., Kjirsirikul, B., Cercone, N., Ho, T.-B. (eds.) PAKDD 2009. LNCS, vol. 5476, pp. 867–874. Springer, Heidelberg (2009)
13. Li, Z., Imai, J., Kaneko, M.: Robust face recognition using block-based bag of words. In: International Conference on Pattern Recognition (2010)
14. Cherry, E.C.: Discussion on dr. Shannon's papers. IEEE Transactions on Information Theory 1(1), 169 (1953)
15. <http://wang.ist.psu.edu/jwang/test1.tar>

A Random Set and Rule-Based Regression Model Incorporating Image Labels

Guanyi Li and Jonathan Lawry

Intelligent System Laboratory, Department of Engineering Mathematics,
University of Bristol, Bristol, BS8 1UB, UK
peerinle@gmail.com, J.Lawry@bristol.ac.uk

Abstract. A new form of conditional rules is proposed for regression problems in which a rule associates an input label with a corresponding image label on the output space. Here input labels are interpreted in terms of random set and prototype theory, so that each label is defined by a random set neighbourhood around a prototypical value. Within this framework we propose a rule learning algorithm and test its effectiveness on a number of benchmark regression data sets. Accuracy is compared with other several state-of-the-art regression algorithms, suggesting that our approach has the potential to be an effective rule learning methodology.

Keywords: Prototype theory, image labels, random set, rule based learning.

1 Introduction

Takagi-Sugeno rules [11] are a popular and effective rule form which is widely applied in fuzzy systems. A significant feature of these rules is that fuzzy labels only appear in the antecedent of the rule whilst the consequent identifies a particular linear function. However, this feature then means that Takagi-Sugeno systems are less like a set of logical rules and more like a fuzzy piecewise linear approximation function. In this paper, we propose a new type of rules which can be applied to forecasting and regression problems, and which preserve a natural logical structure whilst incorporating an explicit model of the underlying functional mapping. The proposed model is based on the prototype and random set interpretation of label semantics [6], [7], as described in Lawry and Tang [8]. In this approach labels are interpreted as uncertain neighbourhoods of a prototype (or set of prototypes) within an underlying conceptual space. The essence of this current proposal is to approximate, from a database, the images of these neighbourhoods under the functional mapping. These image sets are then taken to be models of a set of image labels describing the output space of the function. In this context a rule simply associates a input label with its corresponding image label on the output space.

An outline of the paper is as follows: Section 2 gives an overview of the prototype and random set model of fuzzy labels proposed in [8]. Section 3 introduces

the new rule structure and proposes a natural inference method. Section 4 describes a new rule induction algorithm for learning rules of this kind from data. Section 5 gives experimental results comparing the new rule-based model with several well-known regression algorithms on benchmark datasets. Finally, section 6 gives conclusions and outlines future work.

2 A Random Set and Prototype Theory Interpretation of Label Semantics

Label semantics is an epistemic theory of uncertainty for vague concepts based on appropriateness measures and mass functions. The underlying calculus is not truth-functional but can be functional in a weaker sense, with the min and max rules for conjunction and disjunction being preserved for a strict class of expression. Here we focus on an interpretation of label semantics, based on random set theory and prototype theory.

The basic formulation of label semantics is as follows: We assume an underlying metric space Ω and a finite set of labels $LA = \{L_1, \dots, L_n\}$ for describing elements of Ω . For the regression modelling problems in this paper we will assume that $\Omega = \mathbb{R}^k$, for $k \geq 1$, and that distance is quantified by the standard Euclidean norm. For each label we then define an *appropriateness measure* $\mu_{L_i} : \mathbb{R}^k \rightarrow [0, 1]$ according to which $\mu_{L_i}(x)$ quantifies the uncertainty that label L_i is appropriate to describe element x . In accordance with Lawry and Tang's [8] model, labels L_i are defined by random set neighbourhoods from which μ_{L_i} are derived as single point coverage functions.

Definition 1. Random Set Neighbourhoods for Labels

Each label $L_i \in LA$ is associated with a prototype $p_i \in \mathbb{R}^k$ and a random variable ϵ_i into \mathbb{R}^+ , with density function δ_i . ϵ_i corresponds to an uncertain threshold according to which if an element x is closer to p_i than ϵ_i , then it is deemed sufficiently close to the prototype for the label L_i to be an appropriate description. Hence, we can naturally define a neighbourhood of p_i corresponding to those elements of \mathbb{R}^k which can be appropriately described as L_i :

$$\mathcal{N}_{L_i}^{\epsilon_i} = \{x \in \mathbb{R}^k : \|x - p_i\| \leq \epsilon_i\}$$

Since ϵ_i is uncertain, then $\mathcal{N}_{L_i}^{\epsilon_i}$ is a random set taking subsets of \mathbb{R}^k as values. We then take the appropriateness measure of L_i as corresponding to the single point coverage function of this random set as follows:

Definition 2. Appropriateness Measures

For all $L_i \in LA$, and $x \in \mathbb{R}^k$ let

$$\begin{aligned} \mu_{L_i}(x) &= \delta_i(\{\epsilon_i \in \mathbb{R}^+ : x \in \mathcal{N}_{L_i}^{\epsilon_i}\}) = \delta_i([\|x - p_i\|, \infty)) = \Delta_i(\|x - p_i\|) \\ &\text{where } \Delta_i(d) = \int_d^\infty \delta_i(\epsilon) d\epsilon \end{aligned}$$

3 Conditional Rules Associating Labels and Image Labels

Consider a regression (forecasting) problem where we have an underlying but unknown functional mapping $f : \mathbb{R}^k \rightarrow \mathbb{R}$. Knowledge of f then takes the form of a database of input-output values $DB = \{(x^j, y^j) : j = 1, \dots, N\}$. Based on the above random set model of labels we now introduce a new form of fuzzy rules which associate input labels L_i with a corresponding image label under f , denoted $f(L_i)$.

Recall that Takagi-Sugeno rules [11] have the following form:

$$\text{IF } x \text{ is } L_i \text{ THEN } y \text{ is } f_i(x)$$

where f_i is a linear function. In contrast we propose a rule base consisting of conditional rules of the alternative form:

$$\text{IF } x \text{ is } L_i \text{ THEN } y \text{ is } f(L_i)$$

Here we define a set of image labels $f(LA) = \{f(L_1), \dots, f(L_n)\}$ describing the output variable y . Furthermore, given the neighbourhood model of input label L_i , then if the underlying function f was known, it would be natural to associate the corresponding image label with the image of this neighbourhood under f . More formally;

Definition 3. Image Labels

The image label $f(L_i)$ is defined by the following image set, corresponding to an appropriateness neighbourhood of $f(L_i)$:

$$\mathcal{N}_{f(L_i)}^{\epsilon_i} = f(\mathcal{N}_{L_i}^{\epsilon_i}) = \{f(x) : x \in \mathcal{N}_{L_i}^{\epsilon_i}\}$$

In fact the definition of $f(L_i)$ given above, does correspond to the definition of a label for the output space, consistent with the random set neighbourhood model of labels. To see this, notice the following: Let f^{-1} be the inverse of f so that $f^{-1}(y) = \{x : f(x) = y\}$ then according to definition 3 we have that:

$$\begin{aligned} \mathcal{N}_{f(L_i)}^{\epsilon_i} &= \{y : \exists x \in f^{-1}(y), \|x - p_i\| \leq \epsilon_i\} \\ &= \{y : \min(\{\|x - p_i\| : x \in f^{-1}(y)\}) \leq \epsilon_i\} \\ &= \{y : d_{\min}(f^{-1}(y), p_i) \leq \epsilon_i\} \end{aligned}$$

where for $S \in \mathbb{R}^k$ $d_{\min}(S, x) = \min(\{\|x - s\| : s \in S\})$

Hence, we see that $f(L_i)$ can be defined by prototype $f(p_i)$ and distance threshold ϵ_i , but relative to a pseudo-distance d on the output space which satisfies $d(y, f(p_i)) = d_{\min}(f^{-1}(y), p_i)$. Furthermore, the following result shows that provided f is a continuous function then the neighbourhood of $f(L_i)$ is a closed interval of \mathbb{R} .

Theorem 1. If $f : \mathbb{R}^k \rightarrow \mathbb{R}$ is a continuous function then $\forall \epsilon_i \in \mathbb{R}^+$, $\mathcal{N}_{f(L_i)}^{\epsilon_i} = [l_i^{\epsilon_i}, u_i^{\epsilon_i}]$ where $l_i^{\epsilon_i} = \min\{f(x) : x \in \mathcal{N}_{L_i}^{\epsilon_i}\}$ and $u_i^{\epsilon_i} = \max\{f(x) : x \in \mathcal{N}_{L_i}^{\epsilon_i}\}$.

Proof. Since $\mathcal{N}_{L_i}^{\epsilon_i}$ is closed, then $\mathcal{N}_{f(L_i)}^{\epsilon_i} = f(\mathcal{N}_{L_i}^{\epsilon_i})$ is also closed, since the image of a closed set under a continuous function is closed. It therefore only remains to show that $\mathcal{N}_{f(L_i)}^{\epsilon_i}$ is convex. For $y, y' \in \mathcal{N}_{f(L_i)}^{\epsilon_i}$ suppose that $y < y'$. Let $x \in f^{-1}(y) \cap \mathcal{N}_{L_i}^{\epsilon_i}$ and $x' \in f^{-1}(y') \cap \mathcal{N}_{L_i}^{\epsilon_i}$. Since, f is a function it follows that $f^{-1}(y) \cap f^{-1}(y') = \emptyset$ and hence $x \neq x'$. Now consider the line segment between x and x' . The function f restricted to this line segment is a continuous function taking values y and y' , at x and x' respectively. Therefore, by the intermediate value theorem, it holds that for any $y'' \in [y, y']$ there exists x'' lying on the line segment for which $f(x'') = y''$. Now since $\mathcal{N}_{L_i}^{\epsilon_i}$ is convex then $x'' \in \mathcal{N}_{L_i}^{\epsilon_i}$ and consequently $y'' \in \mathcal{N}_{f(L_i)}^{\epsilon_i}$ as required.

Definition 4. Appropriateness Measure for Image Labels

We define the appropriateness measure $\mu_{f(L_i)} : \mathbb{R} \rightarrow [0, 1]$ for the image label $f(L_i)$ as follows: $\forall y \in \mathbb{R}$,

$$\mu_{f(L_i)}(y) = \delta_i(\{\epsilon_i : y \in \mathcal{N}_{f(L_i)}^{\epsilon_i}\})$$

The following now shows how the appropriateness measure for $f(L_i)$ can be determined directly from the appropriateness measure for L_i .

Theorem 2. For $f : \mathbb{R}^k \rightarrow \mathbb{R}$ a continuous function and Δ_i strictly decreasing we have that: $\forall y \in \mathbb{R}$,

$$\mu_{f(L_i)}(y) = \max\{\mu_{L_i}(x) : x \in f^{-1}(y)\}$$

Proof. From the above we have that

$$\mu_{f(L_i)}(y) = \Delta_i(d_{\min}(f^{-1}(y), p_i)) = \Delta_i(\min\{\|x - p_i\| : x \in f^{-1}(y)\})$$

Furthermore, since Δ_i is strictly decreasing,

$$= \max\{\Delta_i(\|x - p_i\|) : x \in f^{-1}(y)\} = \max\{\mu_{L_i}(x) : x \in f^{-1}(y)\}$$

as required.

Notice that if $\mu_{L_i} : i = 1, \dots, n$ are interpreted as fuzzy membership functions then theorem 2 corresponds to the well-known extension principle [14].

3.1 Inference from a Rule-Base

Suppose we have a set of rules:

$$\text{IF } x \text{ is } L_i \text{ THEN } y \text{ is } f(L_i) \text{ for } i = 1, \dots, n$$

modelling the function $f : \mathbb{R}^k \rightarrow \mathbb{R}$. Given an input $x' \in \mathbb{R}^k$ what can we infer from the rule-base about the value of $y' = f(x')$? Now each of these rules represent the strict (classical) implication $x' \in \mathcal{N}_{L_i}^{\epsilon_i} \rightarrow y' \in \mathcal{N}_{f(L_i)}^{\epsilon_i}$.

Consequently, $\mu_{f(L_i)}(y') \geq \mu_{L_i}(x')$ for $i = 1, \dots, n$. Hence, adopting the standard α -cut notation for fuzzy sets, so that $f(L_i)_\alpha = \{y : \mu_{f(L_i)}(y) \geq \alpha\}$, we have that:

$$y' \in \bigcap_{i=1}^n f(L_i)_{\mu_{L_i}(x')}$$

The following result show that the above constraint on y' can also be expressed directly in terms of neighbourhood functions.

Theorem 3. *For Δ_i strictly decreasing and $x' \in \mathbb{R}^k$ it follows that:*

$$\mathcal{N}_{f(L_i)}^{\|x' - p_i\|} = f(L_i)_{\mu_{L_i}(x')}$$

Proof. Recall that:

$$\begin{aligned}\mathcal{N}_{f(L_i)}^{\epsilon_i} &= \{y : d_{\min}(f^{-1}(y), p_i) \leq \epsilon_i\} \text{ and hence} \\ \mathcal{N}_{f(L_i)}^{\|x' - p_i\|} &= \{y : d_{\min}(f^{-1}(y), p_i) \leq \|x' - p_i\|\}\end{aligned}$$

Now,

$$\begin{aligned}y \in f(L_i)_{\mu_{L_i}(x')} \text{ iff } \mu_{f(L_i)}(y) \geq \mu_{L_i}(x') \text{ iff } \Delta_i(d_{\min}(f^{-1}(y), p_i)) \geq \Delta_i(\|x' - p_i\|) \\ \text{ iff } d_{\min}(f^{-1}(y), p_i) \leq \|x' - p_i\| \text{ iff } y \in \mathcal{N}_{f(L_i)}^{\|x' - p_i\|}\end{aligned}$$

as required.

Hence, given an input x' we can infer that the output $y' \in \bigcap_{i=1}^n \mathcal{N}_{f(L_i)}^{\|x' - p_i\|}$.

Notice, that since by theorem 1 we have that $\mathcal{N}_{f(L_i)}^{\|x' - p_i\|}$ is a closed interval of \mathbb{R} for $i = 1, \dots, n$, then given input x' the rule-base restricts y' to a closed interval of \mathbb{R} . However, in many applications a precise estimate of y' is required. In the following sub-section we propose a defuzzification method for identifying a single value.

3.2 Defuzzification

Initially we consider an idealized defuzzification method in which the underlying function f is known. We then generalize this to the case in which our only knowledge of f is in the form of a database DB of input-output examples. Recall that given an input $x' \in \mathbb{R}^k$ we can infer that $y' \in \mathcal{N}_{L_i}^{\epsilon'_i} = [l_i^{\epsilon'_i}, u_i^{\epsilon'_i}]$ where $\epsilon'_i = \|x' - p_i\|$. Hence, we propose a defuzzification method based on a weighted combination of approximations of y' from each rule. Furthermore, for the rule with antecedent L_i we take that approximation as a point in $[l_i^{\epsilon'_i}, u_i^{\epsilon'_i}]$ dependent on the gradient of the function f at p_i as follows: Let $\mathbf{v}_i = \nabla f|_{x=p_i}$ and then we take:

$$w_i(x') = \frac{(x' - p_i) \cdot \mathbf{v}_i}{\|x' - p_i\| |\mathbf{v}_i|}$$

We now use $w_i(x')$ to determine the approximation of y' based on rule i as follows:

$$y_i = \begin{cases} f(p_i) + w_i(x')(f(p_i) - l_i^{\epsilon_i}) : w_i(x') \leq 0 \\ f(p_i) + w_i(x')(u_i^{\epsilon_i} - f(p_i)) : w_i(x') > 0 \end{cases}$$

From the above estimates for each rule we then obtain an overall estimate for the output value as follows:

$$y' \approx \sum_{L_i: \mu_{L_i}(x') > 0} P(L_i|x') y_i$$

Here $P(L_i|x')$ is determined from applying the Pignistic possibility-probability transformation [3] to the possibility distribution on LA generated by $\mu_{L_i}(x')$ for $i = 1, \dots, n$.

The above defuzzification method presupposes perfect knowledge of the underlying function f . In reality our knowledge of f will be restricted to a set of input-output examples. Therefore, we now consider how to adapt the above procedure to this case. Essentially, this reduces to the problem of estimating the following from the database DB : $f(p_i)$, $\mathcal{N}_{f(L_i)}^{\epsilon_i}$ and $\mathbf{v}_i = \nabla f|_{x=p_i}$. In the sequel we will propose a method for inferring the image label appropriateness measure $\mu_{f(L_i)}$ from data. Given this we can then determine:

$$\begin{aligned} f(p_i) &\approx \arg \max \{\mu_{f(L_i)}(y) : y \in \mathbb{R}\} \\ \text{and } \mathcal{N}_{f(L_i)}^{\epsilon_i} &\approx f(L_i)_{\mu_{L_i}(x')} \end{aligned}$$

We now propose a gradient search algorithm to estimate \mathbf{v}_i , as follows:

```

 $(x', y')$  such that  $y' = \min(y : (x, y) \in DB, x \in \mathcal{N}_{L_i}^{\Delta^{-1}(\alpha)})$ 
 $y_{\max} = \max(y : (x, y) \in DB, x \in \mathcal{N}_{L_i}^{\Delta^{-1}(\alpha)})$ 
 $V_i(0) \leftarrow p_i - x'$ 
 $j \leftarrow 1$ 
while  $y' \leq y_{\max}$  do
  if  $y \leq f(p_i)$  then
     $(x, y)$  such that  $y = \operatorname{argmin}(y : (x, y) \in DB, \mu_{L_i}(x) \geq \mu_{L_i}(x'))$ 
     $V_i(j) \leftarrow (p_i - x) \cdot \mu_{L_i}(x)$ 
  else
     $(x, y)$  such that  $x = \operatorname{argmax}(\mu_{L_i}(x) : (x, y) \in DB, y > y')$ 
     $V_i(j) \leftarrow (x - p_i) \cdot \mu_{L_i}(x)$ 
  end if
   $(x', y') \leftarrow (x, y)$ 
   $j \leftarrow j + 1$ 
end while

```

Algorithm 1: Gradient Search Algorithm

In the above algorithm, α is a lower bound on the appropriateness measure of the points to be included, and here we take $\alpha = 0.1$. We then estimate \mathbf{v}_i according to:

$$\mathbf{v}_i = \sum_{r=0}^{j-1} V_i(r)$$

4 Rule Learning

In this section we propose an algorithm for learning a rule base of the form:

$$\text{IF } x \text{ is } L_i \text{ THEN } y \text{ is } f(L_i) \text{ for } i = 1, \dots, n$$

from a database of input-output values $DB = \{(x^j, y^j) : j = 1, \dots, N\}$ where $x^j \in \mathbb{R}^k$ and $y^j \in \mathbb{R}$. The number of rules n is determined by the number of input labels and form one of two learning parameters. The prototypes $p_i \in \mathbb{R}^k$ for $i = 1, \dots, n$ are determined by applying *k-means* clustering to DB and then projecting the resulting cluster centres onto the input space \mathbb{R}^k .

4.1 Gaussian Appropriateness Measures

According to the neighbourhood interpretation of labels, each label has an associated threshold random variable with distribution δ_i . Here and in the sequel we take δ_i to be a normalized Gaussian distribution with mode 0 and standard deviation σ_i so that:

$$\delta_i(\epsilon_i) = \frac{2}{\sqrt{2\pi}\sigma_i} \exp \frac{\epsilon_i^2}{-2\sigma_i^2}$$

This gives us that:

$$\Delta_i(d) = 1 - \operatorname{erf}\left(\frac{d}{\sqrt{2}\sigma_i}\right) \text{ so that } \mu_{L_i}(x) = 1 - \operatorname{erf}\left(\frac{\|x - p_i\|}{\sqrt{2}\sigma_i}\right)$$

Notice that in this case Δ_i is a strictly decreasing function.

Given a set of prototypes $p_i \in \mathbb{R}^k$ for $i = 1, \dots, n$ determined using clustering as outlined above, we now fix σ_i such that:

$$\Delta_i(\min\{\|p_i - p_j\| : j \neq i\}) = 1 - \tau \text{ for } \tau \in [0, 1]$$

Here the parameter τ controls the degree to which neighbouring labels overlap, and together with the number of rules n , is the second parameter for the proposed rule-induction algorithm.

4.2 Estimating Appropriateness Measures for Image Labels

We now propose to use asymmetric Gaussian functions in order to approximate $\mu_{f(L_i)}$ given the database DB . Asymmetric Gaussian functions are parameterised by three values; a mode c and two width parameters σ_l and σ_r . The definition is as follows:

$$\mu_{f(L_i)}(y) = AsyGau(y, c, \sigma_l, \sigma_r) = \begin{cases} \exp\left(-\frac{(c-y)^2}{2 \cdot \sigma_l^2}\right) & y < c \\ \exp\left(-\frac{(y-c)^2}{2 \cdot \sigma_r^2}\right) & y \geq c \end{cases}$$

Now from theorem 4 it follows that if $y = f(x)$ then $\mu_{f(L_i)}(y) \geq \mu_{L_i}(x)$. Hence, we propose to find the minimal asymmetric Gaussian such that c is the mode of the values $\{y : (x, y) \in DB, \mu_{L_i}(x) > \alpha\}$, and subject to the constraint that for all such y values in the database $\mu_{f(L_i)}(y) \geq \max\{\mu_{L_i}(x) : (x, y) \in DB\}$. This is achieved by setting the parameters of the asymmetric Gaussian as follows:

$$\begin{aligned} c &= \text{argmax}\{\mu_{L_i}(x) : (x, y) \in DB\} \\ \sigma_l &= \max(\{\sqrt{-\frac{(c-y)^2}{2 \cdot \log(\mu_{L_i}(x))}} : (x, y) \in DB, \mu_{L_i}(x) > \alpha, y < c\}) \\ \sigma_r &= \max(\{\sqrt{-\frac{(y-c)^2}{2 \cdot \log(\mu_{L_i}(x))}} : (x, y) \in DB, \mu_{L_i}(x) > \alpha, y \geq c\}) \end{aligned}$$

Recall that $\alpha \in [0, 1)$ is a threshold governing the minimal appropriateness values for L_i which we consider. Typically we take $\alpha = 0.1$.

5 Experimental Results

In this section, we present experimental results for a number of benchmark datasets to demonstrate the performance of the proposed rule-learning algorithm. Overall results on various regression problems show both good accuracy and transparency. An outline of this section is as follows. Section 5.1 gives a brief description of the different datasets used in our experiments. Section 5.2 then shows the test results of these datasets. Finally, the Sunspot prediction problem will be discussed in detail in section 5.3.

5.1 Description of Data Sets

In this section we give a brief description of the datasets we have used in our experiments.

Abalone and Boston Housing

These two datasets are taken from the UCI repository [1]. The Abalone database concerns the problem of predicting the age of Abalone from physical measurements. Abalone are a type of shellfish, the age of which can be accurately determined by cutting the shell through the cone, staining it, and counting the number of rings through a microscope, which is a laborious and time consuming task. Boston Housing contains data on housing values in the suburbs of Boston, USA. The data set contains 506 instances and 13 continuous attributes (including the target attribute) and one binary attribute.

To test our algorithm the Abalone data set of 4,177 examples was randomly split into a test set of 2,089 instances and a training set of 2,088 instances while for the Boston Housing problem we split the 506 examples into 253 training examples and 253 for testing at random. In each case we repeated the experiment 10 times with random splits of the data, so as to obtain information not just on the predictive accuracy but also the stability of the methods. The learning parameters were then set-up as follows: for Abalone, we set $\tau = 0.9$ and number of rules to $n = 5$ and for Boston Housing, we set $\tau = 0.9$ and $n = 7$.

Mackey-Glass

Mackey-Glass is a time series generated from the following time delay differential equation:

$$\dot{x}(t) = \frac{0.2x(t - \rho)}{1 + x^{10}(t - \rho)} - 0.1x(t)$$

This time series is chaotic, and so there is no clearly defined period. The series will not converge or diverge, and the trajectory is highly sensitive to initial conditions. This is a benchmark problem in the neural network and fuzzy modelling research communities [5,10,12].

The data consisted of time series values at integer points, obtained by applying the fourth-order Runge-Kutta method in order to find the numerical solution to the above MG equation. Here we assumed that $x(0) = 1.2$, $\rho = 17$, and $x(t) = 0$ for $t < 0$.

In this time series prediction the objective is to use known values of the time series up to the point in time t , in order to predict the future time point $t + 6$. For each t , the input training data for the model is a four dimensional vector of the form, $\mathbf{x}(t) = \langle x(t - 18), x(t - 12), x(t - 6), x(t) \rangle$. The output training data corresponds to the trajectory prediction, $y(t) = x(t + 6)$. With t ranging from 118 to 1117, we obtain 1000 input/output data values. We use the first 500 data values for the training dataset DB , while the remaining values are used as test data. Here we set $\tau = 0.9$ and the number of rules is $n = 60$.

Sunspot Prediction

This problem is taken from the Time Series Data Library [4]. Sunspots, which are often considerably larger than the earth, were first discovered in about 1610 shortly after the invention of the telescope, and have an average life time of about 11 years, although this can vary between 7 and 15 years. Sunspots numbers

have been recorded since 1700 but no method has been found to accurately predict or determine when or why these events occur. It is known however, that Sunspot numbers are related to solar activity such as the magnetic field of the sun changing, which occurs about every 22 years. In this experiment we use Sunspot relative numbers between the years 1700 and 1979 which was organized as described in [9]. We use 209 examples (1712-1920) as the training data, and 59 examples (1921-1979) as the test data. The input attributes were $x(t-12)$ to $x(t-1)$ and the output attribute was $x(t)$ (i.e. one-year-ahead). For the sunspot prediction problem, we set $\tau = 0.9$ and the number of rules to $n = 6$.

5.2 Summary of Results

In this section we present some experimental results for the Abalone ,Boston housing, Mackey-Glass and Sunspot regression problems. We use Root Mean Squared Error (RMSE) to compare our algorithm with other learning algorithms.

Table 1. is a summary of the results from the experiments on the Boston Housing, Abalone, Mackey-Glass and Sunspot Prediction problems. Note that the results shown are the best obtained for *each* algorithm over a range of learning parameter values. In the sequel we will also investigate the parameter sensitivity of our rule-learning algorithm.

Table 1. Comparison of results for different learning algorithms from the Boston Housing, Abalone, Mackey-Glass and Sunspot prediction problems

Algorithm	Boston Housing	Abalone	Mackey-Glass	Sunspot
Regression Tree	4.6153	2.8889	0.0285	29.9144
Back. Pro. NN	4.3020	2.1765	0.02	32.0341
ε -SVR system	3.9260	2.1360	0.0098	20.4481
Image Label Rules	3.6229	2.4126	0.0087	18.5134

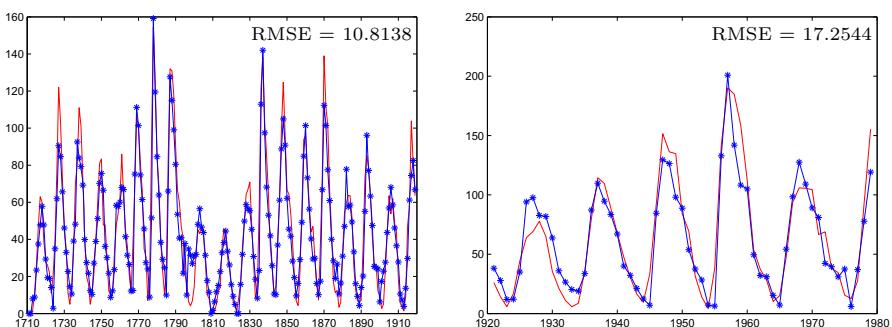


Fig. 1. Sunspots time-series results for the training data (left-hand figure) and for the test data (right-hand figure) with 6 rules: solid line is the actual output, dotted line is the predicted output

From Table 1 we can see that the image label rules perform well. Indeed the proposed algorithm compares well with three other well known regression algorithms: these are ε -Support Vector Regression system (ε -SVR) [13], implemented in the Matlab Support Vector Machine Toolkit (Version 3.1) [2], Back Propagation Neural Network (Matlab Neural Network Toolbox) and the Matlab implemented regression tree. In addition, figure 1 shows time series of predicted and actual results for the Sunspot problem on both the training and test data sets.

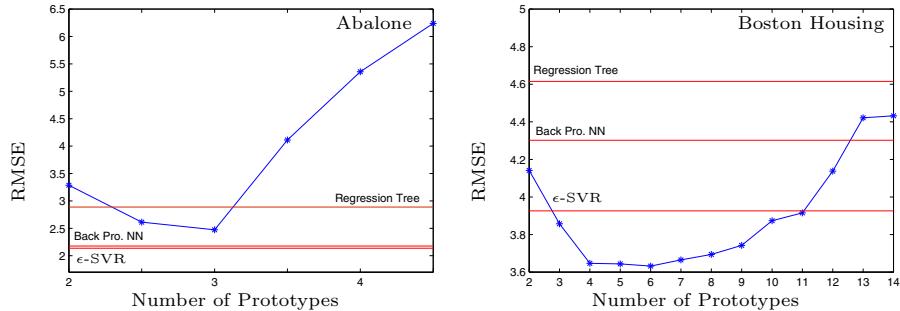


Fig. 2. These figures show sensitivity to the number of rules n for the Abalone and Boston Housing data sets. For all result we set $\tau = 0.9$. The horizontal axis shows the number of rules n , and the vertical axis shows the corresponding RMSE on the test set. The solid lines give the RMSE from Regression trees, Back Pro. NN and ε -SVR.

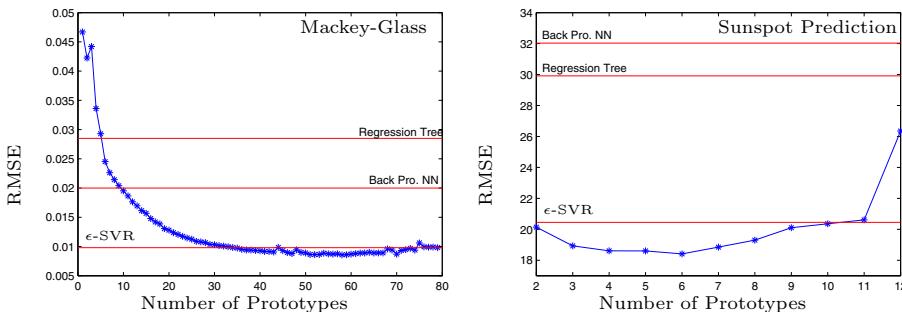


Fig. 3. These figures show sensitivity to the number of rules n for the Mackey-Glass and Sunspot data sets. For all result we set $\tau = 0.9$. The horizontal axis shows the number of rules n , and the vertical axis shows the corresponding RMSE on the test set. The solid lines give the RMSE from Regression trees, Back Pro. NN and ε -SVR.

Recall that the results in table 1 are obtained for certain fixed parameter values τ and n . We now investigate the sensitivity of our algorithm to these

parameters. In fact, here we focus on the number of rules n and set $\tau = 0.9$ as was the case for all results in table 1. We then vary the number of rules n between 3 and 8 for the Abalone data set, between 3 and 14 for Boston Housing, between 3 and 81 for Mackey-Glass, and between 3 and 12 for Sunspot. Different initializations of k -means can potentially result in different prototypes for the input labels, so in order to reduce the effect of this we computed the results for 10 different runs of k -means for both training and testing data. We then calculated the average RMSE across all of them. Figures 2 and 3 shows the sensitivity to prototype number n across the 4 data sets.

6 Conclusion and Future Work

In this paper we have introduced a new type of conditional rule for regression problems which associate input labels with corresponding image labels. These rules are formulated within the random set and prototype theory interpretation of fuzzy labels introduced in Lawry and Tang [8]. Within this framework we have proposed a rule-learning algorithm for inferring a rule-base of image label rules from data. We have then tested this new rule induction algorithm on a number of benchmark problems and compared its performance with that of several well-known regression algorithms. The results obtained are shown to be competitive with those of several state-of-the-art regression algorithms.

The proposed rule induction algorithm has two learning parameters; τ , controlling input label overlap, and n , the number of rules. Future work will explore the use of optimisation methods such as *genetic algorithms* to fix these parameter values.

References

1. Blake, C., Merz, C.J.: UCI machine learning repository,
<http://www.ics.uci.edu/~mlearn/MLRepository.html>
2. Chang, C., Lin, C.: LIBSVM: A library for support vector machines. ACM Transactions on Intelligent Systems and Technology 2, 27:1–27:27 (2011)
3. Smets, P.: The transferable belief model. Artificial Intelligence 66, 191–234 (1994)
4. Hyndman, R., Akram, M.: Time series data library,
<http://www-personal.buseco.monash.edu.au/hyndman/TSDL/index.htm>
5. Kim, D., Kim, C.: Forecasting time series with genetic fuzzy predictor ensemble. IEEE Transactions on Fuzzy Systems 5, 523–535 (1997)
6. Lawry, J.: A framework for linguistic modelling. Artificial Intelligence 155, 1–39 (2004)
7. Lawry, J.: Modelling and reasoning with vague concepts. Springer (2006)
8. Lawry, J., Tang, Y.: Uncertainty modelling for vague concepts: A prototype theory approach. Artificial Intelligence 173, 1539–1558 (2009)
9. Randon, N.J.: Fuzzy and random set based induction algorithms. PhD Thesis, University of Bristol (2004)
10. Russo, M.: Genetic fuzzy learning. IEEE Transactions on Evolutionary Computation 4, 259–273 (2000)

11. Takagi, T., Sugeno, M.: Fuzzy identification of systems and its applications to modeling and control. *IEEE Transactions on Systems, Man, and Cybernetics* 15, 116–132 (1985)
12. Tang, Y., Xu, Y.: Application of fuzzy naive bayes and a real-valued genetic algorithm in identification of fuzzy model. *Inf. Sci.* 169, 205–226 (2005)
13. Vapnik, V.N.: *The nature of statistical learning theory*. Springer (1995)
14. Zadeh, L.A.: The concept of a linguistic variable and its application to approximate reasoning - I. *Inf. Sci.* 8, 199–249 (1975)

Mining Statistically Significant Target mRNA Association Rules Based on microRNA

Feng Chen^{1,2,*}, Tingting Wang², Yu Wang¹, Shengfei Li¹, and Jing Wang³

¹ College of Information Science and Engineering, Henan University of Technology, China

² School of Engineering and Mathematical Sciences, La Trobe University, Australia

³ Puyang Branch Company, Henan Electric Power Company, China

chenfeng0312@gmail.com

Abstract. The relation of miRNA and mRNA are very important because miRNA can regulate almost all the biological process by cooperating with mRNA. However, the directed regulation among mRNA has not been concerned a lot. In this paper, we introduce association rule mining and hypothesis test to find the closely related mRNAs and their regulation direction based on their relation with miRNAs. Our research can further the understanding about miRNA and mRNA. Our results uncover the novel mRNA association patterns, which could not only help to construct the biological network, but also extend the application of association rule mining in bioinformatics.

Keywords: microRNA, miRNA, association rule, hypothesis test.

1 Introduction

Association rule mining [1-4] can indicate how much possible item B happens given item A happens, implying a potential directed regulation between A and B. It is an extremely important branch of data mining due to a variety of application, including bioinformatics and biology. For example, the regulation of miRNA (microRNA) and mRNA (message RNA) [5-10] can employ association rule for the discovery of a lot of novel biological knowledge.

miRNA, which is a kind of non-coding RNA, can regulate millions of biological processes among which its regulation for mRNA has raised a lot of concern [5-6]. mRNA, well-known as message RNA, transfers genetic information from DNA to protein [7-10]. So, miRNA can regulate and change biological phenomenon through regulating mRNA. The relation between miRNA and mRNA is extremely complicated so that there are a lot of issues remaining such as the sequence features, regulation mechanism, regulation environment and so forth. One of the most important issues is the determination of gene regulation mechanism by discovering mRNA regulation direction through miRNA. It has been proved that the regulation

* This paper is partly supported by NSFC (61203265, 61001099), Key Project of Henan Province (122102110106) and PhD Fund of Henan University of Technology (150121).

mechanism of mRNA and miRNA acts as a complicated and enormous network [8]. One miRNA can regulate a number of mRNAs while one mRNA can be regulated by diverse miRNAs. Therefore, it is impossible to verify the target mRNAs for every miRNA. Due to the difficulties, the very common methodology finds mRNA modules, or miRNA modules that imply which mRNAs or miRNAs cooperate as a whole to anticipate the regulation between miRNAs and mRNAs. However, this strategy remains a significant issue opening: regulation direction.

For a research concerning miRNA and mRNA, the regulation direction of them is a further step after finding the regulation module. The regulation between genes could be discovered once we can find the regulation direction between mRNAs because mRNA represents all the genetic information for genes. Further more, considering the close regulation of miRNA and mRNA, the directed regulation of mRNA based on miRNA is promising and theoretically sound.

In this paper, we have worked on finding the directed mRNA regulation based on miRNA. First, all the mRNAs are sorted according to their related miRNAs. Second, association rule mining is applied for the identification of the closely related mRNA and their regulation direction. Third, hypothesis test is taken to verify our results. We have conducted a plenty of experiments using PMDR (a plant database), proving that our method can not only find the closely related mRNAs, but also identify the potential regulation direction between them.

This paper is organized as follows: section 2 is related work. Section 3 details our method. Section 4 provides our experiments and analysis. Conclusion is section 5.

2 Related Work

Our work is the improvement and application of association rule mining in bioinformatics. So in this section, we introduce association rule first. Then the research about mRNA and miRNA is involved too. Last, the current combination of them is discussed.

In data mining, association rule learning is an important and well-known strategy for the discovery of the significant relations between items in datasets. For example, Rakesh Agrawal et al. [1] introduce association rules for finding regularities between products in large-scale transaction data. For this method, $A \rightarrow B$ means if a customer buys A, he or she is likely to also buy B. Such information can be employed as the basis for decisions about marketing activities. In addition, association rule mining can be taken in many other applications such as web usage, intrusion detection and bioinformatics. At the same time, the methodology of association rule mining has improved continuously. A variety of mining methods are developed for diverse objects, like quantitative association [3], hierarchical association [2] and weighted class mining [4].

Quantitative association rule categories the data into different types, then focuses on the relation between quantitative data. In other words, the traditional mining can find the relation between A and B while quantitative rule mining can identify how many As is related with how many Bs. Hierarchical association classifies the items

into different levels and find the relation between them. For example, “printer→HP” indicates if one customer buys a printer, how possible he or she would like to buy HP. Weighted class mining allocates a weight to classes to give focus to a particular issue of concern for the consumer of the data mining results. For instance, if one is interested in “bread”, then all the rules including bread will be given more weight and highlighted.

The research about mRNA and miRNA also involves diverse issues such as target mRNA discovery, miRNA module and miRNA network [5-10]. All the issues are unique but relevant to some extent. Since miRNA can regulate almost all the biological processes and a lot of diseases by regulating mRNA, these issues are all important and significant [5-7]. For example, target mRNA aims to find one or more specific mRNAs for one or more miRNAs. Generally, such kind methods use statistics to anticipate how possible one mRNA could be a target for a certain miRNA based on their biological sequence features. However, this method fails to find the complicated corresponding relation between mRNA and miRNA. miRNA module [9-10] means a cluster of miRNAs which either are similar functionally, or regulate the same biological process. For instance, the miRNA regulating one mRNA or occurring in lung cancer could be considered to be a module. The discovery of miRNA module helps much to find how miRNAs cooperate. miRNA network [8] can be thought of as an extension of the above two. It bridges proteins and miRNAs by mRNAs. By analyzing the functions of proteins, miRNA functions can be speculated. Simultaneously, this method takes miRNA and mRNA as a network, so that the complex of the regulation mechanism could be uncovered.

As data mining was introduced into bioinformatics, the association rule has been concerned more and more [11-12]. A frequent itemset indicates the items that are related closely to each other, which is very easy to be used for the regulation between genes. In the field of miRNA and mRNA, association rule mining is promising as well. For example, one target mRNA pattern includes one miRNA and one mRNA, which can be represented by a two-item frequent itemset. A miRNA module either is a set of two-item frequent itemsets, or indicates a multi-item one. Besides, these frequent itemsets can be theoretically linked to be a network, which may imply a regulation network involving miRNAs and mRNAs.

However, these methods do not use association rule to represent a directed miRNA and mRNA. In another word, the current methods may tell us two mRNAs (m_1 , m_2) co-work together. But they fail to show how they regulate each other. m_1 regulates m_2 , or the mutation of m_2 changes m_1 ? So in this paper, we will improve association rule mining in this field. By categorizing mRNAs based on miRNAs, we will find the mRNA association rules with direction, representing how mRNAs regulate. Our research can be extended to the speculation of the regulation of genes and the construction of miRNA regulation network.

3 Our Method

Our method includes three main steps, as shown in Fig. 1. First, we extract association rules from the transferred database. Every rule has support and confidence. For a

comprehensive study, any rule will not be filtered because of small support or confidence. Then, we employ two-dimensional kernel density estimate to combine support and confidence so that how significant a rule would be can be determined reasonably. Last, due to a fact that a fixed threshold has no specific meaning in biology, we introduce hypothesis test to classify the rules as significant and insignificant. During this process, α level is 0.5, as most statistical methods do.

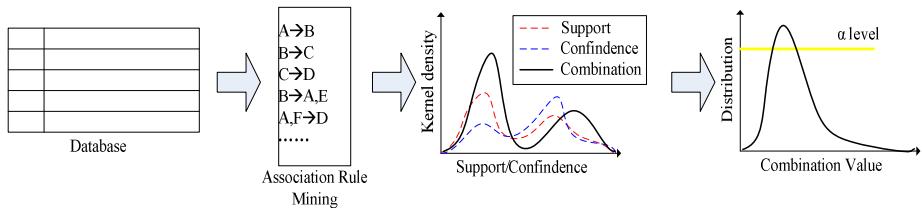


Fig. 1. The Outline of Our Method

3.1 The Application of Association Rule Mining

Following the original definitions [1], the issues in association rule mining are defined as follows:

Let $I = \{i_1, i_2, \dots, i_n\}$ be a set of binary attributes, which is called items. A set including more than one item is known as itemset. Let $D = \{t_1, t_2, \dots, t_m\}$ be a set of transactions, which is our database. Each transaction in D , that has a unique transaction ID, contains a subset of the items in I . A rule is like $X \rightarrow Y$ where $X, Y \subseteq I$ and $X \cap Y = \emptyset$.

For almost every association rule mining method, there are two very important parameters: support and confidence. The support of an itemset ($SUPP(X)$) is the proportion of transactions in the data set which contain the itemset. The confidence of a rule ($X \rightarrow Y$) is defined as $CONF(X \rightarrow Y) = SUPP(X \cup Y) / SUPP(X)$.

In our method, every miRNA is an ID, indicating a series of mRNAs that are regulated by it. In other words, an mRNA is like an item. The mRNA association rules will be extracted after using association rule mining. We will pay attention to two-item mRNA association rules for multi-item rules are very hard to be supported by biological evidence. This part is illustrated in the second sub-graph in Fig. 1. The pseudocode is shown as follows:

Association Rule Mining

Input: mRNA Transaction Database,min_SUPP,min_CONF

Output: mRNA Rules

$k=1$

For every k-itemset

If $SUPP > min_SUPP$

Mark it as frequent itemset

End

End

For every frequent itemset

If CONF>min_CONF

Mark it as significant rule

End

End

Extract all the significant rules

3.2 Kernel Estimate

From biological point of view, support means the mRNAs in one itemset cooperate more than expected while confidence indicates the regulation direction between them. Therefore, the mRNA rules with high support and confidence are supposed to be important. However, we expect more than this type. For example, a rule might imply a rare but close regulation if it has low support and high confidence. So, we have to find a criterion to combine support and confidence so that all the rules that we are interested in can be highlighted. For achieving this issue, we introduce kernel density estimate, which is shown in the third sub-graph in Fig. 1.

All the rules are sorted twice, according to the descending sequence of support and confidence respectively. For support sequence or confidence sequence, a multivariate kernel density estimate of a rule will be obtained by the following formulas:

$$\hat{f}(x) = \frac{1}{nh_1 h_2} \sum_{i=1}^n K_1\left(\frac{x - x_{1i}}{h_1}\right) K_2\left(\frac{x - x_{2i}}{h_2}\right) \quad (1)$$

$$K_1\left(\frac{x - x_{1i}}{h_1}\right) = \frac{1}{\sqrt{2\pi}} e^{-\frac{(x - x_{1i})^2}{h_1^2}} \quad (2)$$

In formula (1), $\hat{f}(x)$ is the kernel density estimate value for x . n is the number of mRNA association rules we extract. x represents support or confidence of a rule that we are analyzing. In K_1 , x and x_{1i} indicates support of the current rule and the others respectively. So does K_2 . K_1 or K_2 is defined as formula (2). h (including h_1 and h_2) is a smooth parameter, called bandwidth. Generally, bandwidth represents how accurate we can achieve. So, the smaller h is, the more accurate the result would be. However,

for a dataset with unknown data distribution, any fixed h could lead to bias. Consequently, we introduce variable h [13], shown in formula (3):

$$h_i = \sigma_i \left(\frac{4}{3n} \right)^{\frac{1}{5}} \quad (3)$$

$i=1, 2$, indicating K_1 or K_2 . In other words, 1 means support and 2 confidence. σ_i is the standard deviation of support or confidence. n represents how many rules we discover.

After this step, the real significant rules will be highlighted according to support and confidence. For instance, if one rule has both of high parameters, then it will be significant. On the other hand, the rule with high confidence and low support could be highlighted as well only if the combination of the parameters is high enough. The pseudocode of this part is outlined below:

Kernel Density Estimate

Input: significant mRNA association rules

Output: sorted mRNA rules with combined values

Sorting mRNA rules by support

Sorting mRNA rules by confidence

For every rule

Calculating kernel density **using** (1),(2)

End

Sorting all the rules **based on** the combined values

3.3 Hypothesis Test

The reasons of using hypothesis test for evaluating the significance of mRNA rules are [14]: 1). It is difficult to determine the fixed support and confidence for a dataset that we do not understand the data distribution or other necessary information; and 2). Pure parameters are unacceptable for bioinformatics because they do not represent any biological meanings. So we introduce hypothesis test. Due to a hypothesis that all the huge data is supposed to be close to normal distribution, we choose the hypothesis test with unknown standard deviation.

First, we need to know the mean of the kernel estimates for all the rules:

$$\bar{f} = \sum_{i=1}^n \hat{f}_i(x) \quad (4)$$

The above formula is the mean of all the rules according to kernel density estimate.

Then we choose two hypotheses: null and alternative. Null hypothesis is $\hat{f}_i(x) = \bar{f}$,

alternative hypothesis is $\hat{f}_i(x) \neq \bar{f}$. The former means one rule is close to the mean so that it is not significant. The later indicates one rule is significant enough.

Because of unknowing standard deviation, we choose T test as statistic, the formula of which is shown below:

$$T = \frac{\hat{f}_i - \bar{f}}{S / \sqrt{n}} \quad (5)$$

where n is the number of mRNA rules. S is sample standard deviation, as shown in (5).

$$S = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (\hat{f}_i - \bar{f})^2} \quad (6)$$

Suppose $\alpha=0.05$, one rule i is significant if it satisfies:

$$T_i > T_{\frac{\alpha}{2}} \quad (7)$$

$T_{\frac{\alpha}{2}}$ can be found in any T test list. (7) represents rule i is significant if T test of it is big than α level.

This part of work guarantees the statistical significance of our results. By hypothesis test, we transfer support and confidence into statistical values through kernel density estimate. Therefore, we can not only find the real significant mRNA rules without losing the original information, but also remain our results important biologically.

4 Experiments

To prove that our method can discover the novel and significant mRNA association rules based on miRNA, we have conducted a variety of experiments on RMDR [15] that is a plant miRNA database. We choose the relation between mRNA and miRNA of cotton. The original list has two columns: the first one is miRNA and the second one mRNA. miRNA and mRNA in one row indicates their relevance. So, we transfer the data into a transaction dataset by the following strategy: for every miRNA, we search all the mRNAs related to it and leave them behind it. Therefore, miRNA becomes an ID while mRNAs are the items in one transaction. Our research will be carried out based on this dataset. All the experiments and analysis have worked on

CPU 2G, Memory 4G, Windows 7. We use C++ and Perl to code the algorithm of association rule mining. The process of kernel density estimate and hypothesis test is programmed in Matlab.

Cotton data includes 22 mRNA families: Glyma01, Glyma02, Glyma03, Glyma04, Glyma05, Glyma06, Glyma07, Glyma08, Glyma09, Glyma10, Glyma11, Glyma12, Glyma13, Glyma14, Glyma15, Glyma16, Glyma17, Glyma18, Glyma19, Glyma20, Glyma0041, Glyma0021. For easier description, we brief the family names by ignoring “Glyma”. So in the next context, 16 means Glyma16, 0041 implies Glyma0041. The rest can be done in the same manner.

Our experiments will analyze the directed regulation mechanism of them. Besides, we also try to analyze the regulation network made by them. Our analysis includes the following aspects: time, in- and out- degree, as well as specific mRNA rules.

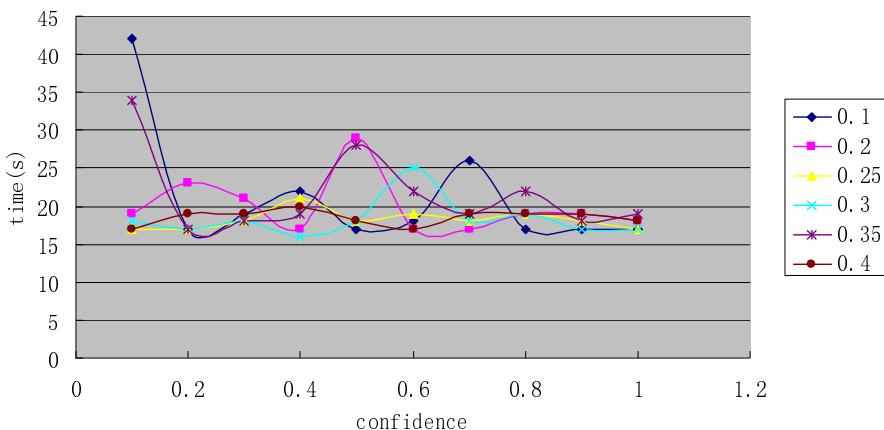


Fig. 2. Running time (s) when different confidence and support are taken

Fig. 2 analyzes how many seconds our method uses when confidence and support are changed. Legend indicates we employ 6 supports: 0.1, 0.2, 0.25, 0.3, 0.35, 0.4. x-axis means the confidence values we utilize: 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1. From this figure, one can see the time cost does not change a lot no matter which parameters are used. Considering some unexpected noises during calculation such as data flow from internet, we can speculate that almost the experiments need similar time cost. Therefore, we are supposed to find the similar rules. This analysis demonstrates that the dataset is very symmetric. The number of families to which any mRNA family are similar as other families do, which can be demonstrated in the following illustration about in- and out- degree.

The research about in- and out- degree just involves two-item mRNA rules for any multi-item rules are extended from them. We first construct a network utilizing two-item mRNA rules. If two mRNA are in one frequent itemset, then they are linked by a line. Then confidence between them can be used for indicating the regulation direction between them. For example, mRNA family A are related to 10 other

families, in which 4 families regulate A while 6 are regulated by A. So, in-degree of A is 4 and out-degree of A 6. Fig. 3 and Fig. 4 show the in- and out- degree for every mRNA family respectively. One can see most of the mRNA families possess the similar in- and out- degree. With taking Fig. 2 into consideration, we can conclude that the relation between mRNA families is very even. They regulate each other with the similar probability. From biological point of view, the mRNA network almost has no bias for any specific mRNA family.

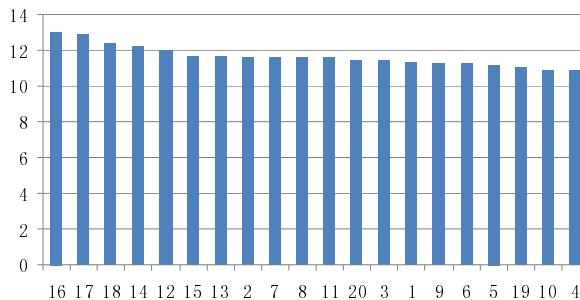


Fig. 3. The descending sequence of out-degree

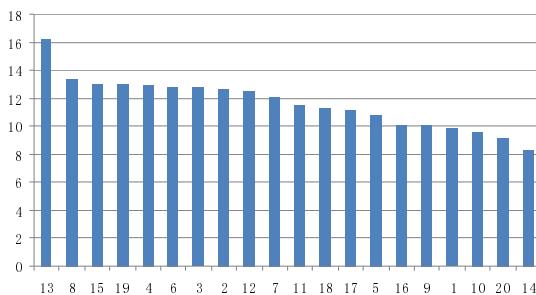


Fig. 4. The descending sequence of in-degree

Next, we list specific mRNA association rules in Table 1 with support and confidence. We still use the traditional parameters because they can naturally express how close two mRNA families are and the regulation direction between them. In Table 2, we list some multi-item mRNA association rules. In both of these tables, the mRNA families in column “antecedent” mean they are in the left side of the association rules. Those in column “consequent” represent they are located in the right side of the rules. The left regulate the right.

We select the specific rules to demonstrate the effectiveness of our method in Table 1. we can classify rule 1, 2, 3, 4, 8, 9 into three groups. For example, from 1 and 2, we can find both of them are important because they not only are related closely, but also could regulate each other from different direction. $14 \rightarrow 13$ and

$05 \rightarrow 12$ is significant as well because they are only regulated strongly from one direction. Compared with them, frequent itemset (07, 10) is not very interesting as on one hand, they are related loosely, on the other hand, they do not have a strong regulation relation.

Table 1. Two-item mRNA association rules

ID	Antecedent	Consequent	support	confidence
1	01	04	63	0.73
2	04	01	63	0.53
3	01	05	48	0.55
4	05	01	48	0.50
5	01	07	50	0.58
6	14	13	61	0.89
7	05	12	73	0.76
8	07	09	66	0.64
9	09	07	66	0.74
10	07	10	43	0.41
11	10	07	43	0.5

Table 2. Multi-item mRNA association rules

ID	Antecedent	Consequent	support	confidence
1	01	04, 05	38	0.44
2	04	01, 05	38	0.32
3	05	01, 04	38	0.4
4	01	04, 13	52	0.61
5	01, 04	13	52	0.82
6	01, 13	04	52	0.82
7	13	01, 04	52	0.44
8	01, 03, 04, 06, 18	02	28	0.96
9	01, 02, 03, 06, 18	04	28	0.9
10	02, 03	01, 11, 20	23	0.28

The multi-item mRNA association rules in Table. 2 can also be classified into different groups. For example, rule 1, 2 and 3 indicates the frequent itemset (01, 04, 05) are important because they have strong directed regulation no matter which mRNAs are regulated. Some directed rules made by frequent itemset (01, 04, 13) are significant such as rule 5 and 6. The specific ones are rule 8 and 9. A frequent itemset (01, 02, 03, 04, 06, 18) only has such low support. However, two directed regulation rules are very sound. Relatively, rule 10 is not significant because this itemset has no either support or confidence.

Summarizing, our method can analyze mRNA regulation mechanism and the relation between mRNA and miRNA by combining association rule mining, kernel density estimate and hypothesis test. Our result can discover the novel and significant information about mRNA network and the directed regulation.

5 Conclusion

The improved association rule mining has been concerned more and more in bioinformatics. However, it has not been used for the discovery of mRNA regulation mechanism with direction so far. In this paper, we introduce association rule mining into the field of mRNA regulation. Besides, we employ kernel density estimate to overcome the drawbacks of the traditional Apriori algorithm. Last, we use hypothesis test to guarantee the biological significance of our research. Our results demonstrate that our method can find the directed regulation relation between mRNAs effectively and efficiently. Further more, our results can help to analyze the difference between mRNA clusters and the potential relation between miRNA and mRNA.

References

1. Agrawal, R., Imieliński, T., Swami, A.: Mining Association Rules between Sets of Items in Large Databases. In: SIGMOD, pp. 207–216 (1993)
2. Zaki, M.J.: SPADE: An Efficient Algorithm for Mining Frequent Sequences. Machine Learning Journal 42, 31–60 (2001)
3. Salleb-Aouissi, A., Vrain, C., Nortet, C.: QuantMiner: A Genetic Algorithm for Mining Quantitative Association Rules. In: IJCAI, pp. 1035–1040 (2007)
4. Agrawal, R., Srikant, R.: Fast Algorithms for Mining Association Rules in Large Databases. In: VLDB, pp. 487–499 (1994)
5. Horner, D.S., Pavesi, G., Castrignano, T., D’Onorio De Meo, P., Liuni, S., Sammeth, M., Picardi, E., Pesole, G.: Bioinformatics Approaches for Genomics and Post Genomics Applications of Next-Generation Sequencing. Briefings in Bioinformatics 11(2), 181–197 (2009)
6. Dai, X., Zhuang, Z., Zhao, P.X.: Computational Analysis of miRNA Targets in Plants: Current Status and Challenges. Briefings in Bioinformatics, 1–7 (2010)
7. Watanabe, Y., Tomita, M., Kanai, A.: Computational Methods for microRNA Target Prediction. Methods in Enzymology 427, 65–86 (2007)

8. Xu, J., Li, C., Li, Y., Lv, J., Ma, Y., Shao, T., Xu, L., Wang, Y., Du, L., Zhang, Y., Jiang, W., Li, C., Xiao, Y., Li, X.: MiRNA–miRNA Synergistic Network: Construction via Co-regulating Functional Modules and Disease miRNA Topological Features. *Nucleic Acids Research*, pp. 1–12 (2010)
9. Joung, J., Fei, Z.: Identification of microRNA Regulatory Modules in Arabidopsis via a Probabilistic Graphical Model. *Bioinformatics* 25(3), 387–393 (2009)
10. Li, W., Zhang, S., Liu, C., Zhou, X.J.: Identifying Multi-layer Gene Regulatory Modules from Multi-Dimensional Genomic Data. *Bioinformatics* 28(29), 2458–2466 (2012)
11. Bandyopadhyay, S., Mitra, R., Maulik, U., Zhang, M.Q.: Development of the Human Cancer MicroRNA Network. *Silence* 1, 6 (2010)
12. Tran, D.H., Satou, K., Ho, T.B.: Finding MicroRNA Regulatory Modules in Human Genome Using Rule Induction. *BMC Bioinformatics* 9, S5 (2008)
13. Bowman, A., Azzalini, A.: Applied Smoothing Techniques for Data Analysis. Oxford University Press, London (1997)
14. Li, X., Shen, Y., Ichikawa, H., Antes, T., Goldberg, G.S.: Regulation of miRNA expression by Src and contact normalization: effects on nonanchored cell growth and migration. *Oncogene*, 1–12 (2009)
15. Zhang, Z., Yu, J., Li, D., Zhang, Z., Liu, F., Zhou, X., Wang, T., Ling, Y., Su, Z.: PMRD: Plant MicroRNA Database. *Nucleic Acids Research* 38, D806–D813 (2010)

Interval Algorithm for Set-Membership Identification of MIMO LTI System

Xiaojun Wang^{1,*}, Chen Yang², Baochang Zhang³, and Lei Wang¹

¹ Institute of Solid Mechanics, Beihang University, Beijing 100191, China
XJWang@buaa.edu.cn

² Qian Xuesen Laboratory of Space Technology, China Academy of Space Technology,
Beijing 100094, China

³ School of Automation Science and Electrical Engineering, Beihang University,
Beijing 100191, China

Abstract. Based on the assumption of Unknown-But-Bounded (UBB) noise, an interval algorithm is presented for set-membership parameter identification of a multiple-input multiple-output (MIMO) linear time-invariant (LTI) system. By virtue of interval mathematics, the objective of this study is to seek the minimal interval estimation (or hyper-rectangle) of parameters to be identified, which is compatible with the measured data and the bounded noise. The present algorithm can obtain not only the center estimations of parameters, but also the bounds of errors on them. Numerical example is used to illustrate its small computation efforts and higher accuracy by comparison with Fogel's ellipsoidal algorithm and the least squares algorithm.

Keywords: set-membership identification (SMI), linear time-invariant (LTI) system, interval algorithm, Multi-Input and Multi-Output (MIMO), Unknown-But-Bounded (UBB).

1 Introduction

System modeling and parameter estimation have wide applications in many areas. Set-membership identification (SMI) has become an intensive research area for parameter estimation. One of the important topics in SMI is the development of identification algorithms based on the mathematical model which is represented by the relationship between input and output data. This study can be used to improve the system control quality, design the advanced control system and achieve the optimal control. Moreover, the SMI-based parameter identification can be generalized to the structural parameter identification and structural damage detection. Unfortunately, there always exist noise and uncertainty in all observing process. The traditional method to deal with the uncertainty problem is based on random description of noise, and it can be used to obtain the point estimation of system parameters. Although the parameter estimation theory and method with random noise assumption is quite mature,

* Corresponding author.

the probability statistics information of noise is always absent. As well known, it is easier to obtain the noise bound than its statistical distribution, especially in some specified circumstances. Based on Unknown-But-Bounded (UBB) noise assumption, SMI demonstrates its advantages with less uncertain information. It can be got the compatible members set between observational data and noise bounded in the parameter space. SMI is presented by Fogel [1, 2], who has put forward an ellipsoid algorithm where energy constraints and noise bounded is instantaneous. Different SMI algorithms have been developed rapidly since then. For instance, in virtue of linear programming, Milanese and Belforte [3] proposed a batch identification algorithm based on box bounds. Cerone [4] has studied in ARMAX model identification with input and output both polluted by UBB noise. The study presents two algorithms, which are applied in equation error model and bounded variable error model, respectively. Sun [5] summarized different SMIs. Yue [6] presents experimentation design in SISO linear time-invariant system.

Typical methods for solving linear programming (LP) problems are the standard simplex method, the ellipsoid algorithm, Karmarkar's algorithm, and some variants of them. Since both the standard and bounded-variable-simplex methods are computationally efficient and easy to understand they are popular methods in engineering practice. Among many SMI algorithms, the most popular ones are a set of algorithms termed OBE (optimal bounding ellipsoids) algorithms. The OBE algorithms provide optimal (in some sense) outer bounding ellipsoids of the membership set. They have similarities, in terms of formulas for updating the estimates, to the weighted recursive least squares (WRLS) algorithm as well as analytical tractability and computational simplicity. Clearly, the proposed algorithms are different than the ellipsoid-outer bounding ones since they are least squares (LS) type, but the weights are chosen to minimize the “volume” of the outer bounding ellipsoid. Moreover, it is well known that the recursive implementation of outer bounding algorithms may introduce some conservatism. The implication is that there is no guarantee that a point inside the outer-bounding set is also in the membership set. However, in contrast to the recursive least squares (RLS) algorithm which obtains an optimal (in the sense of least squares error (LSE)) point estimate at each recursion, the OBE algorithms seek an optimal ellipsoid that bounds the membership set. In recent years, researchers study the nonlinear identification and uncertainty theory frequently. A literature [7] presents a Closed Loop Input Error (CLIE) approach for on-line parametric estimation of a continuous-time model of a DC servomechanism functioning in closed loop. Another paper [8] considers a dynamical level set method for the identification problem of the nonlinear parabolic distributed parameter system, which is based on the solvability and stability of the direct PDE (partial differential equation) in Sobolev space. One study [9] investigates the potential use of the Differential Evolution Genetic Algorithm (DEGA) in the back-analysis of tunnel response in order to obtain improved estimates of model parameters by matching model prediction with monitored response.

Recently, the interval analysis method [10] has shown its unique advantages in dealing with bounded uncertainty problems. Some researchers have studied the convergence properties of the membership set for system identification in the presence

of unknown but bounded noise. In this paper, based on interval mathematical thought and SMI of SISO LTI system [11], a new SMI algorithm of MIMO LTI system is presented. Through numerical example, the comparisons between the proposed interval algorithm, Fogel's ellipsoid algorithm and least square method are performed extensively, which exhibit its advantages.

The paper is organized as follows: in Section 2, the statement of problem is formulated. Section 3 is devoted to the optimized parameter feasible set. One numerical example is given in Section 4. Section 5 concludes the paper.

2 Statement of Problem

Consider the identification problem with MIMO LTI system as follow

$$\mathbf{y}_k = \sum_{i=1}^n \mathbf{A}_i \mathbf{y}_{k-i} + \sum_{j=0}^m \mathbf{B}_j \mathbf{u}_{k-j} + \mathbf{e}_k = \boldsymbol{\theta}^T \mathbf{X}_k + \mathbf{e}_k, \quad k = 1, 2, \dots \quad (1)$$

where \mathbf{y}_k is a p -dimensional output vector, \mathbf{u}_k is a q -dimensional input vector, $\boldsymbol{\theta}^T = (\mathbf{A}_1 \cdots \mathbf{A}_n \mathbf{B}_0 \cdots \mathbf{B}_m)$ is a $p \times [np + (m+1)q]$ -dimensional parameter matrix to be identified, $\mathbf{X}_k = (\mathbf{y}_{k-1}^T \cdots \mathbf{y}_{k-n}^T \mathbf{u}_k^T \cdots \mathbf{u}_{k-m}^T)^T$ is a $np + (m+1)q$ -dimensional recursive vector, \mathbf{e}_k is a p -dimensional Unknown-But-Bounded (UBB) noise sequence. By interval mathematics notation, \mathbf{e}_k can be expressed as

$$\mathbf{e}_k \in \mathbf{e}_k^I = [-\omega_k, \omega_k], \quad k = 1, 2, \dots \quad (2)$$

where ω_k is a p -dimensional known vector. Based on the SMI concept, the problem can be described as follows.

The sequence, $\{\mathbf{y}_k, \mathbf{X}_k, \omega_k; k = 1, 2, \dots\}$, can been found a set $\Gamma \subset \mathbf{R}^{p \times [np + (m+1)q]}$ which is compatible with Eq.(1) and Eq.(2). The set Γ is the unknown parameter matrix of the feasible membership set $\boldsymbol{\theta}$, namely

$$\Gamma = \{\boldsymbol{\theta} : \mathbf{y}_k - \boldsymbol{\theta}^T \mathbf{X}_k = \mathbf{e}_k, \mathbf{e}_k \in \mathbf{e}_k^I, \quad k = 1, 2, \dots\} \quad (3)$$

Any value in the set Γ is a feasible solution. It can be always used to certain geometry center. $\boldsymbol{\theta}^*$ can be used as $\boldsymbol{\theta}_t$ estimation, where $\boldsymbol{\theta}_t$ stands for the true value and $\boldsymbol{\theta}^*$ for the calculative value. Generally speaking, Γ is an abnormal convex set. To analyze and control the system conveniently, it can be found a convex set, which is contained Γ as possible as closely. Based on the interval mathematics, we can seek a hyper-rectangle (or interval matrix) to approximate membership set Γ . Then we can find the lower and upper bound of the membership set about the identified parameter vector.

$$\Theta^O = \{\boldsymbol{\theta} : \boldsymbol{\theta} \in \Theta^I = [\underline{\boldsymbol{\theta}}, \bar{\boldsymbol{\theta}}] = (\boldsymbol{\theta}'_{ij})\} \quad (4)$$

Thus, the center of the set can regard as the estimation of real parameter

$$\boldsymbol{\theta}^* = (\underline{\boldsymbol{\theta}} + \bar{\boldsymbol{\theta}}) / 2 \quad (5)$$

From arguing above, the interval $(\underline{\boldsymbol{\theta}}, \bar{\boldsymbol{\theta}})$ is the smallest membership set of $\boldsymbol{\theta}_t$, which is consistent with the measurements, and $\boldsymbol{\theta}^* = (\underline{\boldsymbol{\theta}} + \bar{\boldsymbol{\theta}}) / 2$ is the uniformly minimum variance unbiased (UMVU) estimator of $\boldsymbol{\theta}_t$.

3 Parameter Feasible Set by Interval Algorithm

Based on the limit memory of the least square method, keep the data length M in each identified process as follow

$$\mathbf{Y}_k = \Phi_k \boldsymbol{\theta} + \mathbf{E}_k \quad (6)$$

$$\text{where } \mathbf{Y}_k = \begin{pmatrix} \mathbf{y}_{k-M+1}^T \\ \mathbf{y}_{k-M+2}^T \\ \vdots \\ \mathbf{y}_k^T \end{pmatrix} \text{ and } \mathbf{E}_k = \begin{pmatrix} \mathbf{e}_{k-M+1}^T \\ \mathbf{e}_{k-M+2}^T \\ \vdots \\ \mathbf{e}_k^T \end{pmatrix}$$

$$\Phi_k = \begin{pmatrix} \mathbf{X}_{k-M+1}^T \\ \mathbf{X}_{k-M+2}^T \\ \vdots \\ \mathbf{X}_k^T \end{pmatrix} = \begin{pmatrix} \mathbf{y}_{k-M}^T & \cdots & \mathbf{y}_{k-M+1-n}^T & \mathbf{u}_{k-M+1}^T & \cdots & \mathbf{u}_{k-M+1-m}^T \\ \mathbf{y}_{k-M+1}^T & \cdots & \mathbf{y}_{k-M+2-n}^T & \mathbf{u}_{k-M+2}^T & \cdots & \mathbf{u}_{k-M+2-m}^T \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{y}_{k-1}^T & \cdots & \mathbf{y}_{k-n}^T & \mathbf{u}_k^T & \cdots & \mathbf{u}_{k-m}^T \end{pmatrix} \quad \text{is the}$$

recursive matrix.

Due to above, it can be obtained a linear matrix formula by transposition of Eq.(6), which $\boldsymbol{\theta}$ is an unknown quantity

$$\Phi_k \boldsymbol{\theta} = \mathbf{Y}_k - \mathbf{E}_k \quad (7)$$

When $M < np + (m+1)q$, parameter $\boldsymbol{\theta}$ is uncertain, this case is therefore unconsidered; if $M = np + (m+1)q$, only when $\mathbf{E}_k = \mathbf{0}$, $\boldsymbol{\theta}$ is just certain. This case is unconsidered neither. Just take into account the case which is $M > np + (m+1)q$.

Using the Moore-Penrose pseudo inverse method, it can be solved unknown $\boldsymbol{\theta}$ is

$$\boldsymbol{\theta}_k = (\Phi_k)^+ (\mathbf{Y}_k - \mathbf{E}_k) = (\Phi_k^T \Phi_k)^{-1} \Phi_k^T (\mathbf{Y}_k - \mathbf{E}_k) \quad (8)$$

Then, we take the nature interval expansion to Eq.(8)

$$\underline{\boldsymbol{\theta}}_k^I = (\boldsymbol{\Phi}_k^T \boldsymbol{\Phi}_k)^{-1} \boldsymbol{\Phi}_k^T (\mathbf{Y}_k - \mathbf{E}_k') = [\underline{\boldsymbol{\theta}}_k, \bar{\boldsymbol{\theta}}_k] = ((\boldsymbol{\theta}_k)_{ij}^I) \quad (9)$$

It can be got the lower and upper bound of the identified parameter by using interval arithmetic. Thus, the component format is

$$(\boldsymbol{\theta}_k)_{ij}^I = [\underline{(\boldsymbol{\theta}_k)}_{ij}, \bar{(\boldsymbol{\theta}_k)}_{ij}], \quad i = 1, 2, \dots, np + (m+1)q, \quad j = 1, 2, \dots, p \quad (10)$$

where

$$\underline{(\boldsymbol{\theta}_k)}_{ij} = ((\boldsymbol{\Phi}_k^T \boldsymbol{\Phi}_k)^{-1} \boldsymbol{\Phi}_k^T \mathbf{Y}_k)_{ij} - \sum_{l=1}^M [(\boldsymbol{\Phi}_k^T \boldsymbol{\Phi}_k)^{-1} \boldsymbol{\Phi}_k^T]_{il} (\boldsymbol{\omega}_{k-M+l})_j \quad (11)$$

and

$$\bar{(\boldsymbol{\theta}_k)}_{ij} = ((\boldsymbol{\Phi}_k^T \boldsymbol{\Phi}_k)^{-1} \boldsymbol{\Phi}_k^T \mathbf{Y}_k)_{ij} + \sum_{l=1}^M [(\boldsymbol{\Phi}_k^T \boldsymbol{\Phi}_k)^{-1} \boldsymbol{\Phi}_k^T]_{il} (\boldsymbol{\omega}_{k-M+l})_j \quad (12)$$

Hence, the parameter feasible set is

$$\begin{aligned} \boldsymbol{\Theta}_k = \{ \boldsymbol{\theta} : \boldsymbol{\theta} \in \boldsymbol{\Theta}_k^I = ((\boldsymbol{\theta}_k)_{ij}^I), (\boldsymbol{\theta}_k)_{ij}^I = [\underline{(\boldsymbol{\theta}_k)}_{ij}, \bar{(\boldsymbol{\theta}_k)}_{ij}], \\ i = 1, 2, \dots, np + (m+1)q, j = 1, 2, \dots, p \} \end{aligned} \quad (13)$$

and then we can get the optimized parameter feasible set $\boldsymbol{\Theta}_k^O$ is

$$\Gamma \subset \boldsymbol{\Theta}_k^O = \bigcap_{i=1}^k \boldsymbol{\Theta}_k \quad (14)$$

In general, the set Γ has a very close but accurate region. To two-dimensional problem, Figure 1 shows its geometrical significance.

4 Numerical Examples

To prove algorithm available, we compare the interval algorithm for SMI with ellipsoid algorithm [2,12] and least square method [2,13]. Consider two-order model:

$$\begin{bmatrix} y_1(k) \\ y_2(k) \end{bmatrix} = \begin{bmatrix} 0.1 & -0.6 \\ -0.2 & 0.7 \end{bmatrix} \begin{bmatrix} u_1(k-1) \\ u_2(k-1) \end{bmatrix} + \begin{bmatrix} e_1(k) \\ e_2(k) \end{bmatrix}$$

Assume $u_1(k)$, which period is 20, $u_1(k) = 1.6, k = 1, \dots, 10$; $u_1(k) = -1.8, k = 11, \dots, 20$. $u_2(k)$, which period is 10, $u_2(k) = 0.6, k = 1, \dots, 5$;

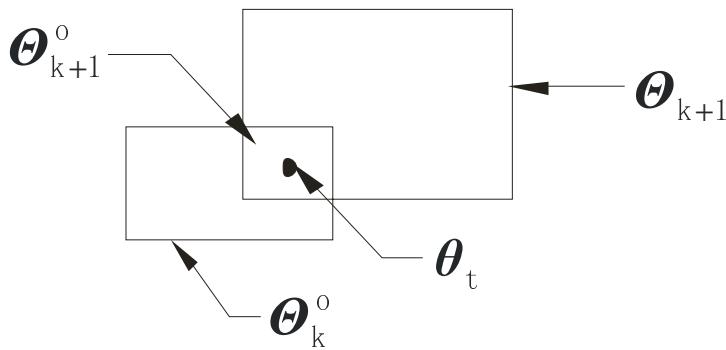


Fig. 1. Schematic Diagram of Interval Algorithm for Set-Membership Identification

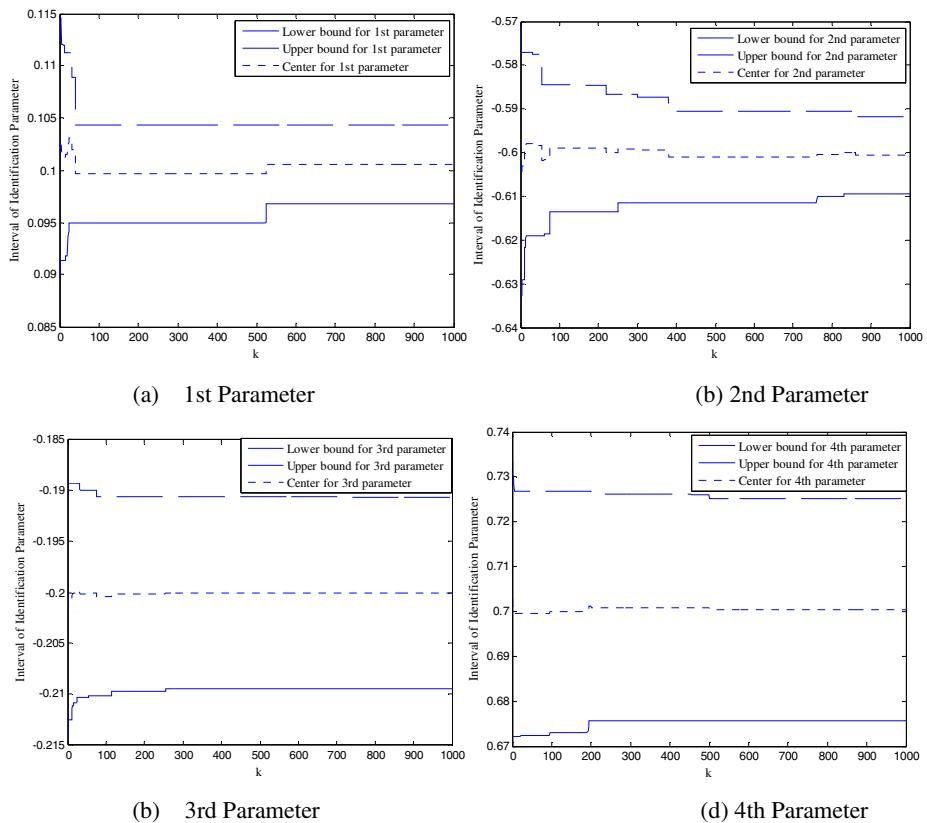


Fig. 2. Intervals and Centers of Identified Parameters

$u_2(k) = -0.7, k = 6, \dots, 10$. $e_1(k)$ and $e_2(k)$ are uncorrelated and uniform noises, which are 1% maximum respond.

Figure 2 indicates the interval bound and central values of identified parameter with observed data change course by interval algorithm. In Figure 3, it is present central values with observed data change course by interval algorithms, ellipsoid algorithm and least square method. Figure 4 and Figure 5 are given its identification error and convergence case about identified members set "volume" respectively. Results in Figure 2 reveal the high precision in interval algorithm to the all parameters. They have converged to real solution when $k = 1000$. From the figure, it is obvious to detect that the converged speed with interval algorithm is much faster than other algorithms in all parameters. Furthermore, its center estimation for the model is very close to the same real solution. For some parameters with less data, ellipsoid algorithm has not been converged, and the identified precision by least square method is also lower than interval algorithms.

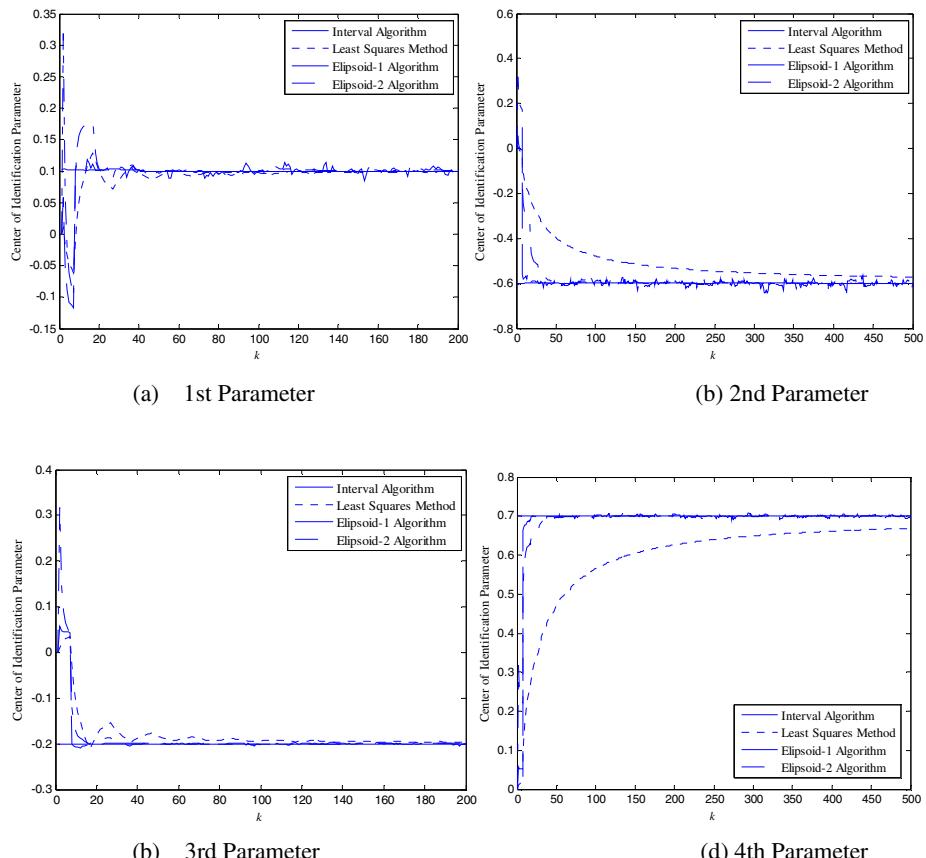


Fig. 3. Centers of Identified Parameters with Four Methods

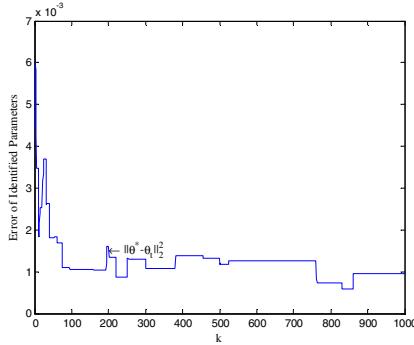
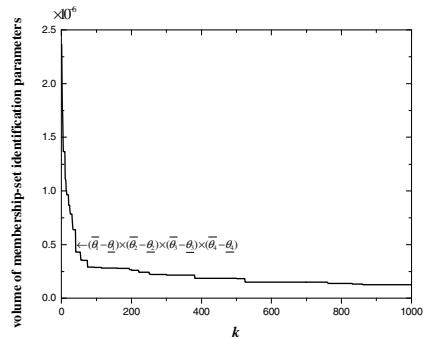
**Fig. 4.** Error of Identified Parameters**Fig. 5.** Volume of Set-Membership

Table 1 indicates identified parameters compared results with different identification algorithm. It can be summarized the accuracy of identified results with less observed data with interval algorithm are obviously higher than ellipsoid algorithm and least square method. In Table 2 we summarize the interval bounds and central values of identified parameters, which is in 1%, 5%, 10% degree of noise with interval algorithms. When noise is 10% and $k = 1000$, the maximum identified error in model is just 0.95%. It is very clear that in large noise case, the central values are all very close to real solution.

Table 1. Centers of Identified Parameters

k	Interval Algorithm		Ellipsoid-1 Algorithm[5]		Ellipsoid-2 Algorithm[5]		Least Squares Method	
	Intervals	Centers	Centers		Centers		Centers	
			Centers	Centers	Centers	Centers	Centers	Centers
200	(0.0950, 0.1044)	-0.6135, -0.5846	0.0997	-0.5990	0.1004	-0.5910	0.1029	-0.5958
	(-0.2097, -0.1906)	(0.6756, 0.7269)	-0.2002	0.7012	-0.1997	0.7051	-0.1993	0.7021
400	(0.0950, 0.1044)	(-0.6115, -0.5906)	0.0997	-0.6010	0.1017	-0.5974	0.1009	-0.6045
	(-0.2094, -0.1906)	(0.6756, 0.7261)	-0.2000	0.7009	-0.1998	0.7006	-0.1998	0.7005
600	(0.0968, 0.1044)	(-0.6115, -0.5906)	0.1006	-0.6010	0.1023	-0.6169	0.1005	-0.6036
	(-0.2094, -0.1906)	(0.6756, 0.7253)	-0.2000	0.7004	-0.1996	0.6976	-0.1995	0.6997
800	(0.0968, 0.1044)	(-0.6101, -0.5906)	0.1006	-0.6003	0.0969	-0.6024	0.0991	-0.6044
	(-0.2094, -0.1906)	(0.6756, 0.7253)	-0.2000	0.7004	-0.2007	0.6967	-0.2003	0.6986
1000	(0.0968, 0.1044)	(-0.6095, -0.5918)	0.1006	-0.6007	0.1005	-0.5889	0.1004	-0.5949
	(-0.2094, -0.1906)	(0.6756, 0.7253)	-0.2000	0.7004	-0.2007	0.7035	-0.2001	0.7016

Table 2. Centers of Identified Parameters in Different Noise with Interval Algorithm

Noise Degree	<i>k</i>	Interval Algorithm		
		Intervals	Centers	
1%	200	(0.0950,0.1044)	(-0.6135,-0.5846)	0.0997 -0.5990
		(-0.2097,-0.1906)	(0.6756,0.7269)	-0.2002 0.7012
	400	(0.0950,0.1044)	(-0.6115,-0.5906)	0.0997 -0.6010
		(-0.2094,-0.1906)	(0.6756,0.7261)	-0.2000 0.7009
	600	(0.0968,0.1044)	(-0.6115,-0.5906)	0.1006 -0.6010
		(-0.2094,-0.1906)	(0.6756,0.7253)	-0.2000 0.7004
	800	(0.0968,0.1044)	(-0.6101,-0.5906)	0.1006 -0.6003
		(-0.2094,-0.1906)	(0.6756,0.7253)	-0.2000 0.7004
	1000	(0.0968,0.1044)	(-0.6095,-0.5918)	0.1006 -0.6007
		(-0.2094,-0.1906)	(0.6756,0.7253)	-0.2000 0.7004
<i>k</i>		Intervals	Centers	
5%	200	(0.0750,0.1218)	(-0.6674,-0.5229)	0.0984 -0.5951
		(-0.2486,-0.1530)	(0.5781,0.8344)	-0.2008 0.7062
	400	(0.0750,0.1218)	(-0.6577,-0.5528)	0.0984 -0.6052
		(-0.2471,-0.1530)	(0.5781,0.8305)	-0.2001 0.7043
	600	(0.0840,0.1218)	(-0.6577,-0.5528)	0.1029 -0.6052
		(-0.2471,-0.1530)	(0.5781,0.8263)	-0.2001 0.7022
	800	(0.0840,0.1218)	(-0.6505,-0.5528)	0.1029 -0.6016
		(-0.2471,-0.1531)	(0.5781,0.8263)	-0.2001 0.7022
	1000	(0.0840,0.1218)	(-0.6475,-0.5591)	0.1029 -0.6033
		(-0.2471,-0.1531)	(0.5781,0.8263)	-0.2001 0.7022
<i>k</i>		Intervals	Centers	
10%	200	(0.0500,0.1437)	(-0.7348,-0.4458)	0.0968 -0.5903
		(-0.2973,-0.1059)	(0.4562,0.9688)	-0.2016 0.7125
	400	(0.0500,0.1437)	(-0.7154,-0.5055)	0.0968 -0.6104
		(-0.2943,-0.1059)	(0.4562,0.9611)	-0.2001 0.7086
	600	(0.0679,0.1437)	(-0.7154,-0.5055)	0.1058 -0.6104
		(-0.2943,-0.1059)	(0.4562,0.9526)	-0.2001 0.7044
	800	(0.0679,0.1437)	(-0.7010,-0.5055)	0.1058 -0.6033
		(-0.2943,-0.1063)	(0.4562,0.9526)	-0.2003 0.7044
	1000	(0.0679,0.1437)	(-0.6950,-0.5181)	0.1058 -0.6065
		(-0.2943,-0.1063)	(0.4562,0.9526)	-0.2003 0.7044

5 Conclusions

To conclude the paper, the main results are summarized. Based on interval mathematics, this paper proposes a kind of interval algorithm for multi-input multi-output (MIMO) linear time-invariant (LTI) system SMI. It has less request for noise prior information, only need to know the upper and lower bounds of uncertainty. Identified results not only show the parameter estimation, but also present the parameter uncertainty bound. Compared to Fogel ellipsoid algorithm and the least squares algorithm, interval algorithm has its advantage in less calculation and high accuracy. This method present in this paper can be also extended to time-varying system identification and used as a further research.

Acknowledgments. We thank the Specialized Research Fund for the Doctoral Program of Higher Education (No.20091102120023), Aeronautical Science Foundation of China (No.2012ZA51010), National Nature Science Foundation of the P. R. China (No.11002013), and Defense Industrial Technology Development Program (No.A2120110001, No.B2120110011) for support. This work was supported in part by the Natural Science Foundation of China, under Contracts 60903065, 61039003, 61070148 and 61272052, in part by the Ph.D. Programs Foundation of Ministry of Education of China, under Grant 20091102120001, in part by the Fundamental Research Funds for the Central Universities.

References

1. Fogel, E.: System identification via membership set constraints with energy constrained noise. *IEEE Transactions on Automatic Control* AC 24(5), 752–758 (1979)
2. Fogel, E., Huang, Y.F.: On the value of information in system identification-Bounded noise case. *Automatica* 18(2), 229–238 (1982)
3. Milanese, M., Belforte, G.: Estimation theory and uncertainty intervals evaluation in present of unknown but bounded errors: Linear families of models and estimators. *IEEE Transactions on Automatic Control* AC 27(2), 408–414 (1982)
4. Cerone, V.: Feasible parameter set for linear models with bounded errors in all variables. *Automatica* 29(6), 1551–1555 (1993)
5. Sun, X.F.: Set membership identification-Algorithm, Convergence, Robustness. (Ph D Thesis). Institute of Automation, Chinese Academy of Sciences Beijing (1994) (in Chinese)
6. Yue, Z.J.: Near-optimal input design for set-membership identification. *Journal of PLA University of Science and Technology* 2(2), 96–98 (2001) (in Chinese)
7. Garrido, R., Miranda, R.: DC servomechanism parameter identification: A closed loop input error approach. *ISA Transactions* 51(1), 42–49 (2012)
8. Li, L., Liu, W.Y., Han, B.: Dynamical level set method for parameter identification of nonlinear parabolic distributed parameter systems. *Commun. Nonlinear Sci. Numer. Simulat.* 17(7), 2752–2765 (2012)
9. Vardakos, S., Gutierrez, M., Xia, C.C.: Parameter identification in numerical modeling of tunneling using the Differential Evolution Genetic Algorithm (DEGA). *Tunnelling and Underground Space Technology* 28, 109–123 (2012)

10. Qiu, Z.P., Wang, X.J.: Comparison of dynamic response of structures with uncertain-but-bounded parameters using non-probabilistic interval analysis method and probabilistic approach. *International Journal of Solids and Structures* 40(20), 5423–5439 (2003)
11. Wang, X.J., Qiu, Z.P.: Interval Algorithm for Membership-Set Identification of Linear Time-invariant System. *Chinese Journal of Theoretical and Applied Mechanics* 37(6), 713–718 (2005)
12. Wang, W.Z., Cai, J.S.: Optimal Algorithms of Set Member Identification for Parameters of MIMO Systems. *Control Theory and Applications* 14(3), 402–406 (1997)
13. Yuan, Z.D., Xu, Q.N.: Set Membership Identification for the Parameters of MIMO System. *Control Theory and Applications* 11(4), 404–412 (1994)

An Approach for Obtaining Bargaining Set in Multi-choice Games

Tianwen Li^{1,2}, Feng Ma^{1,3}, and Weiyi Liu¹

¹ School of Information Science and Engineering, Yunnan University, Kunming, China

² School of Science, Kunming University of Science and Technology, Kunming, China

³ Information school, Yunnan University of Finance and Economics, Kunming, China

bushltw@163.com

Abstract. In the field about the intersection of game theory and decision making, it is a hot concern how to obtain stable payoff vectors. In this paper, we first extend the solution of bargaining set in classical cooperative game to multi-choice games with coalition structure. The extended bargaining set is a generalization of the classical bargaining set. Then we give an approach to find a stable payoff vector belonging to the extended bargaining set, based on genetic algorithm. Finally, we give some experiments to find a stable payoff vector by our proposed algorithm and evaluate the finding algorithm.

Keywords: Multi-choice games, Stable payoff vectors, Bargaining set, Genetic Algorithm.

1 Introduction

Game theory [1] is the study of mathematical models for conflict and cooperation between decision makers, and it has become an effective tool to describe strategic interactions in real world applications. The field about the intersection of game theory and decision making is a research hot. Many solutions in game theory have been applied to the field of decision making. Luis Castillo et al.[18] applied a Nash genetic algorithm to solve the decision making optimization in Liquefied Natural Gas projects. Nanlin Jin et al. [16] used an evolutionary algorithm to generate some efficient and stable strategies for complicated bargaining problems. Onn Shehory and Sarit Kraus [17] adopted the kernel and the local Pareto optimality to give an algorithm for forming coalitions.

Multi-choice games introduced by Hsiao and Raghavan [8, 9] is a generalization of the classical transferable utility cooperative game. Players in multi-choice games can contribute to the worth of a coalition at several activity levels, and those in classical cooperative games can only contribute at two activity levels, which are active or inactive [15]. To achieve a better understanding of the multi-choice games, we give an example. A company with many workers is divided to several departments, where a worker belongs to only one department. The division of company to department forms a coalition structure. The worth of a coalition depend on the performance of

each player, each worker, who works at a certain activity level. This situation gives rise to a multi-choice game with coalition structure.

Some solution concepts about core, Shapley value and Owen value in multi-choice games have appeared. Several extensions of core on multi-choice TU games were respectively proposed by Van den Nouweland et al. [15], Hwang et al. [11], Michel Grabisch et al. [6] and R. Branzei et al. [2, 3]. Hsiao and Raghavan [9] proposed an extension of the Shapley value to multi-choice games, presupposing the existence of weights at each activity level. Then, Van den Nouweland et al. [15], Emilio Calvo et al. [4], Michel Grabisch et al. [7], Hwang et al. [12] and Michael A. Jones et al. [13] further contribute to several extensions of Shapley value in multi-choice games.

In multi-choice games, the core can be sometimes empty, and Shapley value can give a fair payoff. But the payoff is uncertain to be stable. It has been a hot concern how to obtain stable payoff vectors in the field about the intersection of game theory and decision making. In this paper, we give an approach to obtain a stable payoff vector based on genetic algorithm. In classical cooperative game, the bargaining set given by Aumann and Maschler [1] is an important solution concept, which can give stable payoff vectors. In Section 2, we extend the solution of the bargaining set to the multi-choice games with coalition structure, based on the bargaining set in classical cooperative game proposed by Aumann and Maschler [1] and the bargaining set in cooperative fuzzy games given by Jiuqiang Liu and Xiaodong Liu [14]. The extended bargaining set is a generalization of the classical bargaining set. In Section 3, we give an approach to find a stable payoff vector based on genetic algorithm. In Section 4, we give some experiments to find a stable payoff vector by our proposed algorithm and evaluate the finding algorithm.

2 Bargaining Set in Multi-choice Games with Coalition Structure

2.1 Multi-Choice Games with Coalition Structure

Let $N = \{1, 2, \dots, n\}$ be a nonempty finite set of players. In a multi-choice game, each player has $m_i + 1$ activity levels at which he or she can choose to work. We set $M_i = \{0, 1, 2, \dots, m_i\}$ as the action space of a player i , where action 0 means not participating. An element of M^N , $S = (s_1, s_2, \dots, s_n)$, is called a coalition, where each the player i works at level $s_i \in M_i$. The coalition $(0, \dots, 0)$ is denoted an empty coalition. Furthermore, we introduce the notation about $M_i^+ = M_i \setminus \{0\}$ and $M_0^N = M^N \setminus \{0, \dots, 0\}$, and import the carrier of S by $car(S) = \{i \mid s_i > 0, S \in M^N, i \in N\}$. The reward that a coalition can obtain depends on the effort of the cooperating players. In classical cooperative game, coalition structure

[1] $\beta = \{B_1, \dots, B_p\}$ for N is a partition of N as $B_j \cap B_k = \emptyset$, $j \neq k$, $\bigcup_{j=1}^p B_j = N$. The

multi-choice game with coalition structure is a triple (N, v, β) , where N is the set of players, $v: M^N \rightarrow \mathbb{R}$ is the characteristic function, and $\beta = \{B_1, \dots, B_p\}$ is the coalition structure. If there will be no confusion, we denote the multi-choice game with coalition structure (N, v, β) by v .

A payoff for v is a vector $x \in R^{M^N}$, where x_{ij} denotes the increase of the payoff corresponding to a change of the activity level from $j-1$ to j for this player i , for all $i \in N$, $j \in M_i^+$, and $x_{i0} = 0$. The payoff vector x is called level increase rational if x_{ij} is not less than the worth that the player i can obtain when he works alone and changes his activity level from $j-1$ to j , i.e. $x_{ij} \geq v(je^i) - v((j-1)e^i)$. Van den Nouweland et al. [18] gave the concepts of imputation set. Let v be a multi-choice game with coalition structure. Then, the set $I(v, \beta)$ of feasible payoff vectors for v is denoted as Eq. 1, satisfying efficient and level increase rational.

$$I(v, \beta) = \{x \in R^{M^N} \mid \sum_{i \in B \in \beta} \sum_{j=1}^{m_i} x_{ij} = v(B), \text{and } x_{ij} \geq v(je^i) - v((j-1)e^i), \text{for all } i \in N, j \in M_i^+\} \quad (1)$$

2.2 Bargaining Set

In the classical bargaining set $M_1^{(i)}$ [1], (N, w, β) is a classical cooperative game with coalition structure, and x is a payoff vector for (N, w, β) . An objection of one player k against another l at x is a pair (y, R) where $k, l \in B \in \beta$, R is a coalition containing k and not containing l , and y is a vector in \mathbb{R}^R satisfying $\sum_{i \in r} y_i = w(R)$ and $y_i > x_i$ for all $i \in R$. A counter-objection to the objection is a pair (z, T) , where T is a coalition containing l and not containing k and z is a vector in \mathbb{R}^T satisfying $\sum_{i \in t} z_i = w(T)$, $z_i \geq y_i$ for all $i \in R \cap T$, and $z_i \geq x_i$ for all $i \in T \setminus R$. A payoff vector x belongs to the classical bargaining set $M_1^{(i)}$, if there exists a counter-objection for any objection of one player against another at x .

In a multi-choice game with coalition structure, each player can work at several activity levels, and the player levels of T may be different from those of R . So it is not reasonable that $\sum_{j=1}^{t_l} z_{ij} \geq \sum_{j=1}^{r_l} y_{ij}$ for all $i \in car(R) \cap car(T)$ in counter-objection, and the classical bargaining set can not be directly applied to multi-choice game with coalition structure. But we notice that the counter-objection (z, T) can be reasonable only when $\sum_{j=1}^{t_l} (z_{ij} - x_{ij}) \geq \sum_{j=1}^{r_l} (y_{ij} - x_{ij})$ for all $i \in car(R) \cap car(T)$ and $\sum_{j=1}^{t_l} z_{ij} \geq \sum_{j=1}^{t_l} x_{ij}$ for all $i \in car(T) \setminus car(R)$, where $\sum_{j=1}^{t_l} (z_{ij} - x_{ij})$ denotes a payoff gain.

Let v be a multi-choice game with coalition structure. Then, objection, counter-objection and bargaining set in multi-choice game with coalition structure can be defined as Definition 1, Definition 2 and Definition 3.

Definition 1. Let $x \in I(v, \beta)$ and $k, l \in B \in \beta$, $k \neq l$. An objection of one player k against another player l at x is a pair (y, R) , where $R \in M_0^N$ with $r_k > 0, r_l = 0$ and y is a payoff vector of the coalition R satisfying Eq. 2 and Eq. 3.

$$\sum_{i \in car(R)} \sum_{j=1}^{r_i} y_{ij} = v(R) \quad (2)$$

$$\sum_{j=1}^{r_i} y_{ij} > \sum_{j=1}^{r_i} x_{ij} \quad \text{for all } i \in car(R) \quad (3)$$

Definition 2. Let (y, R) be an objection of k against l at x . A counter-objection to the objection is a pair (z, T) , where $t \in M_0^N$ with $t_k = 0, t_l > 0$ and z is a payoff vector of the coalition T satisfying Eq. 4, Eq. 5 and Eq. 6.

$$\sum_{i \in car(T)} \sum_{j=1}^{t_i} z_{ij} = v(T) \quad (4)$$

$$\sum_{j=1}^{t_i} (z_{ij} - x_{ij}) \geq \sum_{j=1}^{r_i} (y_{ij} - x_{ij}) \quad \text{for all } i \in car(R) \cap car(T) \quad (5)$$

$$\sum_{j=1}^{t_i} z_{ij} \geq \sum_{j=1}^{r_i} x_{ij} \quad \text{for all } i \in car(T) \setminus car(R) \quad (6)$$

Definition 3. A payoff vector $x \in I(v, \beta)$ belongs to bargaining set for v , if there exists a counter-objection for each objection of k against l at x . The bargaining set can be denoted as $M(v, \beta)$.

There is a counter-objection for each objection in v , so $M(v, \beta)$ is a set of stable payoff vectors. When each player can work at only two levels, active and inactive, $M(v, \beta)$ is same as the classical bargaining set $M_1^{(i)}$, that is to say, $M(v, \beta)$ is a generalization of $M_1^{(i)}$.

3 Finding a Stable Payoff Vector

In fact, the definition of the bargaining set does not even provide a method to compute any stable payoff vector to the players. It only provides a method to test a

payoff vector whether to belong to the bargaining set or not. With some following definitions and theorems, we give an approach to obtain a stable payoff vector belonging to the bargaining set based on genetic algorithm.

Excess and surplus in classical cooperative games have been defined by Morton Davis and Michael Maschler [5]. Firstly, we denote the concepts of excess and surplus in a multi-choice game with coalition structure, and give a relative theorem as Theorem 1 and its proof.

Let $S \in M^N$ and $x \in I(v, \beta)$. The excess of S at x is $e(x, S) = v(S) - \sum_{i \in car(S)} \sum_{j=0}^{s_i} x_{ij}$. Let $S \in M^N, k \neq l, s_k > 0, s_l = 0$ and $x \in I(v, \beta)$. The surplus of k over l at x is $S_{kl}(x) = \max e(x, S)$.

Note that, the coalition s can be used as an objection if and only if $e(x, S) > 0$. If $S_{kl}(x) > 0$, there exists a positive $e(x, s)$, where $S \in M^N, k \neq l, s_k > 0, s_l = 0$, and then there exists an objection of k against l at x . So $S_{kl}(x)$ can be as a strength of k against l at x .

Theorem 1. Let v be a multi-choice game with coalition structure. Let $x \in I(v, \beta)$ and $k, l \in B \in \beta$, where $k \neq l$. If $S_{kl}(x) \leq S_{lk}(x)$, there exists a counter-objection for each objection of k against l at x .

Proof Let $x \in I(v, \beta)$ and $k, l \in B \in \beta$, where $k \neq l$. We can get several formulas as Eq. 7, Eq. 8, Eq. 9 and Eq. 10.

$$S_{kl}(x) = \max e(x, S) = e(x, R), \text{ for all } S \in M^N, s_k > 0, s_l = 0 \quad (7)$$

$$S_{lk}(x) = \max e(x, S) = e(x, T), \text{ for all } S \in M^N, s_l > 0, s_k = 0 \quad (8)$$

$$S_{kl}(x) \leq S_{lk}(x) \quad (9)$$

$$e(x, R) \leq e(x, T) \quad (10)$$

Suppose there exists a justified objection (y, R) of k against l at x . We define a payoff vector of coalition T as z , satisfying Eq. 11.

$$\sum_{j=1}^{t_l} z_{ij} = \begin{cases} \sum_{j=1}^{t_l} x_{ij}, & \text{if } i \in car(T) \setminus car(R) \text{ and } i \neq l \\ \sum_{j=1}^{r_l} (y_{ij} - x_{ij}) + \sum_{j=1}^{t_l} x_{ij}, & \text{if } i \in car(T) \cap car(R) \\ v(t) - \sum_{i \in car(T) \setminus \{l\}} \sum_{j=1}^{t_l} z_{ij}, & \text{if } i = l \end{cases} \quad (11)$$

Next, we need prove $\sum_{j=1}^{t_l} z_{lj} \geq \sum_{j=1}^{t_l} x_{lj}$ as Eq. 12.

$$\begin{aligned}
\sum_{j=1}^{t_l} z_{lj} - \sum_{j=1}^{t_l} x_{lj} &= v(t) - \sum_{i \in car(T) \setminus \{l\}} \sum_{j=1}^{t_l} z_{ij} - \sum_{j=1}^{t_l} x_{lj} \\
&= v(t) - \left(\sum_{j \in car(T) \cap car(R)} \sum_{j=1}^{t_l} (y_{ij} - x_{ij}) + \sum_{j \in car(T) \cap car(R)} \sum_{j=1}^{t_l} x_{ij} \right) - \sum_{i \in car(T) \setminus car(R)} \sum_{j=1}^{t_l} x_{ij} \\
&= v(t) - \left(\sum_{i \in car(T) \setminus car(R)} \sum_{j=1}^{t_l} x_{ij} + \sum_{i \in car(T) \setminus car(R)} \sum_{j=1}^{t_l} x_{ij} \right) - \sum_{i \in car(T) \cap car(R)} \sum_{j=1}^{t_l} (y_{ij} - x_{ij}) \\
&= (v(t) - \sum_{i \in car(T)} \sum_{j=1}^{t_l} x_{ij}) - (v(r) - \sum_{i \in car(R) \setminus car(T)} \sum_{j=1}^{t_l} y_{ij} - \sum_{i \in car(T) \setminus car(R)} \sum_{j=1}^{t_l} x_{ij}) \\
&\geq (v(t) - \sum_{i \in car(T)} \sum_{j=1}^{t_l} x_{ij}) - (v(r) - \sum_{i \in car(R)} \sum_{j=1}^{t_l} x_{ij}) \\
&= e(x, T) - e(x, R) \\
&\geq 0
\end{aligned} \tag{12}$$

That is to say, (z, T) is a counter-objection to (y, R) . Hereto, we obtain a contradiction about the above supposing. We can also deduce that there does not exist any k and l satisfying $S_{kl}(x) > S_{lk}(x)$, where $k, l \in B \in \beta$ and $k \neq l$, if $x \in M(v, \beta)$, where $x \in I(v, \beta)$.

In the cooperative games, Schmeidler [10] proposed the definition of the lexicographical ordering. Let v be a multi-choice game with coalition structure and $x \in I(v, \beta)$. The lexicographical ordering $\theta(x)$ of x can be defined as $\theta(x) = (\theta_1(x) = e(S_1, x), \theta_2(x) = e(S_2, x), \dots, \theta_k(x) = e(S_k, x), \dots, \theta_H(x) = e(S_H, x))$, where the various excesses of coalitions are arranged in decreasing order. When $\theta_i(x) < \theta_j(y)$ for $j=t$ and $\theta_j(x) = \theta_j(y)$ for $1 \leq j < t$, where $1 \leq t \leq H$, $\theta(x)$ is lexicographically smaller than $\theta(y)$, denoted as $\theta(x) <_L \theta(y)$, where $x, y \in I(v, \beta)$. When $\theta(x) <_L \theta(y)$, the most dissatisfaction degree of players to the payoff vector y is commonly more than that to the payoff vector x , that is to say, players can be more willing to accept the more stable payoff vector x . So the stability of a payoff vector can be judged in the means of comparing lexicographical ordering.

Based on Theorem 1, lexicographical ordering and genetic algorithm, we can find a stable payoff vector in a multi-choice game with coalition structure, which belongs to the bargaining set $M(v, \beta)$, where there does not exist any k and l satisfying $S_{kl}(x) > S_{lk}(x)$, where $k, l \in B \in \beta$ and $k \neq l$.

The genetic algorithm is a global random optimization method, which has so many advantages, such as concurrent searching and colony optimize, and has been applied successfully in many fields of NP complete problems. In the process of finding a stable payoff vector, we need several important concepts in the genetic algorithm.

Individual Coding: An individual is coded in the form of floating point number, which is a payoff vector belonging to $I(v, \beta)$.

Fitness: We randomly select H coalitions. The lexicographical orderings of any individual to these selected coalitions can be obtained. The fitness of an individual is the order number in descending order of the lexicographical orderings about all individuals of a population. The less the order number of an individual is, the larger its fitness is.

Selection: Population can be selected according as the fitness in the form of a rotating disc game.

Crossover: We randomly select two individual, which are two payoff vectors, x and y . We randomly obtain a coalition B from the coalition structure β , where $B \in \beta$. Then, at each level of every member in B , two corresponding payoff values in x and y are reset to their average value.

Mutation: We randomly select H coalitions and two members, $k, l \in B \in \beta$, and solve $S_{kl}(x) = e(x, S)$ and $S_{lk}(x) = e(x, T)$ aiming to the selected coalitions, where $k, l \in B \in \beta$. If $|S_{kl}(x) - S_{lk}(x)| > \alpha$, the payoff values at $1, 2, \dots, \min(s_k, t_l)$ levels of k respectively increase δ and the payoff values at $1, 2, \dots, \min(s_k, t_l)$ levels of l respectively decrease δ , when the level increase rational can be satisfied, where $\alpha > 0$ and $\alpha \geq \delta > 0$. If $|S_{kl}(x) - S_{lk}(x)| \leq \alpha$, the individual need not mutate.

The best stable payoff vector can be obtained, when no selected individual mutates for continuous w generation times, that is to say, $S_{kl}(x) = S_{lk}(x)$ for every selected individual. The method to find the best stable payoff vector is showed as Algorithm 1.

Algorithm 1.

Step 1. Initialize a population.

Step 2. Calculate the fitness for each individual in the population.

Step 3. Select the population in the form of a rotating disc game.

Step 4. Generate new population by randomly pair-matched crossover.

Step 5. Select an individual from the population randomly. $num = 0$ when the individual can mutate, otherwise $num = num + 1$.

Step 6. Goto Step 2, if the generation number is not reached, and $num < w$. Otherwise, the best stable payoff vector can be obtained, whose lexicographical ordering is smallest.

4 Experiments

Example 1. Let (N, v, β) be a multi-choice game with coalition structure, where $N = \{1, 2, 3\}$, $M = (2, 2, 2)$ and characteristic function v can be defined as Eq. 13.

$$v(S) = \begin{cases} 0 & \text{if } |car(S)| \leq 1 \\ 30 & \text{if } |car(S)| \geq 2, \text{and } s_i \leq 1 \\ 40 & \text{if } |car(S)| \geq 2, \exists! s_i = 2 \\ 45 & \text{otherwise} \end{cases} \quad (13)$$

When $\beta = \{B_1, B_2\} = \{(1, 2), (3)\}$, we can obtain the imputation set $I(v, \beta)$ as Eq. 14.

$$I(v, \beta) = \{x \in R^{M^N} \mid \sum_{i=1}^2 \sum_{j=1}^2 x_{ij} = 45, \sum_{j=1}^2 x_{3j} = 0, \text{and } x_{ij} \geq 0, \text{for all } i \in N, j \in M_i^+\} \quad (14)$$

By Algorithm 1, we can find the best stable payoff vector belonging to the bargaining set and make performance evaluations. Our experiments are based on the machine platform of Intel Core2 Duo P8600 CPU 2.40GHz, 2GB main memory and Windows 7 operating system. And all codes were written in Matlab 7.6.0(R2008a). The detailed steps solving the best stable payoff vector are as follows.

Step 1: Initialize a population as the following four individuals.

$$\begin{aligned} x_1 &= \begin{pmatrix} 6.7460 & 8.9452 \\ 12.6063 & 16.7025 \\ 0 & 0 \end{pmatrix} & x_2 &= \begin{pmatrix} 13.5562 & 16.1163 \\ 10.9443 & 4.3832 \\ 0 & 0 \end{pmatrix} \\ x_3 &= \begin{pmatrix} 2.4726 & 16.8925 \\ 15.1714 & 10.4635 \\ 0 & 0 \end{pmatrix} & x_4 &= \begin{pmatrix} 2.8909 & 18.4406 \\ 15.3187 & 8.3498 \\ 0 & 0 \end{pmatrix} \end{aligned}$$

Step 2: The incremental lexicographical orderings of the four individuals is as $\theta(x_2) <_L \theta(x_1) <_L \theta(x_4) <_L \theta(x_3)$. By the lexicographical orderings, four corresponding fitnessnes can be calculated as follows.

$$\begin{aligned} f(x_1) &= \frac{3}{1+2+3+4} = 0.3 & f(x_1) &= \frac{4}{1+2+3+4} = 0.4 \\ f(x_3) &= \frac{1}{1+2+3+4} = 0.1 & f(x_4) &= \frac{2}{1+2+3+4} = 0.2 \end{aligned}$$

Step 3: Obtain the selected population in the form of a rotating disc game as follows.

$$\begin{aligned} x_1 &= \begin{pmatrix} 6.7460 & 8.9452 \\ 12.6063 & 16.7025 \\ 0 & 0 \end{pmatrix} & x_2 &= \begin{pmatrix} 13.5562 & 16.1163 \\ 10.9443 & 4.3832 \\ 0 & 0 \end{pmatrix} \\ x_3 &= \begin{pmatrix} 13.5562 & 16.1163 \\ 10.9443 & 4.3832 \\ 0 & 0 \end{pmatrix} & x_4 &= \begin{pmatrix} 6.7460 & 8.9452 \\ 12.6063 & 16.7025 \\ 0 & 0 \end{pmatrix} \end{aligned}$$

Step 4: We randomly select two pairs of crossover individuals, x_1 and x_4 , x_2 and x_3 . Due to $x_1 = x_4$ and $x_2 = x_3$, the crossovers need not to be done and the new population is retained as follows.

$$x_1 = \begin{pmatrix} 6.7460 & 8.9452 \\ 12.6063 & 16.7025 \\ 0 & 0 \end{pmatrix} \quad x_2 = \begin{pmatrix} 13.5562 & 16.1163 \\ 10.9443 & 4.3832 \\ 0 & 0 \end{pmatrix}$$

$$x_3 = \begin{pmatrix} 13.5562 & 16.1163 \\ 10.9443 & 4.3832 \\ 0 & 0 \end{pmatrix} \quad x_4 = \begin{pmatrix} 6.7460 & 8.9452 \\ 12.6063 & 16.7025 \\ 0 & 0 \end{pmatrix}$$

Step 5: We randomly select a individual x_1 and two members 1 and 2, and solve $S_{12}(x_1) = e(x, S) = 33.2540$ and $S_{21}(x_1) = e(x, T) = 27.3937$, where $S = (1, 0, 2)$, $T = (0, 1, 2)$, and $\min(s_k, t_l) = 1$. Due to $|S_{12}(x_1) - S_{21}(x_1)| > \alpha$, x_1 can mutate to induce $x_{11} = x_{11} + \delta$ and $x_{21} = x_{21} - \delta$, where $\alpha = 0.01$ and $\delta = 0.005$. The mutation of x_1 leads to $num = 0$, and the mutated population is as follows.

$$x_1 = \begin{pmatrix} 6.7510 & 8.9452 \\ 12.6013 & 16.7025 \\ 0 & 0 \end{pmatrix} \quad x_2 = \begin{pmatrix} 13.5562 & 16.1163 \\ 10.9443 & 4.3832 \\ 0 & 0 \end{pmatrix}$$

$$x_3 = \begin{pmatrix} 13.5562 & 16.1163 \\ 10.9443 & 4.3832 \\ 0 & 0 \end{pmatrix} \quad x_4 = \begin{pmatrix} 6.7460 & 8.9452 \\ 12.6063 & 16.7025 \\ 0 & 0 \end{pmatrix}$$

Step 6: Goto Step 2 because of $num < w$, where $w = 150$. When the iterative times come to 5513, $num = w$. The individual is as follows, which the best stable payoff vector is and whose lexicographical ordering is smallest.

$$x = \begin{pmatrix} 11.6095 & 14.3235 \\ 11.6039 & 7.4630 \\ 0 & 0 \end{pmatrix}$$

Here, the player 2 (or 1) has always a counter-objection for each objection of the player 1 (or 2) against the player 2 (or 1) at x , so x belongs to the bargaining set.

In Example 1, we can get the corresponding generation times and Success rates to obtain a stable payoff vector by changing the values of three parameters α , δ and w . When α varies in 0.01, 0.05, 0.1, 0.15, 0.2 and 0.25, two curves in Fig. 1 show the generation times respectively under $\delta = 0.005$ and $\delta = 0.01$. For a fixed δ , the generation times are gradually decreasing with the increasing of α . For a fixed α , the generation times are gradually decreasing with the increasing of δ .

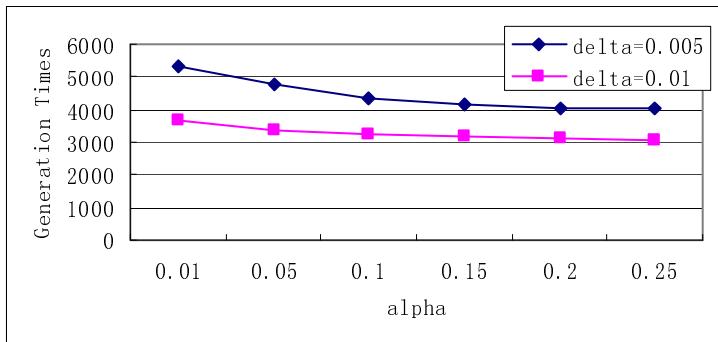


Fig. 1. Generation times with the variation of α under $\delta = 0.005$ and $\delta = 0.01$

When w varies in 10, 50, 100, 150, 200 and 250, two curves in Fig. 2 show the generation times respectively under $\delta = 0.005$ and $\delta = 0.01$. For a fixed δ , the generation times are gradually increasing with the increasing of w . For a fixed w , the generation times are gradually decreasing with the increasing of δ .

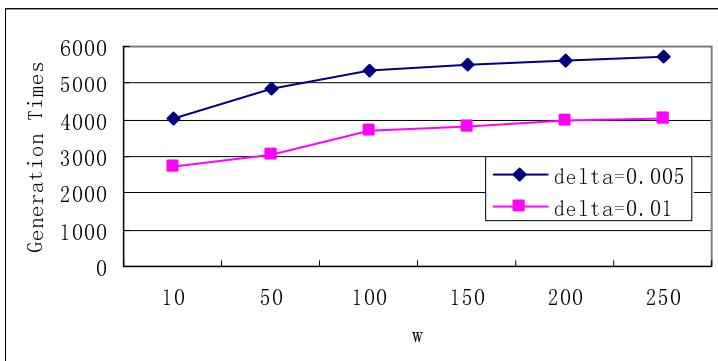


Fig. 2. Generation times with the variation of w under $\delta = 0.005$ and $\delta = 0.01$

When w varies in 10, 50, 100, 150, 200 and 250, the curve in Fig. 3 shows the success rates with the variation of w . For a fixed w , generating a stable payoff vector using Algorithm 1 is regarded as a success, and the ratio of success times to the total test times is named as the success rate under the w . Success rates are gradually increasing with the increasing of w .

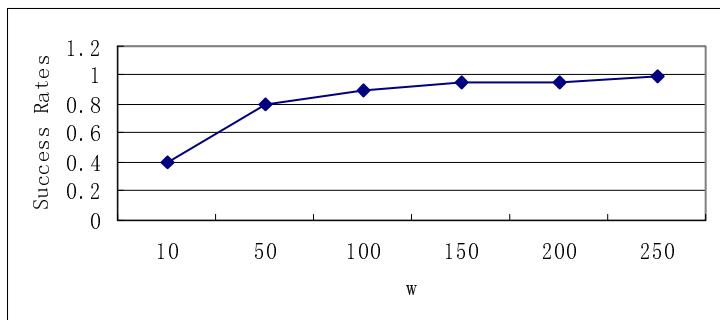


Fig. 3. Success rates with the variation of W

5 Conclusions

We summarize the main contributions of this paper.

- We extend the solution of bargaining set in classical cooperative games to the multi-choice games with coalition structure, which is a generalization of the classical bargaining set.
- We give a approach to find a stable payoff vector belonging to the extended bargaining set based on genetic algorithm.
- The experiment shows the process of finding a stable payoff vector.

Acknowledgements. This work is supported by the National Natural Science Foundation of China (Nos. 61063009, 61163003), and the Ph.D. Programs Foundation of Ministry of Education of China (No.2010531120001), the Yunnan Provincial Foundation for Leaders of Disciplines in Science and Technology (2012HB004), the Foundation for Key Program of Department of Education of Yunnan Province (2011Z015).

References

1. Auman, R.J., Maschler, M.: The bargaining sets for cooperative games. In: Dresher, M., Shapley, L.S., Tucker, A.W. (eds.) *Advances in Game Theory*, pp. 443–476. Princeton University Press, Princeton (1964)
2. Branzei, R., Dimitrov, D., Tijs, S.: *Models in Cooperative Game Theory*, 2nd edn. Springer, Berlin (2008)
3. Branzei, R., Llorca, N., Sánchez-Soriano, J., Tijs, S.: Multi-choice clan games and their core. *Top* 17, 123–138 (2009)
4. Calvoa, E., Santosb, J.C.: A value for multichoice games. *Mathematical Social Sciences* 40, 341–354 (2000)
5. Davis, M., Maschler, M.: Existence of stable payoff configuration for cooperative games. *Bulletin of the American Mathematical Society* 69, 106–108 (1963)

6. Grabisch, M., Xie, L.: A new approach to the core and Weber set of multichoice games. *Math. Meth. Oper. Res.* 66, 491–512 (2007)
7. Grabisch, M., Fabien, L.: Games on lattices, multichoice games and the Shapley value: a new approach. *Math. Meth. Oper. Res.* 65, 153–167 (2007)
8. Hsiao, C.R., Raghavan, T.E.S.: Monotonicity and dummy free property for multi-choice cooperative games. *Int. J. Game Theory* 21, 301–302 (1992)
9. Hsiao, C.R., Raghavan, T.E.S.: Shapley value for multichoice cooperative games. *Games Econ. Behav.* 5, 240–256 (1993)
10. Schmeidler, D.: The Nucleolus of a characteristic function Game. *SIAM Journal of Applied Mathematics* 17, 1163–1170 (1969)
11. Hwang, Y.-A., Li, W.-H.: The core of multi-choice NTU games. *Math. Meth. Oper. Res.* 61, 33–40 (2005)
12. Hwang, Y.-A., Liao, Y.-H.: Equivalence theorem, consistency and axiomatizations of a multi-choice value. *J. Glob. Optim.* 45, 597–613 (2009)
13. Jones, M.A., Wilson, J.M.: Multilinear extensions and values for multichoice games. *Math. Meth. Oper. Res.* 72, 145–169 (2010)
14. Liu, J., Liu, X.: Fuzzy extensions of bargaining sets and their existence in cooperative fuzzy games. *Fuzzy Sets and Systems* (2011), doi:10.1016/j.fss.2011.08.001
15. Van den Nouweland, A., Tijs, S., Potters, J., Zarzuelo, J.: Cores and related solution concepts for multichoice games. *Math. Meth. Oper. Res.* 41, 289–311 (1995)
16. Jin, N., Tsang, E.: Bargaining strategies designed by evolutionary algorithms. *Applied Soft Computing* 11(8), 4701–4712 (2011)
17. Shehory, O., Kraus, S.: Feasible formation of coalitions among autonomous Agents in nonsuperadditive environments. *Computational Intelligence* 15(3) (1999)
18. Castillo, L., Dorao, C.A.: Consensual decision-making model based on game theory for LNG processes. *Energy Conversion and Management* 64, 387–396 (2012)

An Optimized Method for Selection of the Initial Centers of K-Means Clustering

YongJun Zhang^{1,2} and Enxiu Cheng²

¹ Shandong Institute of commerce and technology, Jinan 250103, China

² National Engineering Research Center for Agricultural Products Logistics, Jinan, China

zyj420177@163.com

Abstract. At present, K-Means algorithm as a method of clustering based on partition has more applications. By analyzing the problem of K-Means, we propose a kind of optimized algorithm to select initial clustering points, which utilize an adjacent similar degree between points to get similarity density, then by means of max density points selecting our method heuristically generate clustering initial centers to get more reasonable clustering results. This method compares to traditional methods lie in it will get suitable initial clustering centers and also has more stable clustering result. Finally, through comparative experiments we prove the effectiveness and feasibility of this algorithm.

Keywords: initial center, similarity density, clustering evaluation coefficient.

1 Introduction

Clustering analysis is unsupervised machine learning method and it is also an active branch in data mining has very extensive applications. An algorithm for clustering depends the rule of like attracts like, to classify a group of individuals by certain similarities, to make the difference of some individuals belong to one group as small as possible, while the difference among individuals belong to different groups is larger. In order to clustering data objects, nowadays there are many classical algorithms; it can be roughly classified, which based on partition, hierarchy, density, grid and model [1] [2]. In there, algorithm for K-Means clustering [3][4][5] applies very extensively, which based on partitions and it has many advantages such as conciseness and quick speed. It can show high performances of clustering in geometry and statistics, especially for dataset with attributes of numerical value. K-Means also has its drawbacks [6][7][8][9]. The concerned algorithms need the setting of parameter K, what is worse it unsteadily chooses initial centers. The parameter of clustering number is very difficult to guarantee because we lack the knowledge of data distribution. By full analyzing the problem of K-means , this paper does improvement in three aspects : 1) In according to objects similar density, we propose an initial centers selection method ; 2) and design a new evaluating coefficient to measure clustering quality; 3) heuristically generate the number of clustering results.

2 Problem Description

Presumably the analysis of the data is denoted by an array of vector $X = \{x_1, \dots, x_n\}$ and each element in X is the m-dimensional vector, $C_j (j=1,2,\dots,k)$ denotes the sub clusters of clustering, $c_j (j=1,2,\dots,k)$ is initial clustering centers, the Euclidean distance between two vectors is $d(x_i, x_j) = \sqrt{\sum_{i=1}^n (x_i - x_j)^2}$, the clustering centers is $c_j = \frac{1}{n_j} \sum_{x \in C_j} x$, the main idea of K-Means is to divided iteratively objects into different clusters so as to make the object function $E = \sum_{i=1}^k \sum_{j=1}^{n_j} d(x_j, c_i)$ to reach the minimum and to make the clusters more compact. K-Means algorithm need the number of clustering and the dataset with n objects, final output minimal k clusters that contents the object function. Firstly, we pick up k objects from n objects as clustering initial centers; repeatedly to calculate the distance from every object to centers, and to partition objects by the shortest distance between objects until the object function E is unchangeable. Because the clustering center is randomly chosen, which may lead to the instability of clustering result and what is worse it cannot get the optimized global solution. So it is very important how to select initial centers so as to achieve better cluster results, to eliminate the instability of K-Means.

3 Definitions of Optimized Method

Definition 1. We use symbol $Sim(x_i, x_j)$ to represent clustering similarity degree. We can define the clustering dataset $X = \{x_1, \dots, x_n\}$, in there, any pairs of data x_i and x_j the similarities between them can be denoted in formula

$$Sim(x_i, x_j) = \lambda d_s(x_i, x_j) + (1 - \lambda) |\cos(x_i, x_j)| \quad (1)$$

In there, $d_s(x_i, x_j) = \frac{d(x_i, x_j) - \min_{x_i, x_j \in X} \{d(x_i, x_j)\}}{\max_{x_i, x_j \in X} \{d(x_i, x_j)\} - \min_{x_i, x_j \in X} \{d(x_i, x_j)\}}$ is normalized Euclidean distance, the symbol $d(x_i, x_j)$ is $\sqrt{\sum_{k=1}^m (x_i - x_j)^2}$ that is Euclidean distance; the coefficient λ is the weighting factors of value and changing between vectors, which range from zero to one. On the basis of experience, we always set λ to 0.5. This parameter usually is acquired by experiential computation that is more suitable for clustering analysis. The symbol $\cos(x_i, x_j)$ is the cosine of the intersection angle of two vectors,

can be computed by following formula $\frac{\sum_{i=1}^m (x_i \cdot x_j)}{\sqrt{\sum_{i=1}^m x_i^2} \sqrt{\sum_{j=1}^m x_j^2}}$ and the value $\cos_{x_i, x_j \in X}(x_i, x_j)$ ranges from -1 to 1. Traditional cluster analysis often utilizes factor of distance.

(Euclidean distance, Mahalanobis distance, etc.) or similar factor (correlation coefficient, cosine value, etc.) to measure the similarities of vectors. No matter whatever statistical tools do you take, which does not take the two aspects into account, that is to say we should not only weight difference of value between vectors, but also needs to think it's changing of values in each dimension among those vectors. To calculate the similarity degree of the vector set x , we can use Lower triangular matrix $SimMatrix[n][n]$ to denote similar matrix because for any two vectors' similarity degree $Sim_{x_i, x_j \in X}(x_i, x_j)$ is equivalent to $Sim_{x_i, x_j \in X}(x_j, x_i)$.

Definition 2. The similar neighborhood of vector x_i in vector set x is denoted by $SimNeighor_{x_i, \alpha}$. For every object x_i , take x_i as center, the clustering similarity degree of all objects to x_i larger than the given threshold value α is the similar neighborhood to x_i . The formal definition as:

$$SimNeighor(x_i, \alpha) = \{x | \alpha \leq sim_{x_i, x \in X}(x_i, x) \leq 1, 0 \leq \alpha \leq 1, X = \{x_1, \dots, x_n\}\} \quad (2)$$

Definition 3. The similarity density of vector x_i is $Density_{x_i}$ that is under the condition $SimNeighor(x_i, \alpha)$ to x_i . The density $Density_{x_i}$ be denoted as a formula:

$$Density_{x_i} = \frac{\sum_{j=1}^{|p_{neighbor(x_i)}|} Sim(x_i, p_{neighbor(x_i)}^j)}{|p_{neighbor(x_i)}|}, X = \{x_1, \dots, x_n\}. \quad (3)$$

The symbol $p_{neighbor(x_i)}^j \in SimNeighor(x_i, \alpha)$ denotes these vectors satisfy the threshold α in a similar neighborhood x_i . Absolute value $|p_{neighbor(x_i)}|$ is the number of objects in the neighbor of the object x_i .

Definition 4. Clustering homogeneity defines the average similar degrees belong to the same group. Suppose there are two objects x_i, x_j , the clustering similarity degree is $0 \leq sim_{x_i, x_j \in X}(x_i, x_j) \leq 1$, the clustering homogeneity of cluster C_p can be expressed by formally in $Hom(C_p) = \frac{1}{|C_p|^2} \sum_{i=1}^{|C_p|} \sum_{j=1}^{|C_p|} sim_{x_i, x_j \in X}(x_i, x_j)$. Clustering homogeneity value is greater, the objects of cluster more closely.

Definition 5. Clustering heterogeneity is $Het(C_p, C_q) = \frac{1}{|C_p||C_q|} \sum_{i=1}^{|C_p|} \sum_{j=1}^{|C_q|} sim(x_i, x_j)$. That is analogous to the clustering homogeneity. For the given two clusters C_p, C_q , The above-mentioned formula used to calculate the average similarity of the different clustering objects.

4 Evaluation Coefficient of Clustering Result

The goal of clustering is to give an dataset an optimal partition. That is to divide the similar data with certain similarities into a same class as much as possible, and vice versa. The clustering evaluation coefficient obj is defined formally as follows:

$$obj = \mu \times \text{Compactness}(C) \times \text{Separation}(C) \quad (4)$$

It is not very difficult to find the qualities of clustering can be evaluated by compactness of inner cluster and the separated degree between clusters. Compactness reflects the similarity of members of the same cluster, which denotes as $\text{Compactness}(C) = \sum_{i=1}^k \frac{\text{Hom}(C_i)}{k}$. The separated degree can be expressed as $\text{Separation}(C) = 1 - \sum_{i=1}^k \sum_{j=1}^k \frac{HET(C_i, C_j)}{k}$, it manifests the similarities among the members of different classes. We hope certain partition which has large inner clustering compactness and also has fewer similarities among the members of different clusters. Therefore, we cannot difficultly discover the larger the value obj is, the better the clustering result will get. In formula obj , the coefficient μ is an adjustment coefficient for the convenience of computing. After utilizing clustering evaluation coefficient obj we can overcome the influence of the number of clustering results.

5 Description of Initial Centers Selection of K-Means Algorithm

The basic idea of this algorithm is to calculate the similarity matrix $SimMatrix[n][n]$ firstly that is vector x with m dimensions and use an array X' to record the identity of vectors to select candidate initial centers. To use X' by means of $SimMatrix$ to choose the max similar density point as a first initial center in the specified similar neighborhood. Next we will do the K-Means algorithm in the original vector set and then compute the clustering centers through clustering operation. When finishing calculation of the evaluating coefficient we delete point x_i and all objects that satisfy the constraint $SimNeighor(x_i, \alpha)$, then to acquire next initial center according to above method. Repeatedly do clustering operation until the center point set $initC$ has K points and if the current clustering evaluation coefficient

subtract previous one is small enough. Finally we stop the iteration and get the clustering result.

6 Optimized Selection of the Initial Centers of K-Means

The algorithm of heuristic selection of K initial center as following:

Input: An array of n objects x with m dimensions and its index array X , similar neighborhood threshold α , similarity weighting factors λ , initial centers for clustering $initC = \emptyset$, the evaluation coefficient of the quality of clustering is $oldobj = \infty$ and $newobj = 0$

Output: A number of K result set C and $C_i \cap C_j = \emptyset$, $0 \leq i, j \leq K$, $i \neq j$;

Method:

Step1. To compute every pair of two objects in vector set X according to clustering similarity degree and save them in a similar matrix $SimMatrix[n][n]$;

Step2. For every object $x_i \in X$, to select vector x_i with $\max_{x_j \in X} Density(x_i)$ by referring

the similar matrix $SimMatrix$ that is larger than a similar neighborhood threshold α ;

Step3. To delete vector x_i in X and all vectors in $SimNeighbor(x_i, \alpha)$, update array X
do $X \leftarrow X - \{x_i\} - \{SimNeighbor(x_i, \alpha)\}$;

Step4. To select point $x_i \in X$, set $initC \leftarrow x_i \cup initC$, utilize all points of $initC$ as K-means initial centers to do clustering operation with facility by $SimMatrix[n][n]$, then save the current clustering results in C to reset $initC$ by the latest calculated centers and compute the current evaluation coefficient $newobj$;

Step5. If $|newobj - obj| \leq \zeta$ or the number of loops is bigger than number n , the algorithm will finish, or $oldobj = newobj$ and will go to Step2.

This algorithm heuristically finds the max similarity density points to guide the initial center selection that is very meaningful to K-Means clustering. In this paper we set boundary ζ to 0.001 as the end of iteration.

7 Analysis of Experiments

7.1 The Error of Clustering Result

We can analyze the quality of clustering result of K-Means and use a symbol $|C^{error}|$ to indicate the number of wrong classification in one of the whole partitions. It is supposed that the real number of classification is k , clustering result formally denotes by $C = \{c_1, \dots, c_k\}$. In every partition c_i , the quantity of elements in a partition is defined as $|c_i|$, $1 \leq i \leq k$ and to express the clustering error rate by the symbol

$er = \frac{1}{k} \sum_{i=1}^k \frac{|C_i^{error}|}{|c_i|}$, which describes the case of wrong clustering. Obviously, the wrong partition smaller clustering effect is better.

7.2 Experimental Environment and Results of Comparisons

In order to verify the effectiveness of this algorithm, we use Java development toolkit JDK6.0, Intel core i7, 2.67GHZ CPU, 4GMB memory. We take two arrays of dataset Iris and water-treatment as test dataset. Database of UCI is a special database for testing of machine learning and data mining, all its data has their classifications so it is self-evident to show the quality of clustering results. We can compare our optimized algorithm with traditional K-Means such as the methods of Euclidean distance, the Correlation coefficient and the Cosine of vectors. The real centers in dataset Iris are (5.00,3.42,1.46,0.24), (5.93,2.77,4.26,1.32), (6.58,2.97,5.55,2.02). From Table1, by means of different methods to partition we can see the distance to the real clustering centers. It can be concluded from Table1 that the error of our proposed method is minimal and efficient.

Table 1. Standard deviation comparison of the various clustering methods to deviate from the actual center of the data set Iris.(The real number of clustering $K=3$, $\alpha=0.75$, $\lambda=0.5$)

Euclidean distance	Correlation coefficient	Cosine value	Our method
0.233	0.172	0.145	0.0568

As the result is shown in Table2 we can find our method is convergent at K is equal to 14 that is a little bit more than the real cluster number 13. So we can list the values of K are 12, 13 and 14 to analyze wrong clustering. In contrast to other random selection method of clustering results, we find the initial point selection method for data clustering analysis of Water-treatment dataset is equally effective. Our method can get the more reasonable centers than randomly selected methods. We use a similarity density algorithm to pick initial centers to perform K-Means clustering after setting coefficients α , λ and the value of α , λ is aim to optimize the evaluation coefficient. When the value of K is stable the error of our method to reach minimal.

Table 2. Error rate comparison clustering results of various clustering methods for water treatment data. (The real number of clustering $K=13$, $\alpha=0.75$, $\lambda=0.5$)

Euclidean distance	Correlation coefficient	Cosine value	Our method
0.310 ($k=12$)	0.375 ($k=12$)	0.323 ($k=12$)	0.069 ($k=12$)
0.273 ($k=13$)	0.229 ($k=13$)	0.233 ($k=13$)	0.059 ($k=13$)
0.255 ($k=14$)	0.233 ($k=14$)	0.276 ($k=14$)	0.054 ($k=14$)

From the view of experience, we can analyze sample data to get proper value of parameters α , λ before we really analyze large data set. After adjusting coefficient α to 0.56 and λ to 0.6, which will get a better result. The comparative result can be seen from below Table3.

Table 3. The clustering error rate comparison after the parameter adjustment(The real number of clustering K=13, $\alpha=0.56$, $\lambda=0.6$)

Euclidean distance	Correlation coefficient	Cosine value	Our method
0.387 (k=11)	0.385 (k=11)	0.256 (k=11)	0.087 (k=11)
0.310 (k=12)	0.275 (k=12)	0.223 (k=12)	0.074 (k=12)
0.273 (k=13)	0.229 (k=13)	0.237 (k=13)	0.062 (k=13)

8 Conclusions

In order to get better result the traditional clustering need user to set the number of partitions, it will hinder the development of many applications. The choice of initial centers is unreasonable will lead to unstable of final clustering result and very vulnerable trapped to local extreme points. The traditional methods of evaluation are not sound enough, so it is difficult to get optimal clustering result. In this paper we propose an initial centers selection method that is based on density of similarity to selection which heuristically generate the number of clustering. In the future, we will design a intelligent method to reduce complexity of learning of clustering parameters. We through comprehensive test to evaluate our method, and prove it can get a more reasonable clustering result.

Acknowledgments. This work was financially supported by the national high technology research and development program (863 Program) IOT in agriculture and food quality and safety control system research foundation (2011AA100702).

References

1. Huang, J.Z., Ng, M.K., Rong, H., et al.: Automated variable weighting in K-means type clustering. *IEEE Transaction on Pattern Analysis and Machine Intelligence* 27(5), 657–668 (2005)
2. Dhillon, I.S., Guan, Y., Kogan, J.: Refining Clusters in High Dimensional Text Data. In: Price of the 2nd SIAM Workshop on Clustering High Dimensional Data, Arlington, USA, pp. 59–66 (2002)
3. Zhang, B.: Generalized K-Harmonic Means: Denamic Weighting of Data in Unsupervised learning. In: Proceedings of 1st SIAM International Conference on Data Mining, Chicago, USA, pp. 1–13 (2001)
4. Han, J., Kamber, M., Pei, J.: *Data Mining: Concepts and Techniques*, 3rd edn. MK (2011)
5. Shi, Z.: *Knowledge discovery*, 2nd edn. Tsinghua University Press, Beijing (2011)
6. Lucas, A., Vlassis, M., Verbeek, J.: The global K-means clustering algorithm. *Pattern Recognition* 36(2), 451–461 (2003)

7. Park, H.S., Jun, C.H.: A simple and fast algorithm for K-medoids clustering. *Expert Systems with Applications* 36(2), 3336–3341 (2009)
8. Ester, M., Kriegel, H., Sander, J., et al.: A Density-Based Algorithm for Discovering clustering Clusters in large spatial database with Noise. In: Proceedings of the 2nd International Conference, on Knowledge Discovery and Data Mining, Portland, USA, pp. 226–231 (1996)
9. Birgin, E.G., Matinez, J.M., RonConi, D.P.: Minimization subproblems and Heuristics for an Applied Clustering Problem. *European Journal of Operational Research* 146(1), 19–34 (2003)

Improving Short Text Classification Using Public Search Engines^{*}

Wang Meng, Lin Lanfen, Wang Jing, Yu Penghua, Liu Jiaolong, and Xie Fei

College of Computer Science and Technology, Zhejiang University, Hangzhou, China
{wangmeng, 11f, cswangjing, 11121053, jl_liu,
skyfire_xiefei}@zju.edu.cn

Abstract. In Web2.0 applications, lots of the texts provided by users are as short as 3 to 10 words. A good classification against the short texts can help the readers find needed messages more quickly. In this paper, we proposed a method to expand the short texts with the help of public search engines through two steps. First we searched the short text in a public search engine and crawled the result pages. Secondly we regarded the texts in result pages as some background knowledge of the original short text, and extracted the feature vector from them. Therefore we can choose a proper number of the result pages to obtain enough corpuses for feature vector extraction to solve the data sparseness problem. We conducted some experiments under different situations and the empirical results indicated that this enriched representation of short texts can substantially improve the classification effects.

Keywords: short text, classification, feature expansion.

1 Introduction

Classification of short texts presents a new challenge to NLP and IR researchers, and the widely used microblogging services (e.g., Twitter) make this work more and more important. The limited length of short documents results in less common terms among different short texts. Directly applying conventional text classification methods often produces inadequate results. For example, the tweets in Twitter or the questions in automatic Q&A systems (e.g., Yahoo! Answers system) are so short that the feature extraction from the original text segments is extremely hard.

To solve this problem, we employed the documents on the web to obtain greater contexts for a short text. Inspired by the pseudo-relevance feedback (Xu Y et al[13]) algorithm used in query expansion, we leveraged the strength of public search engines in expanding the short text. In detail, we submitted the short text as a query to the public search engine, and extracted the abstracts from the result pages as an expansion

* This work is supported by the Doctoral Program of Higher Education of China (No.2011010111006S), National Science and Technology Support Program of China (No.2011BAGOSB04, No.2012BAD3SB01), and Zhejiang Key Science and Technology Innovation Team Plan of China (No. 2009RSOOIS).

of the original short text. The expansion is long enough to be processed using the conventional classification methods. Furthermore, based on the pseudo-relevance feedback theory, the results returned by the search engines are relevant to the original query, and in our case the query is the short text to be classified. We analyzed our approaches using a dataset from the Yahoo! Answers system. The experiments showed that the expansion can greatly improve the results of the short text classification both in precision and recall.

The remainder of this paper is organized as follows. First we briefly review previous works related to our research in section 2. We then describe the overall framework of our method in section 3 and propose the details of the short text expansion method in section 4. In section 5, we analyze the results of our method using a dataset from the Yahoo! Answers system. We summarize this paper in section 6.

2 Related Work

Before the microblogging services (e.g., “Twitter”) became popular, short text classification was probably not a research hotspot, but there were some useful works in related fields such as the query similarity computation. Mehran Sahami et al[2] and Bollegala et al[3] measured the semantic relatedness between text snippets by using search engines (e.g., Google). Furthermore, Wen-tau Yih and Christopher Meek[4] improved this method using a better keyword extractor[5] instead of the traditional TFIDF score or its variations.

For the problem of short text classification itself, Sarah Zelikovitz and Haym Hirsh[7] described a method using the labeled training data and the unlabeled but related longer documents together. Steven M. Beitzel et al[1] proposed a framework for automatic web query classification that combined three existing methods (exact matching, supervised learning and rule-based classifier produced by mining from unlabeled queries) together. Their work outperformed each of the component approaches, and achieved high recall in particular. Kwanho Kim et al [10] proposed a kernel which extended the feature vector of each document by incorporating syntactic and semantic information using three levels (word similarity, document similarity, category similarity) of semantic annotations. Qiang Pu and Guo Wei Yang[8] used the topic model methods in the classifying process. They combined ICA and LSA together and used LSA as a data preprocessing method. Xuan-Hieu Phan et al[9] expanded the short texts with a rich set of hidden topics mining from a large-scale external data collection called “universal dataset” with LDA(which was another kind of topic model methods). Miles Efron et al[11]’s chief contribution was a novel approach to representing collections of short texts. Their method modeled not only lexical properties, but also temporal properties of documents. Bharath Sriram et al [12] attempted to use a small set of domain-specific features extracted from the tweet as well as the author’s profile to expand the twitter content for text classification.

Our approach is different from the works above in two ways: 1) we use an online framework while most of the previous works is in offline mode, which makes our

method is able to deal with the future items as well as the test ones in this paper. 2) In order to expand the short text, we use a search engine to generate a corpus instead of using the static corpus for all short text items. In other words, our expansion is based on the knowledge on the whole web, which makes it capable of adapting to different areas.

3 The General Framework

In this section, we present the proposed framework which consists of the following three steps depicted in Fig. 1.

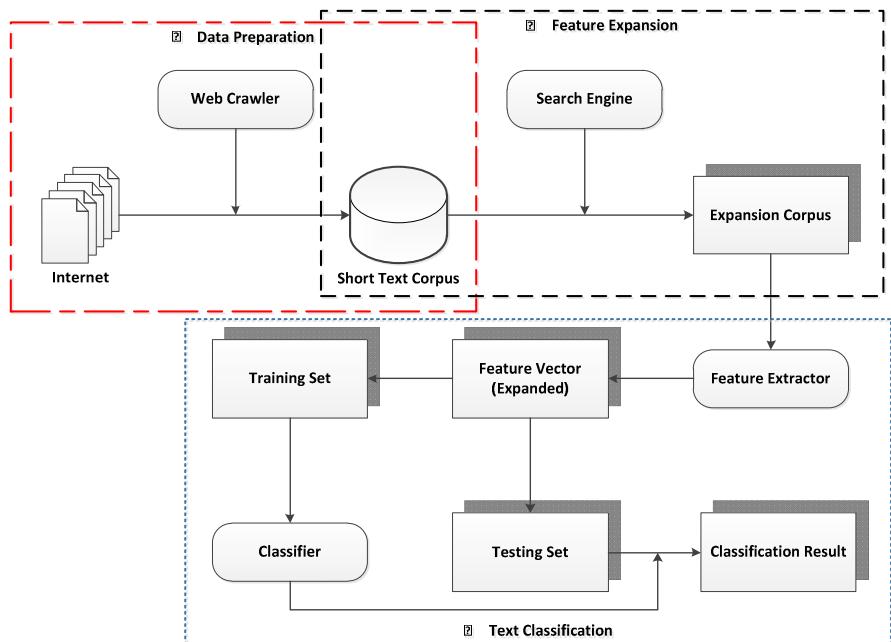


Fig. 1. The framework of short text classification with search engine expansion

The first step is preparing the short text corpus. The main problem in this part is that most of the public short text resources (e.g., Twitter) on the web are not labeled or classified very well (which indicates the necessity of short text classification), which means that the researchers have to classify the texts in training set and testing set manually. In order to reduce the work in data preparing period, we chose the well-known Answering System--Yahoo! Answers as the object of our web crawler. As shown in Fig. 2, all questions (which are typical cases of short texts) have been classified into 26 categories by the website, and we can crawl the pages category by category to obtain the classified questions.

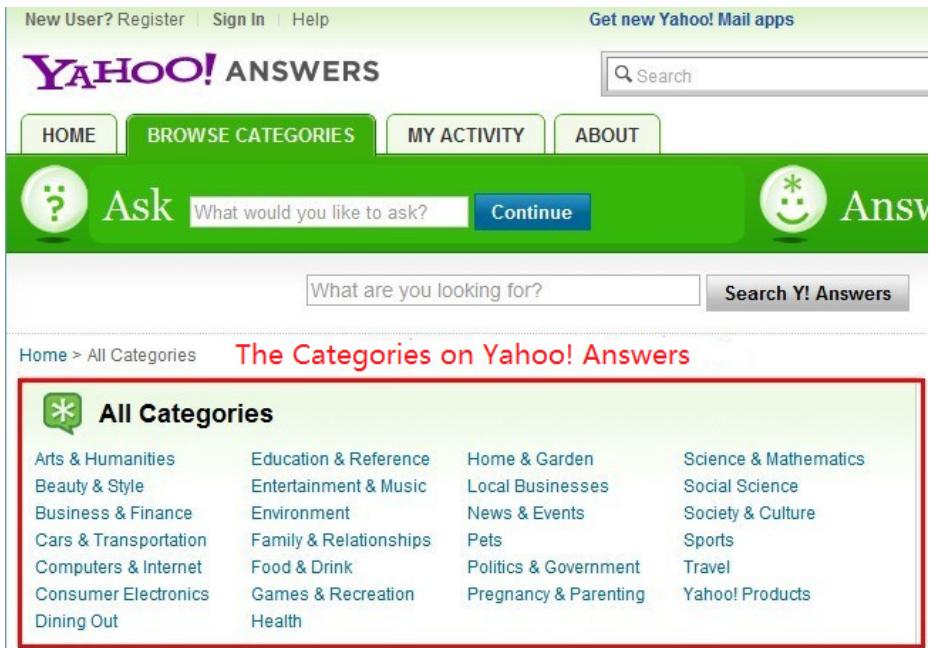


Fig. 2. “All Categories” Pages of Yahoo! Answers

Among the three steps, feature expansion process is probably the most important one. More and more researchers[6][7][9][11][12] have realized that the words in the short text itself are not enough to represent the semantics of the text, and a bigger and related corpus of each short text item is necessary for feature extraction. For this purpose, we consider the short text as a query and submit it to a search engine. Obviously the result pages returned by the search engine are related to the original short text. In addition, the result pages returned by the search engine are long enough for traditional feature extraction methods such as TFIDF. The details will be discussed in Section 4.

With the expanded corpus, a short text item can be represented by a feature vector extracted from the bigger corpus in which the texts are one-to-one mapped to the original short texts. Then the feature vectors are divided into two groups: the training set and the testing set. Therefore the automatic classification algorithms such as Naïve Bayes and KNN can be applied to the sets. More information about the classification methods and experimental results can be found in Section 5.

4 Short Text Expansion

After crawling large amounts of questions from Yahoo! Answers, we construct the feature vector for each question item. Most of the questions are so short that they only

contain less than 10 words (some examples of the questions are shown in Table 1), and if we extract the feature terms directly from the short text using the traditional methods, the number of common terms among different feature vectors will be so few that it will be hard to classify them using vector space model.

In order to solve this problem, we employ a new short text expansion method based on the pseudo-relevance feedback theory. Let Q_t be a question in Yahoo! Answers system consisting of $|Q_t|$ word tokens $\{q_1, q_2, q_3 \dots q_n\}$ and $n=|Q_t|$. Also let S be a corpus of $|S|$ questions, obviously $S = \{Q_1, Q_2, Q_3 \dots Q_m\}$ and $m=|S|$. For each question, a category C is assigned and we assume that each question has only one category. We can get the expansion of a short question through the following two steps.

Step 1: Query Searching. For a question Q_t ($t \in [1, m]$), we search it as a query in a public search engine and crawl the top k result pages R_t returned by the search engine. We define the page i in R_t as P_i , so apparently $R_t = \{P_1, P_2, P_3 \dots P_k\}$.

Each of the result pages of question Q_t is full with items related to Q_t . Usually each item contains two main parts: a hyperlinks L pointing to the webpage relevant to Q_t and an abstract A of that webpage. So in fact the page P_i in R_t can be represented as $P_i = \{(L_{i1}, A_{i1}), (L_{i2}, A_{i2}), (L_{i3}, A_{i3}) \dots (L_{ih}, A_{ih})\}$ and h is the number of the items contained in page P_i . In general, the hyperlink L is the title of the webpage which is a short text as well, thus we only keep the abstract part of each item in result pages.

Step 2: Abstract Extraction. For each item in P_i , we extract the abstract part A_{ij} of it and combine them together. The combined abstracts set is not only much longer than the original question but also related to it. In theory, it is a good expansion of the original short question. We denote the combined abstracts set of Q_t as E_t and $E_t = \{(A_{11}, A_{12}, A_{13} \dots A_{1h}), (A_{21}, A_{22}, A_{23} \dots A_{2h}), (A_{31}, A_{32}, A_{33} \dots A_{3h}) \dots (A_{k1}, A_{k2}, A_{k3} \dots A_{kh})\}$ where A_{ij} means the No. j item's abstract of page i in the search result set R_t of question Q_t .

We can apply most of the text classification techniques on the longer representation E_t which has similar semantic meanings with Q_t . Details of the classification methods and their experimental results will be discussed in the next section.

Table 1. Some Examples of Questions

Question	Category
Looking for a nascar store in canada?	Auto Racing questions
What does "strike out the side" mean?	Baseball questions
How long is an nba 3 point shot?	Basketball questions
What is the speed of a boxers jab?	Boxing questions
Do hot golf balls go further than cold golf balls?	Golf questions
Where is a swim suit place with cheap prices?	Swimming & Diving questions
Is it better to bike before lunch or after lunch?	Cycling questions

5 Experimental Results

5.1 Experimental Setup

We downloaded some questions of the sports category in the English version of Yahoo! Answers. We got 24 subcategories under sports topic and 2396 questions (approximately 100 questions per subcategory) in total. The number of the words included by each subcategory is shown in Table 2.

Table 2. Details of the experimental data

No.	Category	Number of words	total	Number of question	words per
1	Basketball questions	1013		10.13	
2	Boxing questions	1077		10.77	
3	Cricket questions	1024		10.24	
4	Cycling questions	1031		10.31	
5	Handball questions	887		8.87	
6	Water Sports questions	1109		11.09	
7	Martial Arts questions	1022		10.22	
8	Cricket questions	1185		11.85	
9	Hockey questions	1152		11.64	
10	Swimming&Diving questions	999		9.99	
11	Tennis questions	1024		10.24	
12	Motorcycle Racing questions	1031		10.31	
13	Fantasy Sports questions	1218		12.18	
14	Horse Racing questions	1006		10.27	
15	Football(American) questions	962		9.62	
16	Wrestling questions	983		9.83	
17	Golf questions	1090		10.90	
18	Surfing questions	928		9.28	
19	Running questions	978		9.78	
20	Rugby questions	1086		10.86	
21	Snooker & Pool questions	1224		12.24	
22	Football(Canadian) questions	1000		10.00	
23	Volleyball questions	1027		10.37	
24	Other - Sports questions	1007		10.07	

In order to get a more universal result of the classification methods, we randomly divided the dataset into 10 equal partitions and performed a 10-fold cross validation. We also employed three different classification methods on the dataset above to

evaluate our expansion method in different situations. Furthermore, the public search engine we used here is the Bing search created by Microsoft.

5.2 Performance Evaluation

Experiments Using Different Classification Algorithms with Expanded Corpus. The three methods we used here was the Rocchio, Naïve Bayes and KNN. In Rocchio algorithm, each question was represented by a feature vector which was extracted from the expansion corpus using TFIDF. And the cosine coefficient between the test question's feature vector and the category's feature vector was computed to determine the similarity of them. The category's feature vector was defined by the average of the feature vectors belonging to this category in training dataset.

Table 3 shows the 10-fold cross validation results of Rocchio method, and we got 75.3% precision and 75.9% recall on average.

Table 3. The results of the Rocchio algorithm

FOLD	Precision	Recall	F-score
0	0.78	0.77	0.76
1	0.7	0.73	0.71
2	0.77	0.77	0.75
3	0.79	0.79	0.78
4	0.72	0.73	0.7
5	0.75	0.76	0.75
6	0.78	0.77	0.76
7	0.74	0.75	0.72
8	0.74	0.76	0.74
9	0.76	0.76	0.74
Average	0.753	0.759	0.741

The second classification method we employed here was the Naïve Bayes. Table 4 shows the details of this method. In this situation, the precision and the recall were 74.8% and 74.4%.

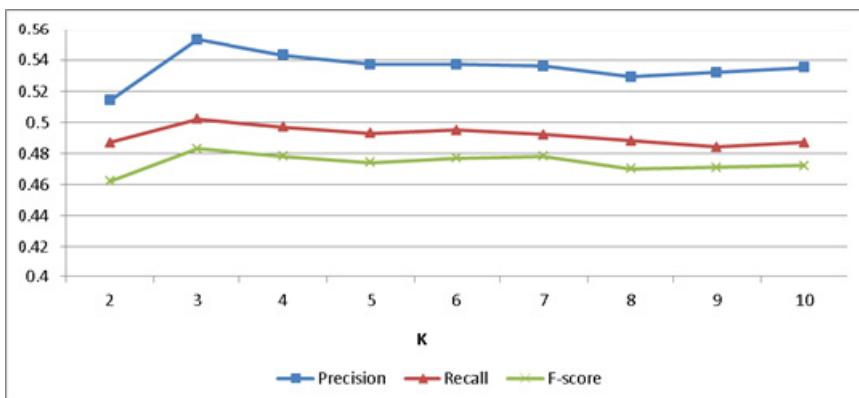
Table 4. The results of the Naive Bayes algorithm

FOLD	Precision	Recall	F-score
0	0.78	0.76	0.75
1	0.72	0.74	0.71
2	0.75	0.74	0.72
3	0.77	0.75	0.74
4	0.7	0.69	0.67

Table 4. (Continued)

5	0.77	0.77	0.76
6	0.76	0.76	0.75
7	0.73	0.72	0.7
8	0.75	0.74	0.73
9	0.75	0.77	0.73
Average	0.748	0.744	0.726

The last algorithm we used was the KNN classification algorithm. As its name implies, a feature vector's category is determined by the k-nearest neighbors' categories in the vector space. So we have to find a proper k to get the best result. We performed the classification process with different k on the expansion corpus, and drawn curves in Fig 3. Obviously with k equaled 3, we can obtain 55.3% in precision as well as 50.2% in recall, and this was the best result we can achieve with KNN algorithm on our dataset.

**Fig. 3.** The results of KNN algorithm with different k

Experiments on the original corpus and the expanded corpus with different classification algorithms. In this part, we applied the three algorithms mentioned above both on the original short text corpus and the expanded corpus which had been processed with our method. For each of the three algorithms, we performed a 10-fold cross validation with both corpuses.

As shown in Fig 4, our expanded method increased the precision, recall and F-score for all the three methods. For Rocchio algorithm, the expansion process helped us increase the F-score by 11.2%. For Naïve Bayes, the enhancement was 22%. And for KNN, we got a 32.2% improvement in F-score. Furthermore, among all situations, the Rocchio method on the expanded corpus achieved the best result with 75.3% in precision, 75.9% in recall and 74.1% in F-score.

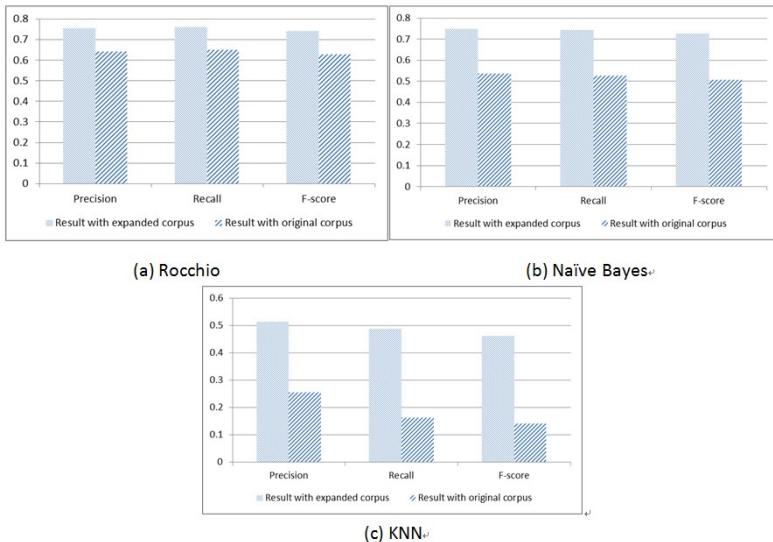


Fig. 4. The results of different classification algorithms between the original corpus and the expanded corpus

6 Conclusion

We have proposed a novel approach to classify the short texts. This approach expanded the short texts based on the pseudo-relevance feedback theory with the help of public search engines. So we can map the original short text corpus to a larger expanding corpus on which the traditional classification algorithms can get acceptable results. The experiments with Yahoo! Answers dataset show that the classification processes with our expanding method achieved better results than the method without it. There are several potential areas for future work, including the extraction of the relationships among short texts, the semantic comprehension of short texts, and automatic classification on particularly short texts (2-5 words length) or even on single words.

References

- Beitzel, S.M., Jensen, E.C., Frieder, O., Grossman, D., Lewis, D.D., Chowdhury, A., Kolcz, A.: Automatic Web Query Classification Using Labeled and Unlabeled Training Data. In: 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Salvador, Brazil, pp. 581–582 (2005)
- Sahami, M., Heilman, T.D.: A Web-Based Kernel Function for Measuring the Similarity of Short Text Snippets. In: 15th International Conference on World Wide Web, Edinburgh, Scotland, pp. 377–386 (2006)
- Bollegala, D., Matsuo, Y., Ishizuka, M.: Measuring Semantic Similarity Between Words Using Web Search Engines. In: 16th International Conference on World Wide Web, Banff, Canada, vol. 7, pp. 757–786 (2007)

4. Yih, W.T., Meek, C.: Improving Similarity Measures for Short Segments of Text. In: 22nd National Conference on Artificial Intelligence, Vancouver, Canada, pp. 1489–1494 (2007)
5. Yih, W.T., Goodman, J., Carvalho, V.R.: Finding Advertising Keywords on Web Pages. In: 15th International Conference on World Wide Web, Edinburgh, Scotland, pp. 213–222 (2006)
6. Banerjee, S., Ramanathan, K., Gupta, A.: Clustering Short Texts Using Wikipedia. In: 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Amsterdam, The Netherlands, pp. 787–788 (2007)
7. Zelikovitz, S., Hirsh, H.: Improving Short Text Classification Using Unlabeled Background Knowledge to Assess Document Similarity. In: 17th International Conference on Machine Learning, Stanford University, USA, pp. 1191–1198 (2000)
8. Qiang, P., Guo, W.Y.: Short-Text Classification Based on ICA and LSA. In: Wang, J., Yi, Z., Zurada, J.M., Lu, B.-L., Yin, H. (eds.) ISNN 2006. LNCS, vol. 3972, pp. 265–270. Springer, Heidelberg (2006)
9. Phan, X.H., Nguyen, L.M., Horiguchi, S.: Learning to Classify Short and Sparse Text & Web with Hidden Topics from Large-Scale Data Collections. In: 17th International Conference on World Wide Web, Beijing, China, pp. 91–100 (2008)
10. Kim, K., Chung, B.S., Choi, Y.R., Park, J.: Semantic Pattern Tree Kernels for Short-Text Classification. In: IEEE 9th International Conference on Dependable, Autonomic and Secure Computing, Sydney, Australia, pp. 1250–1252 (2011)
11. Efron, M., Organisciak, P., Fenlon, K.: Improving Retrieval of Short Texts Through Document Expansion. In: 35th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Portland, United States, pp. 911–920 (2012)
12. Sriram, B., Fuhr, D., Demir, E., Ferhatosmanoglu, H., Demirbas, M.: Short Text Classification in Twitter to Improve Information Filtering. In: 33rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Geneva, Switzerland, pp. 841–842 (2010)
13. Xu, Y., Jones, G.J.F., Wang, B.: Query dependent pseudo-relevance feedback based on wikipedia. In: 32nd ACM SIGIR Conference on Research and Development in Information Retrieval, Boston, MA, United States, pp. 59–66 (2009)

Using Social Networks for Integrating a Tacit Knowledge Repository to Support the Performance of Software Development Teams

Reyes Juárez-Ramírez^{*}, Rafael Pimienta-Romo, and Violeta Ocegueda-Miramontes

School of Chemical Sciences and Engineering,
Autonomous University of Baja California. Calzada Universidad 14418,
Parque Industrial Internacional Tijuana, C.P. 22390, Tijuana, B.C. México
{reyesjua,aaron.pimienta,violeta.ocegueda}@uabc.edu.mx

Abstract. Tacit knowledge provides an important support to enterprise activities; software development is not an exception. Traditionally, tacit knowledge is communicated face-to-face while performing activities; in distributed environments it is difficult to carry out. On the other hand, documenting tacit knowledge becomes a difficult task due to its informal nature. Several efforts have been made in order to document tacit knowledge, but some challenges remain. Nowadays, social media provide possibilities for communicating knowledge and are a substantive source of tacit knowledge generation. In this paper we present a case study about how we are using Facebook to supervise the project performance and for integrating a tacit knowledge repository. This repository will allow us to assess future projects at the level of team performance, knowledge sharing, and project success factors. Based on the characterization of tacit knowledge sharing through Facebook, we present an initial proposal to use uncertainty reasoning to predict the project success and for classifying the messages that express tacit knowledge.

Keywords: Social media, software development, coordination, communication, tacit knowledge, uncertainty reasoning.

1 Introduction

Software development is conceived of as a team activity, that is, people working together during a period of time generating a product. Software development is commonly seen as a project. The knowledge generated is an important element during a project execution [1]. In order to manage a project, several elements should be considered. Whitehead *et al* [2] emphasize the following elements: *communication*, *coordination*, experience, tools, teams, *roles*, *language*, working time zone, and culture. Additionally, project management involves the following specific activities [3]: *task assignment*, *task scheduling*, *expressing technical problems*, and *giving*

^{*} Corresponding author.

possible solutions. Sometimes, performing those activities is complicated in face-to-face environments; this complication is more significant in distributed environments. Some works [3, 4] emphasize some factors that can affect the distributed software development, such as: *communication style, communication media, roles* and responsibilities assignment, *coordination*, time zone, *cooperation, response time, resolution time*, team experience, *knowledge sharing*, culture, and technical support.

Two of the most important challenges in project management are *communication* and *coordination* [5, 6], but recently knowledge sharing has been growing in importance [3]. Communication is the basis of the effectiveness in project management; it is necessary to identify specific problems faced in this context. Table 1 shows some of the more common communication (Pb-Cm-00) and coordination (Pb-Cr-00) problems.

Table 1. Communication/coordination problems to be solved

Problem ID	Description
Pb-Cm-01	Problems in information distribution among team members.
Pb-Cm-02	Lack of an administration of the knowledge and documentation generated.
Pb-Cm-03	Difficulty of prominent access and contextual of information.
Pb-Cm-04	Slowness and waste of time in the collection of information.
Pb-Cm-05	Difficulty to <i>share the knowledge for the coproduction of the same one.</i>
Pb-Cr-01	Inefficient coordination within the team to resolve the <i>problems that arise in the administrative and technical processes.</i>
Pb-Cr-02	<i>Non-existence of synchronization</i> between the times and the activities that are carried out.
Pb-Cr-03	<i>Deficiencies in the structures of the project's execution.</i>
Pb-Cr-04	Difficulty in the <i>monitoring and control of the advances</i> of the projects.
Pb-Cr-05	Difficulty in the <i>monitoring and control of the activities to carry out.</i>
Pb-Cr-06	Increment in the costs in activities of supporting and monitoring projects.

Several proposals have been introduced in the context of solving these problems in distributed environments; those proposals vary from models integrating frameworks that enable both communication and coordination to the ones that use integrated methodologies such as SCRUM. However, some problems still remain. Despite the proliferation of those solutions, traditional knowledge sharing mechanisms and information technologies have been found to be unsuccessful in facilitating tacit knowledge sharing among team members and professionals [7]. We need to emphasize that this problem is derived from the nature of tacit knowledge [18], which is defined as [19]: *the knowledge that is not codified and is found in experiences, beliefs, skills, know-how, etc.*

A possible solution might be the use of online environments where team members can stay in touch and the communication can flow as needed. Past work [8, 10] and contemporary research [9, 11] support this proposal. Marwick [8] argues that tools and systems such as online discussion forums, chat rooms, and other real-time online interactions can effectively facilitate tacit knowledge sharing among team members.

Lai [10] has also confirmed the possibility of tacit knowledge transferring through Internet discussion and chat sessions. Wahlroos [11] observed that the emerging social media represents a significant potential in enhancing tacit knowledge sharing by providing live conversations, relationship networking, and collaboration among individuals.

Despite this evidence demonstrating a heavy use of social media platforms, there is a lack of understanding of how to maximize the benefits, and specifically those related to maximizing tacit knowledge sharing in these environments [7]. In recent years Facebook was considered an enemy of efficient working environment and several companies prohibited the use of Facebook during work hours; however, research such as [1, 6] has considered this not a liability but an opportunity.

By becoming a highly popular communication and social networking platform Facebook can be a prominent choice for knowledge management [1]. The nature of Facebook supports the *articulation* of knowledge. In this case, as is stated in [14], articulation can be taken to denote the process of expressing tacit knowledge into some socially shared code or symbolic representation –in its simplest form, *ordinary natural language*.

In this paper we present preliminary results of using Facebook as a tool to support the execution of software projects. We show how communication, coordination and knowledge sharing can be implemented properly in this environment. Moreover, we present a proposal for categorizing the information and knowledge generated in this distributed environment. Finally we present a set of concepts needed to adopt approximate reasoning for classifying messages and predicting project success.

This paper is organized as follows. Section 2 presents some related work. Section 3 presents a proposal for information and knowledge categorization. Section 4 describes a case study using Facebook as a tool to support software projects developed for undergraduate computer engineering students, generating statistics of the interactions. Section 5 describes a proposal to integrate a tacit knowledge repository. Section 6 presents a proposal to introduce approximate reasoning in order to classify messages as tacit knowledge, and to assess and predict some aspects of the project, such as the efficiency in coordination and communication, and the project success. Finally, section 7 presents conclusions and future work.

2 Related Works

In this section we describe some of the most closely related work, but in section 6 other related research on approximate reasoning are described.

Tacit Knowledge Sharing. Panahi *et al* [7] proposed that tacit knowledge sharing amongst physicians is known to have a significant impact on the quality of medical diagnoses and decisions. They presented a study (based on questionnaires and interviews) of what kind of social media tools physicians use, finding that the most used are: *blogs*, *micro-blogs* (Twitter), *social networks* (Facebook, Google+), and *multi-media sharing sites* (YouTube, Vimeo). Also, they found that the types of tacit

knowledge shared by physicians are: *clinical questions and answers, day-to-day clinical experiences, clinical tips, writing about clinical cases, clinical expert opinions, discussing clinical issues, problem solving, and individual commentary.*

Message Classification. In [6], a proposal to classify marketing messages is presented. They proposed two categories for messages: *marketing* and *communication*. In order to differentiate both messages, they proposed that a *marketing message* is defined as a *one-way* and often *persuasive* message of selling or promoting a service, a product, or a brand to Facebook users; while a *communication message* is defined as a *one- or two-way* message without direct sale or promotion information. They use the number of “*likes*” responses and the number of comments to measure and compare the popularity of marketing and communication messages. They used the Support Vector Machines (SVM) technique to automatically separate these two kinds of messages.

In addition, in [24] a Web-based survey and interviews are used to collect project data about requirements, sponsors and customers. Tools such as association, neural networks, clustering, Naive Bayes and decision trees are used to discover common characteristics and rules that govern project success and failure.

3 A Proposal for Information and Knowledge Categorization

Guided by the classification shown in [6], we are proposing to classify information and knowledge in three main groups, according to project management concerns. This classification is focused on the three main aspects we are interested on (See Tables 2 and 3): *coordination, communication, and knowledge sharing*. This classification is proposed based on the experience of executing a set of software projects during the last academic year using Facebook as a tool to manage and monitor those projects. Table 3 shows expressions of tacit knowledge.

Table 2. Classification for coordination/communication

Category	Description
Task assignment	Distributing a task to a team member.
Task schedule	Establishing calendar dates for a task.
Task reminder	Reminding of task deadlines.
Report task advance	Reporting the advance level for a task.
Call a meeting	Calling team members for a meeting.
Notice	Informing a notice about the project to team members.
Acceptance	Notifying of the acceptance of a task assignment, a meeting, etc.
Affirmation	Confirming a notice, request, answer, etc.
Negativity (negation)	Notifying the negation of acceptance of a task assignment, a meeting, etc.
To agree	Expressing agreement with a task, a proposal, a solution proposal.
Notify information	Notifying information about the project, scheduling, etc.
Consultation	Consulting information, technical advising, etc.
Guidance	Providing a guidance to do a task, to solve a problem, etc.

Table 2. (*Continued*)

Doubts	Expressing a doubt about task assignment, ways to do things, etc.
Motivation	Expressing encouraging words or messages to team members.
Documents	Indicating the creation or disposal of documents.
Request documents	Requesting a document.
Request information	Requesting information.
Request work style	<i>Requesting how to work</i>

Table 3. Classification for tacit knowledge sharing

Concept	Description
Notify problems	The fact of notifying problems (technical, team work).
Technical question	Asking for technical aspects or facts.
Proposals	Expressing a proposal to do things, to solve problems, etc.
Guidance	Providing guidance (technical aspects, ways to do tasks) to team members.
Solutions	Expressing/proposing solutions for a technical aspect, ways to do things.
Project management question	Expressing a question about project management issues.
Doubts	Expressing a doubt about technical aspects of the project.
Technical advice	Providing advice about technical issues.
Notify information	Notifying information about the project.
Request authorization	Asking authorization about technical or administrative aspects of the project.
Grant authorization	Giving authorization about technical or administrative aspects of the project.
Decline authorization	Denying authorization about technical or administrative aspects of the project.
Suggesting work style	Suggesting work style to the team members.

In the last academic year we have observed the performance of different roles in software development using Facebook. We have identified that some kinds of messages are generated mainly by specific roles. Table 4 shows these findings.

Table 4. Message generation by specific roles

Role	Main messages generated
Team leader	Task assignment, task schedule, task remainder, guidance, technical advance, grant authorization, suggesting work style, call a meeting, motivation.
Project manager	Call a meeting, grant authorization, decline authorization, suggesting work style, motivation.
Analyst	Acceptance, agreement, affirmation, consultation, doubts, information request, document requests, task advance report.

Table 4. (Continued)

Programmer	Consultation, problem notifications, proposals, solutions, technical advice. (Plus the same as the analyst)
Designer	(The same as the analyst)
Documenter	(The same as the analyst)

4 Case Study: Statistics of Information and Knowledge Sharing Using Facebook

During the last academic year, we adopted Facebook as a tool to support 18 software projects developed by students of a computer engineering undergraduate program, in the course of Software Engineering. Below we describe the conditions of this experiment. Students work mainly in a distributed environment, having only some face-to-face work sessions. Next we describe the methodology used.

Team integration: Each team was integrated by proximately four to six students. Each student assumed a role based on his abilities and skills. We defined six main roles: *team leader, project manager, analyst, programmer, designer, and documenter*.

Software developing model: The students, with the guidance of supervisors, decided the developing model, choosing between waterfall, agile (SCRUM) and a combination of both.

Facebook group: Each team created a private Facebook group for their project, enrolling all the members and supervisors (professor and technical staff). The group was used for publishing the project plan, the project progress, and information and knowledge generated during the project execution.

Project supervision: The professors (supervisors) monitored the group activity to review the project progress and see the problems presented.

Support tools: Besides Facebook group each team used other public systems to store documents, such as Dropbox. When students uploaded documents to Dropbox they notified to the Facebook group.

In the first academic semester the projects were scheduled from February to May, while in the second semester from August to November. Figure 1 shows the occurrence of communication and coordination comments in the first semester (August –AGO, September –SEP, October –OCT, and November –NOV).

As we can see, the majority of the comments are of *acceptance, information notification, notice, affirmation, and negation*. It is important to emphasize that by the moment the accounting of messages was performed manually due that we don't have an automatic tool. This allowed us to characterize the different types of *posts* and *comments*.

Figure 2 shows the *posts* expressing knowledge sharing. As we can see, *guidance*, and *technical advice* are the most frequent *posts*. Also in this case, September and October are the months that present more *posts* of these types.

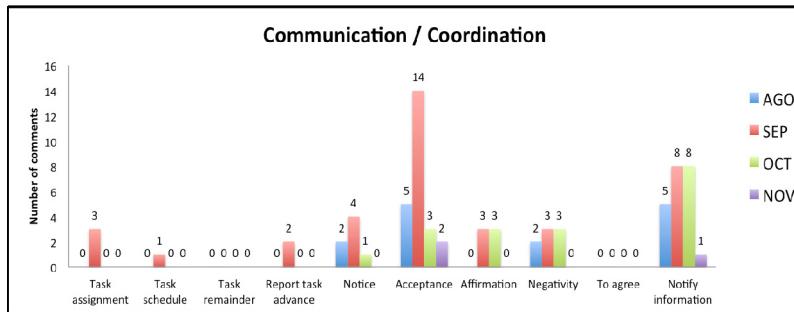


Fig. 1. Comments counting for communication/coordination

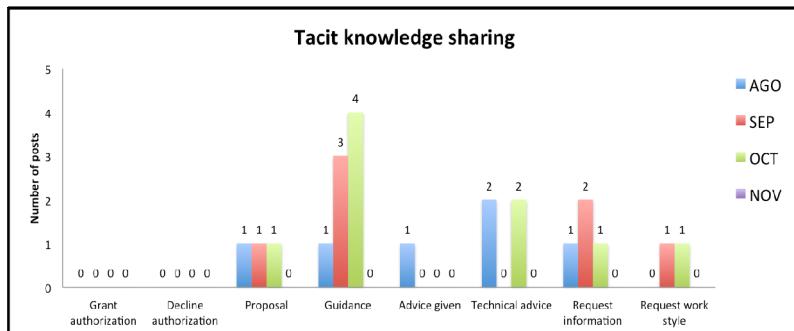


Fig. 2. Posts for tacit knowledge sharing

Based in the statistics for 18 projects, some findings are that there were fewer *comments* for coordination tasks, while there were more *comments* for knowledge sharing. In the case of *posts*, the opposite was observed, that is, there were more *posts* for coordination tasks than for knowledge sharing.

5 Integrating the Knowledge Repository

As we are interested in using Facebook as a tool to increase communication, coordination and knowledge sharing, we need to identify the information flow in both directions, from the sender to the receiver and vice versa. We have designed a database structure to insert the classified information and knowledge extracted from the Facebook database. We have selected some fields from the *post* and *comment* structures as we show in Tables 5 and 6, respectively. This information is extracted using the Graph API for Facebook.

Table 5. Fields for a *Post*

Name	Description
<i>id</i>	The post ID.
<i>from</i>	Information about the user who posted the message.

Table 5. (*Continued*)

<i>to</i>	Profiles mentioned or targeted in this post.
<i>message</i>	The message.
<i>message_tags</i>	Objects tagged in the message (Users, Pages, etc.).
<i>likes</i>	Number of likes for this post.
<i>place</i>	Location associated with a Post, if any.
<i>with_tags</i>	Objects (Users, Pages, etc.) tagged as being with the publisher of the post ("Who are you with?" on Facebook).
<i>comments</i>	Comments for this post.
<i>created_time</i>	The time the post was initially published.
<i>updated_time</i>	The time of the last comment on this post.

The “*likes*” field gives us some kind of evidence that team members (as receivers) read the message, which is a good evidence of communication.

Table 6. Fields for a *comment*

Name	Description
<i>id</i>	The comment ID.
<i>from</i>	The user that created the comment.
<i>message</i>	The comment text.
<i>created_time</i>	The time the comment was initially published.
<i>likes</i>	The number of times this comment was liked.
<i>user_likes</i>	This field is returned only if the authenticated user likes this comment.
<i>with_tags</i>	Objects (Users, Pages, etc) tagged as being with the publisher of the post ("Who are you with?" on Facebook).
<i>comments</i>	Comments for this.
<i>created_time</i>	The time the post was initially published.
<i>updated_time</i>	The time of the last comment on this.

In order to create a structure to store messages, we introduced a field called “*type*”, which help us to distinguish each kind of message taking into consideration the classification expressed in Tables 2 and 3.

6 A Proposal to Introduce Approximate Reasoning

As we can see in previous sections, activity in Facebook reveals evidence of constant interactions between team members. We have found that some kinds of interactions enable us to predict the project results. For instance, in terms of project or work success, if the number of interactions was low during a period of time, we visualized a possibility of not having a good level of progress. This situation encouraged us to reach a way to predict the project’s success based on the occurrence and frequency of some kinds of interactions. However, there exists uncertainty due that we are not sure

that all the activities and interactions are tracked in the Facebook groups. In order to formulate a proposal to predict and assess aspects of the project we need to consider some concepts of uncertainty reasoning.

Approximate Reasoning

In [20] is stated that *uncertainty* is the lack of exact knowledge that would enable us to reach a perfectly reliable conclusion. In this context, *approximate reasoning* is a process of interpretation of knowledge in a presence of uncertainty in a form of vague and contradictory knowledge, incomplete past data, uncertain new facts, not clear goals, etc. [12]. In [21] is stated that in the presence of forms of inexactness of data and knowledge, the following methodologies are suggested: (1) *probabilistic techniques*, (2) *certainty factor-based reasoning*, and (3) *fuzzy techniques*.

The certainty factors method uses rules of the following form:

IF E is true
THEN H is true $\{cf\}$
Where:
 cf represent the level of belief in hypothesis H given that evidence E .

Applying Uncertain Reasoning in Project Success Prediction

Nasir and Sahibuddin [22] proposed twenty-six critical success factors. Other authors such as Chow and Cao [23] suggested critical success factors for agile software development projects. They are classified in three categories: (1) people-related factors, (2) process-related factors, and (3) technical-related factors. Table 7 shows specific factors that we are considering for this analysis.

Table 7. Critical factors for project's success

Category	Factor
<i>People-related</i>	Good leadership (<i>per-01</i>)
	Committed and motivated team (<i>per-02</i>)
	Coherent, self-organizing teamwork (<i>per-03</i>)
<i>Process-related</i>	Effective communication and feedback (<i>pqr-01</i>)
	Effective monitoring and control (<i>pqr-02</i>)
	Strong communication focus with daily face-to-face meetings (<i>pqr-03</i>)
<i>Technical-related</i>	Supporting tools and good infrastructure (<i>thr-01</i>)

Taking into account the statistics shown in previous sections, we can focus our attention on *per-01*, *pqr-01*, and *pqr-03*. In the case of *pqr-03* face-to-face communication is equivalent to the message-based conversation in Facebook.

If we take into account *pqr-01* for effective communication, we need to consider the occurrence of messages such as those shown in Table 3. We need to emphasize that each message of a sender should have one or more responses from one or more receivers. Based on this, in the case of certainty factors, an example of how we can manage the assessment of effective communication is the following:

Rule 1:

**IF response is request information
AND response is acceptance
THEN task assignment is effective {cf 0.8}**

Rule 2:

**IF response is affirmation
AND response is report task advance
THEN request reporting progress is effective {cf 0.9}**

Based on the statistics that we gathered in the last academic year, a set of uncertain terms and their interpretation can be established, as is managed in [20]. The uncertain terms and the certainty factors can be: definitely not (-1.0), almost certainly not (-0.8), probably not (-0.6), maybe not (-0.6), unknown (-0.2 to +0.2), maybe (+0.4), probably (+0.6), almost certainly (+0.8), definitely (+1.0). This can help us to manage conclusions for rules that involve multiple antecedents and consequents because the effectiveness of communication involves a lot of variables and circumstances, which can be expressed as evidences or hypothesis in the rules.

Applying Uncertain Reasoning in Text Classification

In order to propose a classification scheme for messages, we need to consider how the text classification is implemented by different techniques, such as those shown in [13, 14, 15, 16, 17]. Some of the classification tools and techniques use the probability of occurrence of characters or full words. Most of these techniques provide good levels of efficiency for a few variables. In our situation we have many variables to be considered (type of messages, author/role, timeline of the project, etc.), which conduct us to consider other techniques such as approximate reasoning.

Next, we will establish some conditions for the application of approximate reasoning in our research.

Limitations in tagging post and comments

- Due to the nature of informal messages in Facebook, there is not a clear or explicit identification of the kind of information and knowledge contained inside a *post* and a *comment*.
- In the same way, for now there is not a categorization for tagging the messages in Facebook.
- In a *post* or a *comment*, the sender can write more than one fact, that is, in the same *post* people can write a notice, followed by an information request, followed by a task schedule, and so on.
- A “seen” notation over a message does not mean that a *post* or a *comment* has been read and analyzed. This contributes to *uncertainty*.

Fortunately we have some positive contributions of using Facebook as an information and knowledge container:

- Role of the sender: Even though we don't have exclusive types of messages for a specific role, sometimes depending on his role, the sender can send specific types of messages. This is the same situation for the receiver.
- The timeline in the project calendar: Depending on the calendar of the project, some types of messages can have prevalence, as we have shown in the case study (see section 4).
- In the case of a “*likes*” notation there is more probability that the message (*post* or *comment*) has been read and analyzed. This contributes to *certainty*.

In the case of certainty factors, an example of how we can manage the text-based messages classification is:

Rule 1:

**IF role is team leader
AND time is first week
AND message is post
THEN message is task schedule {cf 0.8}**

Rule 2:

**IF role is analyst
AND time is first week
AND message is comment
THEN message is doubt {cf 0.6}**

7 Conclusions and Future Work

In this paper we presented examples of how social media, especially Facebook, can serve as a tool for software project execution and monitoring, helping to solve common problems in distributed environment, such as communication, coordination, and knowledge sharing. The success of using Facebook for this objective rests on its popularity between young people and its broad capacity to maintain online all the stories of the everyday activities in the project execution.

Additionally, we presented a proposal for categorization of the information and knowledge generated during project development. We focused on the identification of information and knowledge that supports the coordination and communication during the project. Also, we identified aspects that represent tacit knowledge sharing.

In order to construct a knowledge repository derived from the activity in the Facebook (groups created for the project execution), we selected a set of fields from the *post* and *comment* structures that the Graph API provides. We are integrating specific fields to record the classification of messages.

Finally, we presented a set of concepts and circumstances that serve as a basis to formulate the application of approximate reasoning techniques in order to assess the effectiveness of the communication in order to predict the project success. Also we proposed use of approximate reasoning for the message classification.

We can argue that our main contribution arises from gathering evidence of how project management benefits from the use of social media, specifically from Facebook. Furthermore, we have shown that tacit knowledge can be transferred in an easy way using the nature of informal environments like Facebook.

Even though in the case study we implemented the accounting of the different types of messages manually, in the future to take advantage from the repository, the message classification will be done automatically using an intelligent technique.

References

1. Phosaard, S., Wiriayapinit, M.: Knowledge Management via Facebook: Building a Framework for Knowledge Management on a Social Network by Aligning Business, IT and Knowledge Management. In: The World Congress on Engineering 2011, WCE 2011, London, U.K, July 6-8, vol. III (2011)
2. Whitehead, J., Mistrík, I., Grundy, J., van der Hoek, A.: Collaborative Software Engineering: Concepts and Techniques. In: Mistrík, I., Grundy, J., van der Hoek, A., Whitehead, J. (eds.) Collaborative Software Engineering. Springer, Heidelberg (2010)
3. Richardson, I., Casey, C., Burton, J., McCaffery, F.: Global Software Engineering: A Software Process Approach. In: Mistrík, I., Grundy, J., van der Hoek, A., Whitehead, J. (eds.) Collaborative Software Engineering, pp. 35–54. Springer, Heidelberg (2010)
4. Nguyen, T.H.D.: Communication in Software Engineering Teams. Master thesis. University of Victoria (2007)
5. Kwok, L., Yu, B.: Towards developing a typology of hospitality companies' Facebook messages. Manuscript submitted for publication, Syracuse University (2011)
6. Yu, B., Kwok, L.: Classifying Business Marketing Messages on Facebook (2011), <http://research.microsoft.com/en-us/um/beijing/events/ia2011/n7.pdf>
7. Panahi, S., Watson, J., Partridge, H.: Potentials of Social Media for Tacit Knowledge Sharing Among Physicians: Preliminary Findings. In: 23rd Australasian Conference on Information Systems, Geelong, December 3-5 (2012)
8. Marwick, A.D.: Knowledge Management Technology. IBM Systems Journal 40(4), 814–830 (2001)
9. Panahi, S., Watson, J., Partridge, H.: Social Media and Tacit Knowledge Sharing: Developing a Conceptual Model. In: World Academy of Science, Engineering and Technology (WASET), Paris, France, pp. 1095–1102 (2012)
10. Lai, I.L.A.: Knowledge Management for Chinese Medicines: A Conceptual Model. Information Management & Computer Security 13(2/3), 244–255 (2005)
11. Wahlroos, J.K.: Social Media as a Form of Organizational Knowledge Sharing: A Case Study on Employee Participation at Wärtsilä. In: Department of Social Research, Faculty of Social Sciences, p. 100. University of Helsinki, Helsinki (2010)
12. Kasabov, N.K.: Learning fuzzy rules and approximate reasoning in fuzzy neural networks and hybrid systems. Fuzzy Sets and Systems 82, 135–149 (1996)
13. Bobicev, V.: Text Classification Using Word-Based PPM Models. Computer Science Journal of Moldova 14(2(41)) (2006)
14. Bobicev, V.: Comparison of Word-based and Letter-based Text Classification, http://lilu.fcim.utm.md/Word_letter_compres.pdf

15. Dumais, S., Platt, J., Heckermann, D., Sahami, M.: Inductive learning algorithms and representations for text categorization. In: Intl. Conf. on Info. and Knowledge Management, pp. 148–155 (1998)
16. Lewis, D.D., Ringuelette, M.: A comparison of two learning algorithms for text categorization. In: Annual Symposium on Document Analysis and Information Retrieval, pp. 37–50 (1994)
17. Learningto, T.J.: Classify Text using Support Vector Machine. Methods, Theory, and Algorithms. Kluwer Academic Publishers (May 2002)
18. Häkanson, L.: Creating Knowledge the Power and Logic of Articulation. Organization Science (2002) (unpublished)
19. Nonaka, I., Konno, N.: The concept of “ba”: Building a foundation for knowledge creation. California Management Review 40(3), 40–54 (1998)
20. Netnevitsky, M.: Artificial Intelligence, a guide to intelligent systems, 2nd edn. Addison Wesley (2005)
21. Konar, A.: Artificial Intelligence and Soft Computing, Behavioral and Cognitive Modeling of the Human Brain. CRC Press LLC, Boca Raton (2000)
22. Nasir, M.H.N., Sahibuddin, S.: Critical success factors for software projects: A comparative study. Scientific Research and Essays 6(10), 2174–2186 (2011)
23. Chow, T., Cao, D.-B.: A survey study of critical success factors in agile software projects. Journal of Systems and Software 81(6), 961–971 (2008)
24. Yousef, A.H., Gamal, A., Warda, A., Mahmoud, M.: Software Projects Success Factors Identification using Data Mining. In: The 2006 International Conference on Computer Engineering and Systems, November 5-7, pp. 447–453 (2006)

Tacit Knowledge Formalization to Support the Adoption Process of Software Quality Models

Violeta Ocegueda-Miramontes and Reyes Juárez-Ramírez

School of Chemical Sciences and Engineering,
Autonomous University of Baja California Calzada Universidad 14418,
Parque Industrial Internacional Tijuana, C.P. 22390, Tijuana, B.C. México
{violeta.ocegueda, reyesjua}@uabc.edu.mx

Abstract. Due to the key role played by tacit knowledge in the adoption process of a quality model, this paper aims to present a first approach to formalize its expression so that it can be captured and stored for later reuse. It is proposed to express tacit knowledge in the form of rules, and add a degree of belief to represent an expert judgment. The degree of belief is calculated by using the certainty factors theory to allow individual's expertise to be used in the evaluation of the rules.

Keywords: tacit knowledge, tacit knowledge formalization, uncertainty reasoning, adoption of quality models.

1 Introduction

Since the mid 60's, the software industry exhibits serious quality issues, mainly reflected in late delivery, several bugs and high maintenance costs [1]. In 1968, NATO sponsored a conference in which is recognized the state of crisis of the software industry [2]. However, it is until 80's when early quality models arise that aimed to ensure the quality of the software products by ensuring the quality of its development process [1][3][4][5].

Several studies have suggested that the adoption of a quality model generates many benefits to the organizations that are committed to this strategy, such as: improvement of their processes, project, and quality of the products; reduces their development time, costs, errors; etc. [6][7][8][9].

Unfortunately, the adoption of such models is a long, confusing and expensive process, which has led to a failure rate up to the 70% of the adopting initiatives [10][11][12][13][14][15][16].

Due to the high rate of failure that is around the adoption process of a quality model, various researchers have dedicated efforts to identify the factors affecting this process. Some of them agree that the knowledge that revolves around the adoption of a quality model plays a key role in such process [16][17][18][19][20][21][22][23][24][25]. Therefore, it is of vital importance for this knowledge to be represented in the most appropriate way so it is easily understood. However, the way that the knowledge

is presented/expressed in the model may be not the most suitable for its proper understanding.

This is because the quality models are characterized by documents that show, in rigid structures, large amounts of knowledge, which are in turn expressed in specialized vocabulary. So that, someone who is not clear with such vocabulary, would hardly understand. All this causes that the assimilation of this knowledge becomes a difficult task. That is, the transference of the knowledge that is within the quality model, to the person that wants to interpret it is hindered by the way the knowledge was represented. If the person has not understood properly, correctly, and fully the knowledge expressed in the quality model, then it cannot be expected the person to guide a group of individuals during the adoption process properly.

Something that might be useful for people who act as facilitator during the adoption process, would be to have the knowledge that the experiences in that process could provide. The knowledge inferred from the experiences could help to assimilate the knowledge expressed in the quality model.

Because of this, the main purpose of this work is to propose a first approach to formalize tacit knowledge, which aims to facilitate its capture for its future reuse.

2 Tacit Knowledge to Support the Adoption Process

As mentioned before, some researchers agree that the knowledge expressed in a quality model is not adequately transferred/interpreted so that it can be used in the adoption process [15][17][18][19][20][21][22][23][24][25]. This is mainly because most of the knowledge managed during this process is primarily tacit. The tacit knowledge is particularly significant because it can make the difference when making decisions. Thus, we believe that there is a need for a formal way to express this kind of knowledge, so that it can be easily captured and stored for future reuse. If tacit knowledge is capture and stored in a knowledge base, then when people have a problem/question they can check that knowledge base for a solution/answer so they can solve their problem/question, or at least avoid mistakes others made; and this can save a lot of money, time and effort.

Before giving an attempt to formalize tacit knowledge, it should be explained what tacit knowledge is and its characteristics.

2.1 Tacit Knowledge

To understand what tacit knowledge is, it is necessary to first define the concept of knowledge. According to Salazar [26], knowledge is a set of cognitive, confirmed, experienced and contextual beliefs, which are the basis for objective action and value generation. Knowledge can be classified into two types: tacit and explicit [27].

- Explicit knowledge: refers to the knowledge that is easily codified, this is found written in documents, handbooks, manuals, reports, books, etc.
- Tacit knowledge: refers to the knowledge that is not codified and is found in experiences, beliefs, skills, know-how, etc.

According to Nonaka and Konno [28] there are two ways in which tacit knowledge can be shared, through socialization and externalization.

- Socialization (tacit-tacit): involves the act of sharing knowledge held by individuals within an organization. That is, the knowledge is transmitted through coexistence between members of an organization, which is because tacit knowledge can only be acquired through direct interaction with your source, allowing the person to learn the skills of other individuals through observation, imitation and practice.
- Externalization (tacit-explicit): involves expressing tacit knowledge in understandable formats so others can understand it. This increases the chances for tacit knowledge to be shared.

Some researchers consider that tacit knowledge has two levels, the articulable level and the inarticulable level [29][30][31]. The main difference between these two, is that the articulable tacit knowledge is the one that “we can say we know”, and the inarticulable tacit knowledge is the one that we don’t even realize we know.

As remarked before, there are two ways to share tacit knowledge, the first one is through socialization, such as observation, imitation and practice; and the second one is through externalization, by applying formats to tacit knowledge. For the adoption of a quality model, it is fundamental to capture and store as much as possible the tacit knowledge that revolves around this process so it can be reused. Because of that, we consider crucial to find an appropriate format that helps to express tacit knowledge. But, to express tacit knowledge, it is necessary for this to be articulable so it can be then externalized.

By now, it is relevant to specify some of the forms that articulable tacit knowledge can take, such as [31]:

“Abstract high level plans..., assumptions, behavior, beliefs, business knowledge, common sense..., complex multiconditional rules, concepts, constructs..., contradiction..., decision making..., discussion, every situations..., expectations..., grammatical rules..., habits, heuristics..., information’s placed in meaningful context – eg. message, innovation, interaction, job knowledge, judgement, justified true belief, know how, knowledge base that enables us to face the everyday world, knowledge of designs, logical rules, meaning, methods, negotiation, observation, perceptions, performance, perspectives, practical know how, practice, prescriptive knowledge, principles, private knowledge, procedural in nature, procedures, process, proverbs..., ritual, routine, rules, rules of thumb, schema, script/scripted, semantics..., stories, subjectivity, task management, tasks, team coordination, technique, technology, theories, tradition, trial and error, tricks, understanding, understanding of categories, values, way things are done, wisdom”.

As can be seen, articulable tacit knowledge has many forms so it is not possible to deal with all of them at the same time and in the same way. For that reason, it is important to point out which of them will be addressed in this work. Analyzing the adoption of a quality model, it could be said that a common way to share knowledge during this process is by making questions or formulating problems and getting someone to answer/solve them. So a good way to converge the adoption process, the

externalization process and the forms of the articulable tacit knowledge, could be by capturing both the problems/questions the people involved in the process express, and the solutions/answers given to them.

The next section shows an attempt to give tacit knowledge a formal representation.

2.2 Tacit Knowledge Formalization

As mentioned above, there are different forms of articulable tacit knowledge, but in this article we are going to address those that can take the form of problem/question with its respective solution/answer, some examples of this could be: common sense, complex multiconditional rules, decision making, logical rules, know how, practical know how, principles, procedures, process, rules, rules of thumb, tricks, understanding, way things are done, etc.

Figure 1 shows one of the processes that may occur to acquire articulable tacit knowledge, which can be summarized in some simple steps:

1. It begins with the interrogation if there is a problem/question.
2. a) If the reply is no, then the process ends; b) if the reply is yes, then it is seek a possible solution/answer.
3. It is proposed a solution/answer.
4. It is implemented the solution/answer.
5. a) If the solution/answer did not work, then keep looking; b) if the solution/answer worked, then it is the end.

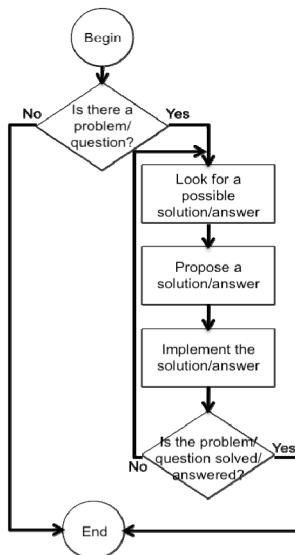


Fig. 1. A process to obtain tacit knowledge

This process shows a structure “problem/question – search solution/answer – find solution/answer”. Looking at one of the ways knowledge is represented, in the form

of rules *if-then*, it could be found a similarity with this process. At this moment, it is important to mention that a rule consists of two parts: the *if* part, called the antecedent (premise or condition); and the *then* part, called the consequent (conclusion or action) [32]:

$$R_1 : if < \text{antecedent} > \text{then} < \text{consequent} >;$$

Now, going back to analyzing the process to obtain tacit knowledge it can be observed that: there is a problem that needs to be solved; the problem could have a lot of solutions but some of them would be more suitable than others from a person's perspective; and the person would stop his/her searching until he/she believes the answer he/she gets is good enough. So matching the process for tacit knowledge and the knowledge in the form of rules, it could be said that: the "problem/question" part of the process for obtaining tacit knowledge could be represented by the *if* $<\text{antecedent}> part of the rule; the "find solution/answer" could be represented by the *then* $<\text{consequent}> part; finally, an *<antecedent>* can have multiple consequents which match with the idea that a person would look for the more suitable solution from his/her perspective, so the multiple consequents represents the "search solution/answer" part of the process to obtain tacit knowledge.$$

$$\begin{aligned} R_1 : if &< \text{antecedent} > \text{then} < \text{consequent} >; \\ &< \text{consequent} >; \\ &\Leftarrow \end{aligned}$$

However, it is worth mentioning that when it comes to tacit knowledge, even though a solution might be useful, from a person's point of view the solution may not fully solve the problem or be the optimal. Or, as mentioned before, a problem can have multiple solutions where some of them would solve the problem better than others. Here arises the need to give these multiple solutions a proper value that represents the degree to which a person believes that a solution solves the problem. Now simple rules are not enough to represent tacit knowledge, there is a need to also rate the solutions given to a problem, so in this paper it is proposed to use heuristic knowledge to give a proper representation to tacit knowledge. For heuristic knowledge it is supposed a set of heuristic rules R that includes [33]:

$$\begin{aligned} R_1 : if E_1 \text{ then } H_{11} \text{ with a degree of belief } C_{11}; \\ H_{12} \text{ with a degree of belief } C_{12}; \end{aligned}$$

$$\Leftarrow$$

$$\begin{aligned} R_2 : if E_1 \text{ then } H_{21} \text{ with a degree of belief } C_{21}; \\ H_{22} \text{ with a degree of belief } C_{22}; \end{aligned}$$

$$\Leftarrow$$

$$\Leftarrow$$

Where E_1, E_2, \dots, E_n are values (or propositions) of the variable E , and E_i is called the problem/question of the rule R_i . H_{ij} is a subset of values (or propositions) of the variable H , and H_{ij} is called one of the solutions/answers of the rule R_i . C_{ij} is called a rule of strength and denotes the degree to which a person believes that a solution/answer H_{ij} solves the problem/question E_i .

In the adoption process, E_i (problem/question) refers to all the problems or questions the people involved in the adoption of a quality model have about things related to this process. H_{ij} (solution/answers) refers to the solutions or answers a person, usually an expert in the domain, propose for a given problem/question.

Because of E_i 's and H_{ij} 's nature, these are merely given by people. In C_{ij} 's case there are some other things that must be taken in count like: people's involved experience, experts' opinions, the experience of people who implement that solution/answer for a similar problem/question, people's opinion should be given a weight based on their expertise, etc. So, until this moment, it is necessary to find a suitable method to give C_{ij} a proper value.

2.3 Tacit Knowledge and Its Degree of Belief

There already exists methods to manage uncertainty and imprecision that could be used to calculate the degree of belief of tacit knowledge. Some of these methods are: probabilistic techniques, such as Bayes' theorem and Dempster-Shafer theory; certainty factor-based reasoning; and fuzzy techniques [32][34]. For this work, it was decided to use the certainty factors technique due that for tacit knowledge it is not easy to have probabilistic data, instead common sense which is more oriented to expertise judgment.

The theory of certainty factors was developed in the early 1970's for a medical project [34]. The certainty factor (cf) was introduced as a number to measure the expert's belief [32]. The maximum value of a certainty factor is 1.0 (which means the expert believes that the rule is definitely true), and the minimum is -1.0 (which means the expert believes that the rule is definitely false). In the rule:

if E_1 then H_{11} with a degree of belief C_{11} ;

C_{ij} refers to the certainty factor, which represents belief in solution H_{ij} given that the problem E_i has occurred. This theory is grounded on two measures: belief and disbelief [36]. To measure belief it was proposed the function $MB(H_{ij}, E_i)$, and $MD(H_{ij}, E_i)$ is the function to measure disbelief. $MB(H_{ij}, E_i)$ indicates the degree to which belief in solution H_{ij} would be increased if problem E_i was observed. $MD(H_{ij}, E_i)$ indicates the degree to which disbelief in solution H_{ij} would be increased by observing the same problem E_i . These functions are defined in terms of conditional and *a priori* probabilities [35]:

$$MB(H_{ij}, E_i) = \begin{cases} \frac{\max[p(H_{ij} | E_i), p(H_{ij})] - p(H_{ij})}{\max[1, 0] - p(H_{ij})} & \text{if } p(H_{ij}) = 1 \\ \frac{\min[p(H_{ij} | E_i), p(H_{ij})] - p(H_{ij})}{\min[1, 0] - p(H_{ij})} & \text{otherwise} \end{cases}$$

$$MD(H_{ij}, E_i) = \begin{cases} \frac{\max[p(H_{ij} | E_i), p(H_{ij})] - p(H_{ij})}{\max[1, 0] - p(H_{ij})} & \text{if } p(H_{ij}) = 0 \\ \frac{\min[p(H_{ij} | E_i), p(H_{ij})] - p(H_{ij})}{\min[1, 0] - p(H_{ij})} & \text{otherwise} \end{cases}$$

where:

- $p(H_{ij})$ is the prior probability of H_{ij} being a solution; and
- $p(H_{ij}|E_i)$ is the probability that H_{ij} is the is a solution given the problem E_i .

Table 1. Uncertain terms and their respective certainty factors

Term	Certainty factor
Definitely not	-1.0
Almost certainly not	-0.8
Probably not	-0.6
Maybe not	-0.4
Unknown	-0.2 to +0.2
Maybe	+0.4
Probably	+0.6
Almost certainly	+0.8
Definitely	+1.0

To combine the belief and disbelief functions in one number and obtain the certainty factor, which indicates the total belief in solution H_{ij} , it is used the following equation [32]:

$$cf = \frac{MB(H_{ij}, E_i) - MD(H_{ij}, E_i)}{1 - \min[MB(H_{ij}, E_i), MD(H_{ij}, E_i)]}$$

The values of $MB(H_{ij}, E_i)$ and $MD(H_{ij}, E_i)$ range between 0 and 1, while the value of cf range from -1 to +1. Table 1 shows some uncertain terms and its certainty factor [37]. So, for example, if we have the rule:

if E_1 then H_{11} with a degree of belief 0.6;

according to the table, it means that H_{11} is ***probably*** the solution to problem E_1 . If a negative value for the degree of belief is encounter, such as:

if E_1 then H_{11} with a degree of belief -0.8;

it means that H_{11} is ***almost certainly not*** the solution to problem E_1 .

As seen before, certainty factors can be used to give a rule's degree of belief a proper value, and so convert normal rules of knowledge into rules that represent tacit knowledge, due that it allows to indicate the degree to which a person beliefs a solution solves a problem, which is the way tacit knowledge works.

For the adoption process of a quality model, an individual can express his or her tacit knowledge as proposed in this paper: first, states the problem (E_i); second, proposes a solution (H_{ij}); third, based on his or her experience must give values to $p(H_{ij})$ and $p(H_{ij}|E_i)$; fourth, the individual calculates $MB(H_{ij}, E_i)$ and $MD(H_{ij}, E_i)$ values; and finally, with the values obtained from applying the $MB(H_{ij}, E_i)$ and $MD(H_{ij}, E_i)$ functions, the value for the degree of belief (C_{ij}) is estimated. If the results of the proposed solution were not good enough, from the perspective of the individual, then he or she may seek for other solutions until the individual finds one that suits his or her needs.

The rules generated can be stored, so that if another individual has the same or a similar problem he or she can search for the solutions proposed for the problem, look at their degree of belief and choose a solution that best suits his or her needs, and/or choose the solution with the highest degree of belief.

2.4 Example

Suppose there is an organization that is running an adoption process of a quality model. This organization employs an expert to guide them during the process. The expert tries to solve the problems and answer the questions that arise from the people involved in the adoption. The organization is interested in keeping track of all this knowledge generated, so the expert and the people involved in the process begin to capture it in a knowledge base. An example of the knowledge they record is shown below.

A problem that people involved in the adoption present is a difficulty in defining the process they're going to implement.

The expert, based on his/her experience, knows that between the things that could be done to solve the problem, one is to to the person in charge of the process, and another one is to adapt the process to the organization's way of working instead of trying to adapt the organization's process to the model.

So, the rule could be stated as follows:

if: $E_1 < \text{difficulty in defining process} >$

then: $H_{11} < \text{ask for advice to the person in charge of the process} >$

$H_{12} < \text{Adapt the process to the organization's way of working} >$

Next, the expert assigns values to $p(H_{11})$, $p(H_{11} | E_1)$, $p(H_{12})$ and to $p(H_{12} | E_1)$:

$$p(H_{11}) = 0.5; \quad p(H_{11} | E_1) = 0.6$$

$$p(H_{12}) = 0.5; \quad p(H_{12} | E_1) = 0.7$$

Applying certainty factors for H_{11} :

$$MB(H_{11}, E_1) = \max[1, 0] - p(H_{11}) = 1 - 0.5 = 0.5$$

$$MD(H_{11}, E_1) = \min[1, 0] - p(H_{11}) = 0 - 0.5 = -0.5$$

$$cf = \frac{MB(H_{11}, E_1) - MD(H_{11}, E_1)}{1 - \min[MB(H_{11}, E_1), MD(H_{11}, E_1)]} = \frac{0.5 - (-0.5)}{1 - \min[0.5, (-0.5)]} = \frac{1}{1.05} = 0.66$$

Given that $p(H_{11}) = p(H_{12})$, then cf for H_{12} is also 0.66. Finally, the rule is shown in Table 2. Based on the criteria expressed in Table 1, the results indicate that both H_{11} and H_{12} are *probably* solutions to the problem E_1 .

if <difficulty in defining process> then <ask for advice to the person in charge of the process> with a degree of belief <0.66>

Table 2. The formed rule for E_1

<i>if</i>	< difficulty in defining process >	
<i>then</i>	H11: < ask for advice to the person in charge of the process >	<i>with a degree of belief: <0.66></i>
	H12: < Adapt the process to the organization's way of working >	<i>with a degree of belief: <0.66></i>

Just as presented above, the tacit knowledge that revolves around an adoption process, such as lessons learned and experiences can be documented in this form of heuristic rules. During an adoption process the tacit knowledge generated, which is as rules and indicating its degree of belief, can be stored in a knowledge base. In the knowledge base could be documented solutions to the problems regardless of its positive or negative degree of belief, so that the organization could capture both good practices and lessons learned. The degree of belief helps the people participating in the adoption process to take a decision over which solution(s) to take into consideration and which not to; so that people can choose the best solution to implement, and at the same time avoid to take paths taken before and that didn't work.

All the knowledge generated and stored can be used to support future adoption initiatives. People participating in the adoption process can take advantage from this knowledge, especially advisers, team leaders and any sponsor.

3 Conclusions

In this paper is presented a brief analysis of tacit knowledge in which explains the way it is shared and its levels. For the adoption of a quality model, the most appropriate way to capture the tacit knowledge is for this to be articulable and then externalized so it can be stored for later reused when needed. Because there are many forms of articulable tacit knowledge, it was necessary to choose some of these that could fit into a similar format, to try to give them a proper representation. Analyzing a way for tacit knowledge to be created, it was observed that it could take the format “problem/question – search solution/answer – find solution/answer”. This structure, with some adjustments made, was adapted to the knowledge rule’s structure. It could be done, due that knowledge can be easily expressed in the form *if-then*, even so it was necessary to add a value that allowed to evaluate the correspondence of a solution with a given problem based on an individual experience. For that reason, the degree of belief C_{ij} was combined with the rule, and instead of representing tacit knowledge with simple rules, it was represented in the form of heuristic knowledge. Then it is proposed to give the degree of belief C_{ij} a proper value by using certainty factors theory, which allows to use peoples’ judgment in the evaluation of the rule.

For future work, this idea could be implemented in an expert system for carrying out experiments to test if the proposed representation of tacit knowledge actually works for the real world. But it would be even more interesting to find a way to implement it in a social network so that more people could get access to it, and to measure the level of interest and participation of people in such tool when implemented in a trending technology.

References

1. Minguet, J., Hernández, J.: *La calidad del software y su medida*, Editorial Universitaria Ramón Areces, Madrid, España (2003)
2. Naur, P., Randells, B. (eds.): *Software Engineering: Report of a conference sponsored by the NATO Science Committee*, Garmisch, Germany, Brussels, Scientific Affairs Division, NATO, October 7-11 (1969)
3. Niazi, M., Wilson, D., Zowghi, D., Wong, B.: A model for the implementation of software process improvement: An empirical study. In: Bomarius, F., Iida, H. (eds.) *PROFES 2004. LNCS*, vol. 3009, pp. 1–16. Springer, Heidelberg (2004)
4. De la Villa, M., Ruiz, M., Ramos, I.: Modelos de evaluación y mejora de procesos: análisis comparativo. In: 5th ADIS workshop (Apoyo a la Decisión en la Ingeniería del Software), Málaga, España (2004)
5. López, A., Cabrera, C., Valencia, L.: *Introducción a la calidad de software*. Scientia et Technica, pp. 326–331 (2008)

6. Oktaba, H., García, F., Piattini, M., Ruiz, F., Pino, F., Alquicira, C.: Software process improvement: the Competisoft Project. In: Computing Practices, pp. 21–28. IEEE Computer Society (2007)
7. Niazi, M., Wilson, D., Zowghi, D.: Critical success factors for software process improvement implementation: an empirical study. *Software Process: Improvement and Practice* 2(11), 193–211 (2006)
8. Larsen, E.A., Kautz, K.: Quality Assurance and Software Process Improvement in Norway. *Software Process: Improvement and Practice*, 71–86 (1997)
9. Humphrey, W., Snyder, T., Willis, R.: Software process improvement at Hughes Aircraft. *IEEE Software* 8(4), 11–23 (1991)
10. Coleman, G., O'Connor, R.: Investigating software process in practice: a grounded theory perspective. *Journal of Systems and Software*, 772–784 (2008)
11. Staples, M., Niazi, M., Jeffery, R., Abrahams, A., Byatt, P., Murphy, R.: An exploratory study of why organizations do not adopt CMMI. *Journal of Systems and Software* 80, 883–895 (2007)
12. Wangenheim, C., Weber, S., Hauck, J., Trentin, G.: Experiences on establishing software processes in small companies. *Information and Software Technology* 48, 890–900 (2006)
13. Hyde, K., Wilson, D.: Intangible benefits of CMM-based software process improvement. *Software Process: Improvement and Practice*, 217–228 (2004)
14. Zeineddine, R., Mansour, N.: Software Quality Improvement Model for Small Organizations. In: Yazıcı, A., Şener, C. (eds.) ISCIS 2003. LNCS, vol. 2869, pp. 1027–1034. Springer, Heidelberg (2003)
15. Richardson: SPI models: what characteristics are required for small software development companies? *Software Quality Journal* 10, 100–113 (2002)
16. Fayad, M., Laitinen, M., Ward, R.: Thinking objectively: software engineering in the small. *Communications of the ACM*, 115–118 (2000)
17. Al-Mashari, M., Ziari, M.: BPR implementation process: an analysis of key success and failure factors. *Business Process Management Journal* 5, 87–92 (1999)
18. Baddoo, N., Hall, T.: Demotivators of software process improvement: an analysis of practitioner's view. *Journal of System and Software* 66, 23–33 (2003)
19. Montoni, M., Miranda, R., Rocha, A.R., Travassos, G.H.: Knowledge acquisition and communities of practice: An approach to convert individual knowledge into multi-organizational knowledge. In: Melnik, G., Holz, H. (eds.) LSO 2004. LNCS, vol. 3096, pp. 110–121. Springer, Heidelberg (2004)
20. Niazi, M., Babar, M.: Demotivators of software process improvement: an analysis of Vietnamese practitioners' view. In: International Conference on Product Focused Software Process
21. Johansen, J., Pries-Heje, J.: Success with improvement – requires the right roles to be enacted- in sysbiosis. *Software process: improvement and practice* 12, 529–539 (2007)
22. Christiansen, M., Johansen, J.: ImprovAbility guidelines for low-maturity organizations. *Improvement and Practice* 13, 219–325 (2008)
23. Valtanen, A., Sihvonen, H.: Employess' motivation for SPI: case study in a small finnish software company. In: O'Connor, R.V., Baddoo, N., Smolander, K., Messnarz, R. (eds.) EuroSPI 2008. CCIS, vol. 16, pp. 152–163. Springer, Heidelberg (2008)
24. Niazi, M., Muhammad, A., Nolin, M.: Demotivators of software process improvement: an empirical investigation. *Software Process Improvement and Practice* 13, 249–264 (2008)
25. Messnarz, R.: Human resources based improvement strategies the learning factor. *Software Process: Improvement and Practice* 13, 355–362 (2008)

26. Salazar, A.: Modelo de implantación de gestión del conocimiento y tecnologías de información para la generación de ventajas competitivas. Trabajo final para obtener el título de ingeniero civil informático, Universidad Técnica Federico Santa María, Valparaíso, Chile, p. 90 (2000)
27. Nonaka, I., Takeuchi, H.: *The knowledge creation company: how Japanese companies create the dynamics of innovation*, p. 304. Oxford University Press, New York (1995)
28. Nonaka, I., Konno, N.: The concept of 'Ba': building a foundation for knowledge creation. *California Management Review* 40, 40–54 (1998)
29. Busch, P., Dampney, C.: Graphically defining articulable Tacit Knowledge. In: VIP2000 Pan – Sydney Area Workshop on Visual Information Processing, December 1. University of Sydney (2000)
30. Busch, P., Richard, D., Dampney, C.: Visual Mapping of Articulable Tacit Knowledge. In: Australian Symposium on Information Visualisation, Conferences in Research and Practice Information Technology, Sydney, vol. 9 (December 2001)
31. Busch, P.: Knowledge Management Implications of Articulable Tacit Knowledge: Case Studies on its Diffusion. A dissertation submitted to Macquarie University for the Degree of Doctor of Philosophy in the Department of Computing, Division of Information and Communication Sciences (2004)
32. Negnevitsky, M.: *Artificial Intelligence: A Guide to Intelligent Systems*, 2nd edn. Addison Wesley Longman Publishing Co., Inc., Boston (2005)
33. Liu, W., Hughes, J., McTear, M.: Representing Heuristic knowledge and Propagating Beliefs in the Dempster-Shafer Theory of Evidence (1991)
34. Konar, A.: *Artificial Intelligence and Soft Computing: Behavioral and Cognitive Modeling of the Human Brain*. CRC Press LLC, Florida (2000)
35. Shortliffe, E., Buchanan, B.: A model of inexact reasoning in medicine. *Mathematical Biosciences* 23(3/4), 351–379 (1975)
36. Ng, K.-C., Abramson, B.: Uncertainty management in expert systems. *IEEE Expert* 5(2), 29–47 (1990)
37. Durkin, J.: *Expert Systems Design and Development*. Prentice Hall, Englewood Cliffs (1994)

Design and Implementation of an Office Automation Assistant Utility Using Adobe AIR Technology

Xin Liu¹, Liang Liao², Bin Yang¹, and Heng Liu³

¹ Information Center of Yellow River Conservancy Commission,
Zhengzhou, Henan province, P.R. China

liuxinis@126.com,
yangbin@xxzx.yrcc.gov.cn

² School of Electronics and Information, Zhongyuan University of Technology,
Zhengzhou, Henan province, P.R. China

liaoliangis@126.com

³ Yellow River Engineering Consulting Co., Ltd.,
Zhengzhou, Henan province, P.R. China

liuheng@yerc.com

Abstract. Comprehensive online office automation platform helps one process/exchange online documents, schedule meetings and complete information integration as well as other daily office work. However active reminding function of incoming documents/scheduled meetings still remains unavailable in most of the current office automation systems. Thus, traditionally, users have to repeatedly refresh related web pages to obtain latest information, which inevitably influences working efficiency. To tackle the problem, we design and implement an assistant utility, providing the needed reminding functions. The utility is built on Adobe AIR technology, a desktop-oriented browser-less runtime frame for developing Rich Internet Applications (RIAs). Through the provided web service, the utility gets the latest information from office automation server. Additionally, HTTP service is employed for providing auto update capability. Moreover, memory optimization is considered in the design and implementation.

Keywords: Adobe AIR, web service, HTTP service, auto update, memory optimization.

1 Introduction

Comprehensive online office automation platform plays an important role in E-government daily office work, including processing/exchanging online documents, scheduling meetings and so forth. With the help of this platform, users can work on the computer, and communicate in the net.

In spite of its wide use, currently most office automation platform still lacks active reminding function for incoming documents and scheduled meetings. Sometimes, some urgent document processing might be delayed and some impending

meetings can be missed. Without an appropriate utility helping avoid these, users had to frequently check related web pages for the latest content, which inevitably influences the working efficiency.

To tackle this problem, we design and implement an office automation assistant utility, providing the needed functions of reminding and dynamic interactions. The assistant utility is based on Adobe AIR [1, 2] technology, a popular desktop-based browser-less runtime framework for developing Rich Internet Applications (RIAs) [3]. Via this assistance utility, the reminding messages for latest documents and meetings are pushed to users, releasing them from the burdens of frequently checking the related web content. This utility is designed to regularly obtain information from our office automation platform through Web Services. After some content comparisons and making sure that the information is new, the utility pops up the reminder window to alert users.

Additionally, an auto update function is offered, which is convenient for users to update the assistant utility, while it also brings some extra memory usage, nearly half of the total count. We discuss the memory optimization in the end of paper.

The remainder of this paper is organized as follows: Section 2 discusses the related background. Section 3 introduces Adobe AIR technology, data source and the architecture. Section 4 shows the implementation. Section 5 provides a discussion of the memory optimization. In Section 6, some conclusions and the future work are given.

2 Background

Until recently, most comprehensive online office automation platform still fails to remind users of incoming documents and scheduled meetings. Compared with desktop-based applications, current web-based applications are generally of slow response and limited interactivity [4]. Some desktop schedule software tools provide reminders according to a time schedule set by users, rather than according to the instant arrival of the latest information extracted from other sources. Thus, it is highly possible that these software tools can still fail to provide the timely important reminders to their users.

Adobe AIR as a browser-less runtime framework [1] for developing RIAs. AIR provides multiple ways to RIAs to work with data, enabling RIAs to use local files, or database servers via web services. Thus, Adobe AIR is suitable for developing desktop applications for proving the above mentioned reminding functions.

Auto update, offering and updating to the latest version of utility, is needed function for desktop application. However, easy updating could cause extra memory usage. So some memory optimization should be introduced providing updating service while using less memory.

3 Design

3.1 Technology

Adobe Flex 4 featuring the AIR SDK (Software Development Kit) marks the latest advancement in the evolution of Adobe Flash [5] technologies. AIR is a cross-operating system runtime that allows developers to use their existing web development skills. With powerful html support and great integration with the desktop, like drag and drop, local file system access, and even an embedded SQL database, users are able to build and deploy RIAs. Even though the user's browser was closed, the AIR application can receive and display status updates and notifications [6].

Developers are willing to choose AIR for some reasons. Unlike Browser/Server applications, AIR application is still running whether a user is connected to the Net or not. Compared with traditional Client/Server application, AIR application uses proven web technologies to build engaging cross-platform RIAs which expands web-based applications.

AIR framework is presented in Fig.1. The Flex framework includes three Remote Procedure Call (RPC) components [7] (HTTP service, web service, and remote object) that enable one to integrate the Flex applications with common application server products.

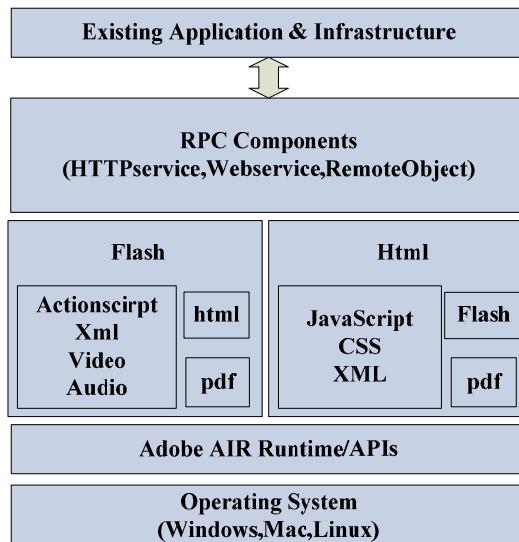


Fig. 1. AIR framework. Adobe AIR is designed to communicate with middleware application servers using a variety of protocols. As a runtime library, AIR is installed on users computer, providing core runtime functionality, needed by AIR-based desktop applications.

3.2 Design Model

SOAP-based Web services [8, 9] employ industry standard XML languages to format both messages and metadata. The SOAP language itself is used to format requests and responses between a client and a server, while WSDL (Web Services Description Language) is used to declare to Web service consumers the structure and capabilities of Web service operations.

The office automation assistant utility is used as a reminder for incoming documents/scheduled meetings. Every ten minutes the utility will call the recent database for new information, and then it will compare new data with old data stored in client computer. The utility window pops up when new information arrives. Web service, offered by office automation platform, supplies latest information about documents and meetings. In the process, the office automation platform provides the method for calling the database, which is published through web service. The utility invoke the web service interface to update local data.

The design model is shown in Fig.2. In Fig.2, four steps are needed for the request-response process. First, the assistant utility sends a request to the office automation platform with the users personal information, as userid. Second, the office automation platform receives the web service request and makes a call for latest data about documents and meetings from database. Then, the office automation platform receives the data from database and provides the results, formatted as SOAP messages, to the assistant utility. Finally, the utility handles the web service results, and make a decision whether to pop up the reminding news or not.

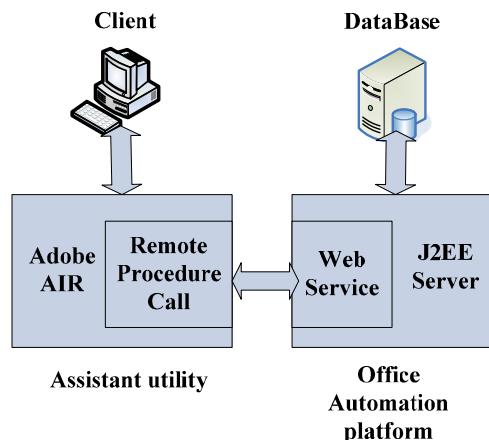


Fig. 2. Design model. Four steps are needed, including client request, invoking web service, web service response, response handling.

4 Application

According to different functions, the office automation assistant utility is supported by four main modules, including network monitor module, todolist module, meeting schedule module, and auto update module.

4.1 Network Monitor Module

The assistant utility startup is accompanied with the computer running boot. When the network connection is later than the startup of utility, the utility would capture the wrong message that the network doesn't work, which is a common problem that most windows startup software faces. In order to ensure the smooth operation of the utility, the network monitor is imperative. Fig.3 shows the network monitor of the assistant utility.

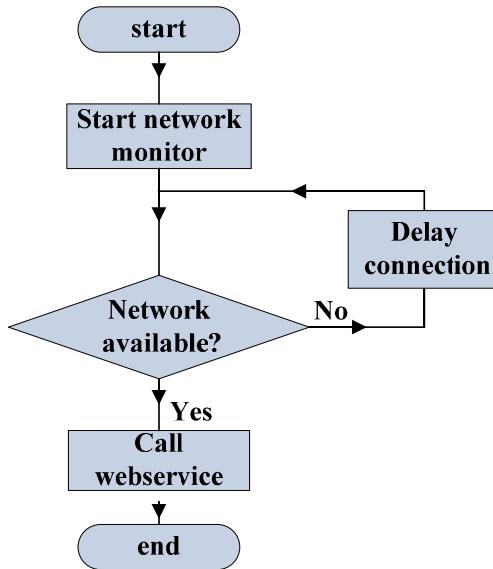


Fig. 3. Network monitor module. The utility is activated, following which the network monitor is started. When the network is available, the utility will directly call the web service from platform. Otherwise, utility delays the connection when the network is unavailable, and then the monitor probes network one minute later.

4.2 Todolist Module

Todolist is used for storing incoming documents. When the assistant utility runs, it helps user connect with todolist of the platform every ten minutes. When the latest documents information is detected, the utility window pops up to remind user of the incoming news. The reminder of todolist adopts the web service

design model in Section 3. Every ten minutes the utility request the web service from office automation platform for latest documents information. The utility also provides the manual refreshing function. The sequence diagram of todolist module is presented in Fig.4.

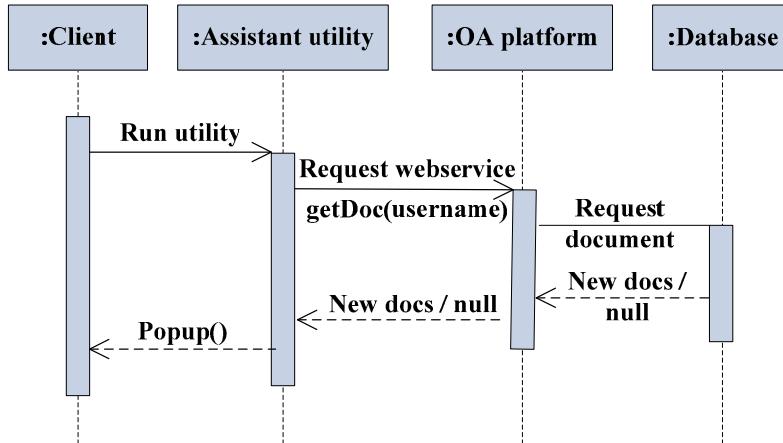


Fig. 4. Sequence diagram of todolist module. The client starts up the assistant utility. Then, the assistant utility connects with OA platform through web service, and sends associated parameter to the platform. Platform requests database for the information about incoming news, and sends the result to the utility. Finally, through some compare between old fresh and the incoming news, the utility decides whether popup or not for reminding the client.

4.3 Meeting Schedule Module

In the comprehensive online office automation platform, meeting schedule includes the public and the personal arrangement of general office work. The assistant utility helps user get the fresh meeting message of today and the next day. If the user has meeting schedule in the morning, the utility window pops up at 8 oclock. The reminder window pops up at 14 oclock if there is some meeting arranged in the afternoon. As the todolist module, meeting schedule module also carries out the web service design model. The difference between the two module is the popup policy. The todolist module pops reminder when fresh new documents coming, while the meeting schedule module pops reminder at regular time when meeting is arranged that day.

4.4 Auto Update Module

Auto update is a necessary function of desktop software. When a new version is available, user can find the updating news as soon as possible through the auto update module, which also provides the software installation. User can choose to

update the utility to the latest version or not. However, the auto update module not only brings convenience for users updating utility, but also consumes a lot more memory, nearly more than half of the total memory usage of the utility. Therefore, we provides an update probe module, using HTTP Service [10] to check the latest version of the utility. HTTP Service component of Adobe AIR has the most flexibility in terms of the format of messages that are exchanged between the client and server at runtime. Unlike the other RPC components, you can use the HTTP Service component with any application server, because it exchanges data in the form of simple HTTP (Hypertext Markup Language) parameters and XML (Extensible Markup Language) of any flavor. The auto update module is provided in Fig. 5.

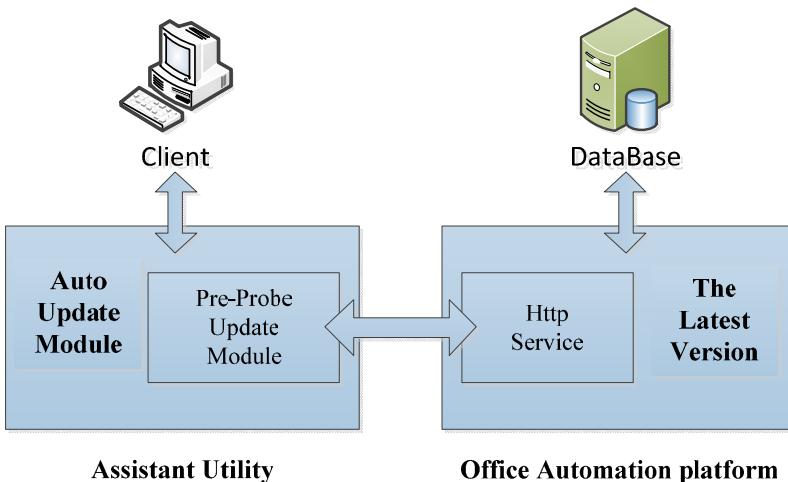


Fig. 5. Auto update module. The latest version of assistant utility is put on the Office Automation platform Server. With the help of pre-probe update module, the auto update module of the utility can easily catch version news from Office Automation platform Server through HTTP Service.

Before the huge auto update module running, the pre-probe module judges whether latest version appears. If the latest version is available, the utility then launches the auto update module. In later Section 5, the details about memory usage will be given.

5 Performance Study

In this section, we evaluate the performance of memory usage of the assistant utility. The environment is based on the E-Government engineering of Yellow River Conservancy Commission. Comprehensive online office automation platform server adopts High Available (HA) cluster of HP Proliant DL580 each server. On the client side, we run the performance tests on an Intel Core2 Quad

CPU 2.40 GHz machine with 3326 MB RAM, and a 260 GB 7200 RPM IDE Drive.

The Adobe Flex profiler [11, 12] is actually an agent running on the Flash player along with the application, providing both performance profiles and memory profiles of application, which helps developers identify performance bottlenecks and memory leaks [13, 14] in the applications. As we launch the assistant utility on the Flex Builder, the profiler records data about the state of the application, including the number of objects, the size of those objects, the number of method calls, and the time spent in those method calls. Therefore, we can quantify how much memory is being used by the utility with the help of Adobe profiler.

For reducing memory usage of utility, we adopt some effective methods like destroying unused objects, removing event listeners, using weak references and so on. Besides, profiler helps us find that auto update module heavily influences the utility memory performance. We run the experiment under two different conditions. First, when there is no latest version of assistant utility, we compare the memory performance of the utility for three different cases, including utility without auto update module, utility with normal update module and utility with pre-probe update module, as shown in Fig.6. Second, when there is new version of the utility, the memory performance is compared between the utility with normal update module and utility with pre-probe update module, as shown in Fig. 7.

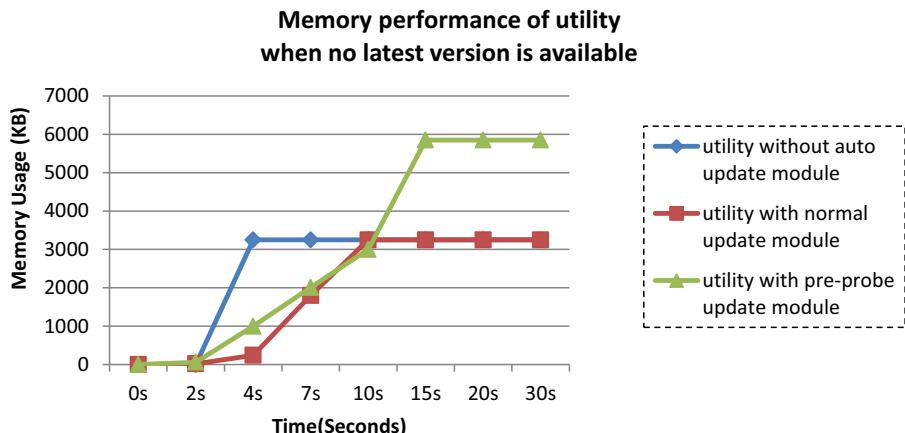


Fig. 6. Memory performance of utility when no latest version is available. After the assistant utility runs stable, the memory cost of utility with pre-probe update module is almost the same as the cost of utility without auto update module, which is nearly a half of the memory usage of utility with normal update module.

From the aforementioned tests, it shows that, using pre-probe update module takes less memory cost than the normal update module in normal cases, although the pre-probe update has no advantage over regular update when the utility

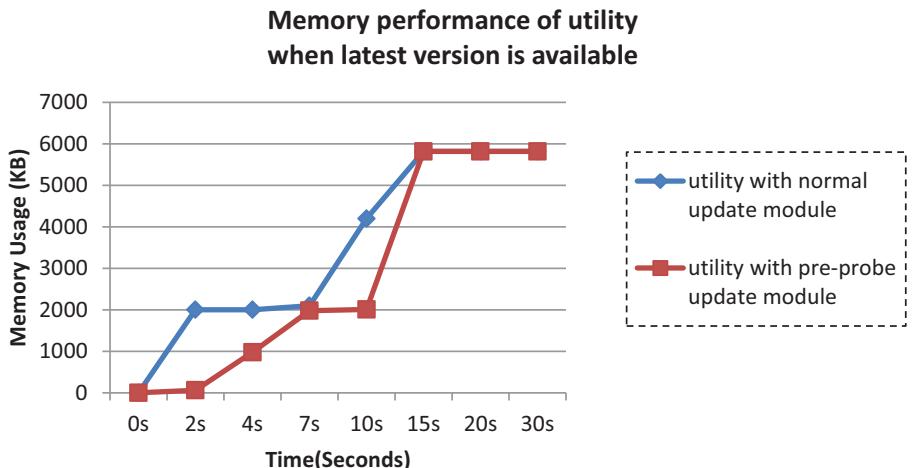


Fig. 7. Memory performance when the latest version is available. This gives us the memory comparison when the utility needs to be updated. Twenty seconds later, the assistant utility with normal update module or with pre-probe update module has the same memory usage.

needs to be updated. However, update situation does not so often happen. In this sense, the per-probe update module helps assistant utility have the update function without the influence of extra memory cost when normal running.

6 Conclusions and Future Work

Traditional office automation platform lacks active reminding function for incoming documents and scheduled meetings. In this paper, we design and implement an office automation assistant utility, representing the needed reminding functions and dynamic interactions. User can directly browse incoming documents/scheduled meetings titles. Through the links of doc/meeting title, platform user can conveniently browse the associated pages and deal with files.

User experience can be enhanced by Adobe AIR technology, expressing richer interface features and providing better information response because of asynchronous communication. Furthermore, the utility provides auto update function, which is convenient for users to update the assistant utility. Our testing results show that a pre-probe update module of the utility takes much less memory cost than the normal one.

For future work, due to its convenience and speed advantage, based on this model, more applications, such as Email system, will be planned to use reminding model in this paper. Besides, we will do more research on improving security and reducing memory usage.

References

1. Tretola, R.: Beginning Adobe AIR: building applications for the Adobe integrated runtime. Wrox (2008)
2. Zhou, W., Wang, Y.: The technology of air applied to automation. In: 2012 International Conference on Computer Science & Service System (CSSS), pp. 159–161. IEEE (2012)
3. Fraternali, P., Rossi, G., Sánchez-Figueroa, F.: Rich internet applications. IEEE Internet Computing 14(3), 9–12 (2010)
4. Liu, X., Liao, L., Duan, Y., Yang, B.: Email information integration with sso in portal service based on ajax. In: 2010 International Conference on Computer Application and System Modeling (ICCASM), pp. V12–V544. IEEE (2010)
5. Leuchner, M., Anderson, T., Wright, M.: Adobe AIR: Create-modify-reuse. Wrox (2008)
6. Labriola, M., Tapper, J.: Breaking out of the web browser with adobe air. IEEE Internet Computing (2008)
7. Gassner, D.: Flash Builder 4 and Flex 4 Bible, vol. 688. Wiley (2010)
8. W3C: Web services activity, <http://www.w3.org/2002/ws/>
9. Curbera, F., Leymann, F., Storey, T., Ferguson, D., Weerawarana, S.: Web services platform architecture: SOAP, WSDL, WS-policy, WS-addressing, WS-BPEL, WS-reliable messaging and more. Prentice Hall PTR (2005)
10. Herrington, J., Kim, E.: Getting Started with Flex 3: An Adobe Developer Library Pocket Guide for Developers. Adobe Dev Library (2008)
11. Tiwari, S.: Tuning applications for superior performance. In: AdvancED Flex 3, pp. 145–178 (2008)
12. Balderson, J., Ent, P., Heider, J., Prekaski, T., Sugden, T., Trice, A., Hassoun, D., Berkovitz, J.: Professional Adobe Flex 3. Wrox (2011)
13. J.Y.C.: Flex memory management and memory leaks,
<http://blogagic.com/163/flex-memory-management-and-memory-leaks> (June 2010)
14. Wang, Y., Zhang, L., Han, Y., Li, H., Li, X.: Flex memory: Exploiting and managing abundant off-chip optical bandwidth. In: Design, Automation & Test in Europe Conference & Exhibition (DATE), pp. 1–6. IEEE (2011)

Modified NFTSM on State in Switched Reluctance Motor Speed Controller

Yong Cheng

School of Electrical and Control Engineering, Xi'an University of Science and Technology,
710054 Xi'an, China
chengbati@163.com

Abstract. In switched reluctance motor driving system(SRD) , accurate controlling speed was important. In the paper, a nonsingular fast terminal sliding mode controller based on states (NFTSM-S) was proposed with judging convergent states, which combined linear sliding mode controller with nonsingular terminal sliding mode with state. NFTSM-S reduced convergent time and ripple of system output, in which state parameters was accurate at convergent state. The simulation results show it is a valid to switched reluctance motor speed tracing control with faster speed response.

Keywords: speed control, sliding mode controller (SMC), nonsingular fast terminal sliding mode controller based on states (NFTSM-S), direct instantaneous torque control (DITC), switched reluctance motor (SRM).

1 Introduction

SRM has lots of merits in driving system. There were strong nonlinear attributes of flux in SRM, which had driven nonlinear feature in torque and flux circuit. Most of these nonlinear attributes were difficult to describe in normal arithmetic. SMC was used successfully in different nonlinear system. Some scholars had performed this robust controller into nonlinear servo system successfully, which was SMC. SMC was simple and robust, which didn't depend on explicit module and used in induction motor, water level and others nonlinear objects. NTSMC(nonsingular terminal sliding mode controller, NTSMC) was introduced, which had solved singular in Terminal Sliding Mode Controller[1]. Some scholars also introduced modified SMC and performed it into nonlinear system[2].

The paper had introduced a nonsingular fast terminal sliding mode controller based on states (NFTSM-S), which had merged linear SMC convergent performance in earlier stage and finite convergent time in NTSMC based on system state variables.

Finally, a new controlling system was derived based on NFTSM-S and DITC(direct instantaneous torque controller, DITC) in SRD. Simulation results showed it has fast convergent speed and higher performance than PID controller.

2 Analyzing

2.1 Design Controller

In terms of uncertain and disturbance of nonlinear system parameters, there was

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = f(X) + g(X) + b(X)u \end{cases} \quad (1)$$

where $X = [x_1, x_2]^T \in \mathbb{R}^2$ and X was state variable of system, $u \in \mathbb{R}$ was input of system, $f(X)$ and $b(X)$ were defined nonlinear function. $g(X) \leq l_g$ was unknown disturbance and uncertain factor.

In paper, a sliding variable in NFTSM-S was defined as

$$s = \alpha x_1 + \beta x_2 + \gamma x_2^{p/q} \quad (2)$$

In equation (2), there were $1 < p/q < 2$, and p, q, α, β and γ were bigger than 0.

It was defined $\|X\|_2 = z_1$, which was boundary point between linear SMC and NTSM. Boundary point was defined as below

$$\begin{cases} \alpha = m_1, \beta = n_1 & \|X\|_2 \geq z_1 \\ \alpha = m_2, \beta = n_2 & \|X\|_2 < z_1 \end{cases} \quad (3)$$

From equation (1,2,3), $\|X\|_2$ was defined by status of system, which was boundary of convergent point between linear SMC and NTSMC comparison to z_1 . So new NFTSMC was convergent to equilibrium point faster and stable. New controller of NFTSMC was yielded

$$u = -b^{-1}(X)[f(X) + \frac{\alpha x_2}{\beta + \frac{\gamma p}{q} x_2^{p/q-1}} + (l_g + \eta) \operatorname{sgn}(s) + \sigma s] \quad (4)$$

where $\eta, p, q, \alpha, \beta, \gamma, \sigma$ were bigger than 0 and $1 < p/q < 2$. In Eq.(4), all relative criteria kept system away singular phenomena in any time.

Define Lyapunov function as $V = s^2/2$, sliding motion was guaranteed by $\dot{V} \leq 0$.

Proof: From Eq.(2), there was

$$\dot{s} = \alpha x_2 + \beta \dot{x}_2 + \gamma \frac{p}{q} x_2^{p/q-1} \dot{x}_2 \quad (5)$$

Derive $V(V = \frac{s^2}{2})$ on t

$$\begin{aligned}\dot{V} &= s \dot{s} = s(\alpha x_2 + \beta \dot{x}_2 + \gamma \frac{p}{q} x_2^{p/q-1} \dot{x}_2) \\ &= s[\alpha x_2 + (\beta + \gamma \frac{p}{q} x_2^{p/q-1})(f(X) + g(X) + b(X)u)] \\ &= s[\alpha x_2 + (\beta + \gamma \frac{p}{q} x_2^{p/q-1})(f(X) + g(X) + b(X)u)] \\ &= s(\beta + \gamma \frac{p}{q} x_2^{p/q-1})[g(X) - (l_g + \eta) \operatorname{sgn}(s) - \sigma s]\end{aligned}$$

$s \neq 0$ and $x_2 \neq 0$, there was

$$\dot{V} \leq -(\beta + \gamma \frac{p}{q} x_2^{p/q-1})[\eta |s| + \sigma] \leq 0$$

Above equation was the condition of the judgment theorem for stability of system.

2.2 Algorithm Comparison

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = 15x_2 + 2\cos(0.5t) + 3u \end{cases} \quad (6)$$

Where $p=9$, $q=7$, $\mu=5$, $\gamma=0.2$, $l_g=40$, $\sigma=5$, $\eta=0.1$, $\begin{cases} \alpha=70, \beta=1 & \|X\|_2 \geq 1 \\ \alpha=1, \beta=0.0001 & \|X\|_2 < 1 \end{cases}$. Take system initial point as (15, -20).

For different algorithm comparing, there sliding faces were selected

(a) NTSM

$$s = x_1 + \gamma x_2^{p/q} \quad (7)$$

Satisfying sliding criteria controller was

$$u = -b^{-1}(X)[f(X) + \frac{q}{rp} x_2^{2-p/q} + (l_g + \eta) \operatorname{sgn}(s) + \sigma s]$$

(b) NFTSM-S

$$s = \alpha x_1 + \beta x_2 + \gamma x_2^{p/q} \quad (8)$$

Satisfying sliding criteria controller was

$$u = -b^{-1}(X)[f(X) + \frac{\alpha x_2}{\beta + \frac{\gamma p}{q} x_2^{p/q-1}} + (l_g + \eta) \operatorname{sgn}(s) + \sigma s]$$

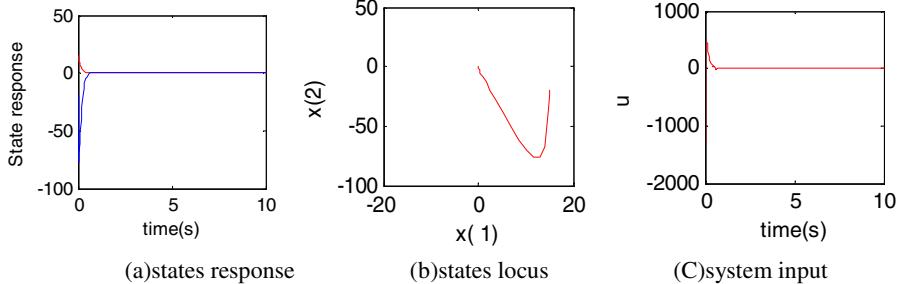


Fig. 1. NTSM simulation results

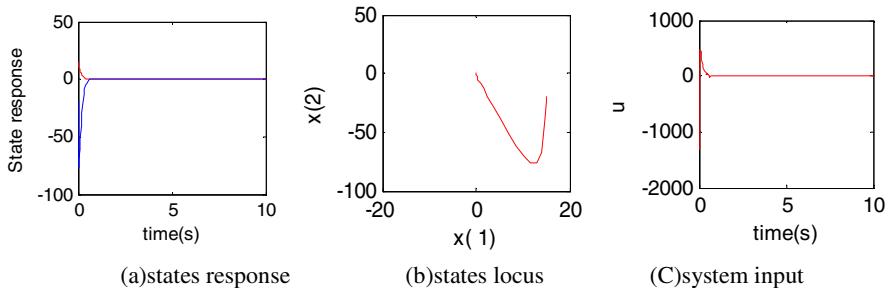


Fig. 2. NFTSM-S simulation results

From simulation results, NFTSM-S took about 0.8s to equilibrium point, at which system was in stable. All these results were new algorithm was used different sliding mode controller in different stage. When system was far away balance point, linear sliding mode controller took main role in convergent. If $\|X\|_2$ was less than 1, equivalent NFTSMC will take main role when system was convergent near to equilibrium points. NTSM used about 1.3s into equilibrium.

3 Speed Regulation Simulation in SRM

3.1 Controller Design

In switched reluctance driving system, it was considered Please note that, if your email address is given in your paper, it will also be included in the meta data of the online version.

$$\begin{cases} \dot{x}_1 = \omega_d - \omega \\ \dot{x}_2 = -\dot{\omega} \end{cases} \quad (9)$$

where ω_d was desired speed , $\dot{\omega}_d=0$ and ω was measured speed of SRM. So function was defined as

$$\begin{cases} \dot{x}_1 = -\omega = x_2 \\ \dot{x}_2 = -\frac{k_\omega}{J}x_2 + \frac{\dot{T}_l}{J} - \frac{1}{J}\dot{T}_e \end{cases} \quad (10)$$

where k_ω , T_l , J and T_e were SRM coefficient of viscosity, load torque, rotational inertia, electric torque.

From Eq.(8), speed controller was \dot{T}_e . This variables must be integrated before inputting into torque controlling system. If load was constant, there was $\dot{T}_l=0$.

From DITC in SRM, control system was simplified as Fig.3 ,which has enhanced system responding time. Measured torque was looked up table $T(\theta,i)$. DITC controlling strategy was took as reference [3].

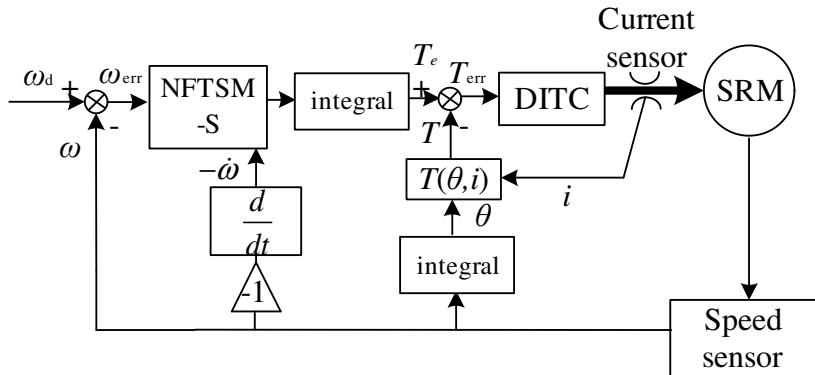


Fig. 3. structure of system controller

3.2 Simulation Results

Simulations were performed with MATLAB7.8.0, at which SRM was four phases. Simulation conditions were set, where turn-on at 2° , switch-off angle at 22° , $k_\omega=0.001\text{N.m.s}$, $J=0.0032\text{kg.m}^2$, $p=9$, $q=7$, $\gamma=0.2$, $l_g=20$, $\eta=0.1$, $X=[x_1, x_2]^T$, $\begin{cases} \alpha=270, \beta=1 & \|X\|_2 \geq 0.5 \\ \alpha=1, \beta=0.0001 & \|X\|_2 < 0.5 \end{cases}, \omega_d=500\text{r/min}$ and $T_i=3\text{N.m}$.

Simulations were Fig.4. In PID, PI variables were $P=8.2$ and $I=2.5$, which were took at trial-and-error method.

From simulation results, NFTSM-S's responding time was faster than PID. From transition time after convergent to equilibrium, NFTSM-S and PID took about 0.014s and 0.031s. Overshoot of NFTSM-S was less than PID.

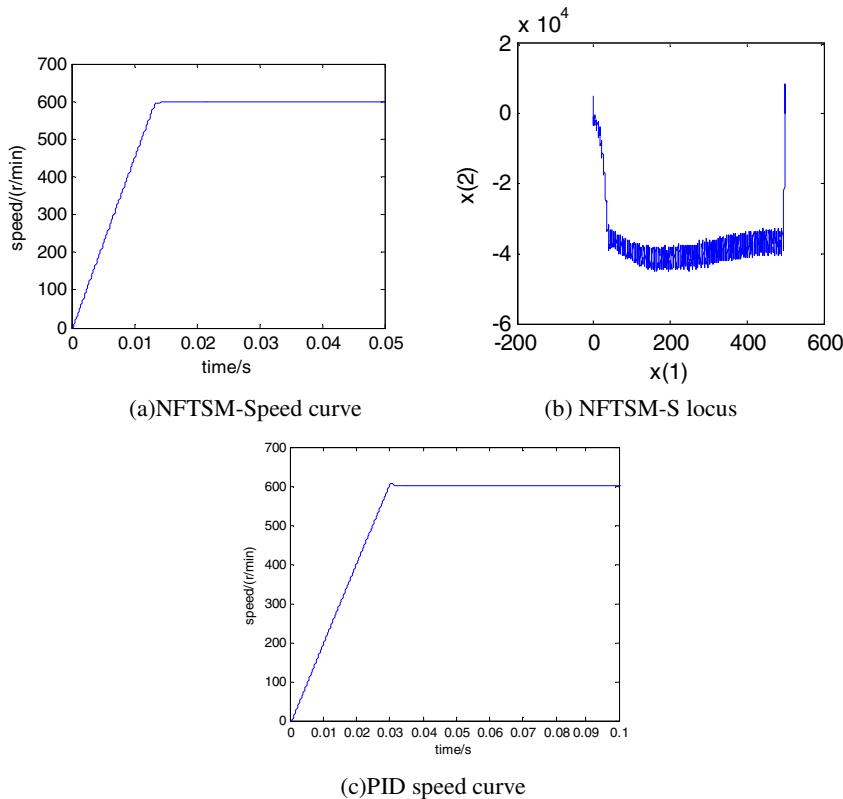


Fig. 4. Simulation results

Boundary of convergent state was set at $\|X\|_2=0.5$, which were combined linear and nonsingular terminal sliding mode controller. If $\|X\|_2$ was set less, it will lead shaking of system during stability boundary.

Acknowledgments. This work was supported by the fund for Young Teachers in Xi'an university of science and technology(201211), fund for Talent Young Teachers in school of Electrical and Control Engineering ,xi'an university of science and technology.

References

1. Feng, Y., Yu, X.H., Man, Z.H.: Non-singular terminal sliding mode control of rigid manipulators. *Automatica* 38, 2159–2167 (2002)
2. Yu, X.H., Man, Z.H.: Fast terminal sliding-mode control design for nonlinear dynamical systems. *IEEE Transactions on Circuits and Systems-I: Fundamental Theory and Applications* 49, 261–264 (2002)
3. Robert, B.I., Rik, W.D.D.: Direct instantaneous torque control of switched reluctance drives. In: IEEE 37th IAS Annual Meeting, pp. 1605–1609 (2002)

Uncertain Nonlinear System Modeling and Identification Using Belief Rule-Based Systems^{*}

Yu-Wang Chen, Jian-Bo Yang, and Dong-Ling Xu

Decision and Cognitive Sciences Research Centre, Manchester Business School,
The University of Manchester, Manchester M15 6PB, United Kingdom
{yu-wang.chen, jian-bo.yang, dong-ling.xu}@mbs.ac.uk

Abstract. Belief rule-based (BRB) systems are an extension of traditional IF-THEN rule based systems and capable of capturing complicated nonlinear causal relationships between antecedent attributes and consequents. In a BRB system, various types of information and knowledge with uncertainties can be represented using belief structures, and a belief rule is designed with belief degrees embedded in its possible consequents. In this paper, we first review the scheme of belief rules for representing and inferring uncertain knowledge. Then we present two BRB system identification methods in which different training objectives are used. Finally, numerical studies are conducted to demonstrate the capability of BRB systems on uncertain nonlinear system modeling and identification.

Keywords: Uncertainty, Identification, Evidential reasoning, Belief rule.

1 Introduction

Uncertain nonlinear system modeling and identification is mainly concerned with characterizing an unknown nonlinear system using measured input-output data in an uncertain environment. It is of fundamental importance in a wide range of areas, such as predictive control, system modeling, decision analysis, etc. Since most real-life systems are nonlinear and often associated with uncertainties due to noises, uncertain physical parameters, or incomplete knowledge [1][3][4], a great difficulty of applying traditional modeling and identification methods is dealing with different types of uncertainties [5]. In the past decades extensive research has been conducted to apply neural networks and fuzzy rule-based systems for identifying uncertain nonlinear systems [2][5]. Sjöberg et al. [5] provided a unified overview on nonlinear black-box system identification models, with structures based on neural networks and fuzzy rules. Nelles [2] performed an in-depth analysis of nonlinear system identification methods, including linear and polynomial approximation, neural networks and fuzzy models. González and Tang [6] presented a continuous-time recurrent neuro-fuzzy network for the black-box identification of a class of dynamic nonlinear systems. On

* This work was partially supported by the UK Engineering and Physical Science Research Council under Grant No.: EP/F024606/1 and the EC REFERENCE project.

the basis of fuzzy identification methodology, Škrjanc et al. [7] presented an interval fuzzy model (INFUMO) to model a class of nonlinear systems with interval parameters. It results in a lower and upper fuzzy model or a fuzzy model with lower and upper parameters. Linear programming techniques are used to find the set of lower and upper parameters using the l_∞ -norm as the optimality criterion. The INFUMO method showed a good capability of modeling internal outputs. However, it used two independent models to identify an uncertain system and it might generate improper identification results with invalid lower and upper bounds [7][8].

Belief rule-based (BRB) systems are an extension of traditional IF-THEN rule based systems [10]. In a BRB system, various types of input information with uncertainties are represented using belief structures, and belief rules are designed with an extended IF-THEN scheme, in which each possible consequent is associated with a belief degree. On the basis of a belief rule base, an input to the BRB system is first transformed into a belief distribution, which is then used to calculate the activation weights of belief rules. The output is inferred by the evidential reasoning (ER) approach. Compared with traditional IF-THEN rule based systems, BRB system are capable of not only providing a more informative scheme for uncertain knowledge representation, but also capturing more complicated nonlinear causal relationships between antecedent attributes and consequents [10]. It has been widely applied to predictive control, fault diagnosis, system modeling, and decision analysis [11][14][15].

The main objective of this paper is to further develop BRB systems for identification of nonlinear uncertain systems. The rest of the paper is organized as follows. In Section 2 uncertain knowledge modeling with belief rules is briefly introduced. In Sections 3 and 4, the identification of BRB systems using least squared error and using l_∞ -norm are presented respectively. Numerical studies are conducted to demonstrate the capability of BRB systems on uncertain nonlinear system modeling and identification. Finally, this paper is concluded in Section 5.

2 Uncertain Knowledge Modeling with Belief Rules

A typical BRB system is given in the form of belief rules as follows.

$$\text{IF } x_1 \text{ is } A_1^k \wedge x_2 \text{ is } A_2^k \wedge \dots \wedge x_{M_k} \text{ is } A_{M_k}^k, \\ R_k : \text{ THEN } \{(D_1, \beta_{1,k}), (D_2, \beta_{2,k}), \dots, (D_N, \beta_{N,k})\}, \left(\sum_{n=1}^N \beta_{n,k} \leq 1 \right), \quad (1)$$

with rule weight θ_k ,
and weight of variables $\delta_{1,k}, \delta_{2,k}, \dots, \delta_{M_k,k}, k \in \{1, \dots, K\}$.

where x_1, x_2, \dots, x_{M_k} denote the antecedent variable in the k th rule, and these variables belong to the whole set of input variables $\mathbf{x} = \{x_i; i = 1, \dots, M\}$. $A_i^k (i = 1, \dots, M_k)$ is the referential value taken by the i th antecedent variable in the k th

rule and $A_i^k \in A_i$. $A_i = \{A_{i,j}; j = 1, \dots, J_i\}$ denotes the set of referential values for the i th antecedent variable and J_i is the number of the referential values. $\beta_{n,k} (n = 1, \dots, N; k = 1, \dots, K)$ represents the belief degree to which the consequent element D_n is believed to be the consequent, given the logical relationship of the k th rule IF x_1 is $A_1^k \wedge x_2$ is $A_2^k \wedge \dots \wedge x_{M_k}$ is $A_{M_k}^k$. Note that the element D_n in the set of consequent elements $D = \{D_n; n = 1, \dots, N\}$ can either be a conclusion or an action and a subset of elements can also be part of the consequent. If $\sum_{n=1}^N \beta_{n,k} = 1$, the k th rule is said to be complete; otherwise, it is incomplete. θ_k is the relative weight of the k th rule, and $\delta_{M_k,k}$ represents the relative weight of variables in the k th rule [10][15].

Under the unified scheme as defined above, a belief rule can represent a functional mapping between antecedent inputs and output possibly with uncertainties. Once a belief rule base is established, the knowledge embedded in all belief rules can be used to perform inference for a specific input vector. Suppose we have an input vector $x_t = \{x_{t,i}, i = 1, \dots, M\}$ at the sampling time t in a complex nonlinear system. Using the referential values defined for the i th antecedent variable x_i , the input $x_{t,i}$ can be transformed to the following belief distribution [9].

$$S(x_{t,i}) = \{(A_{i,j}, \alpha_{i,j}), j = 1, \dots, J_i\} \quad (2)$$

where

$$\alpha_{i,j} = \frac{A_{i,j+1} - x_{t,i}}{A_{i,j+1} - A_{i,j}} \text{ and } \alpha_{i,j+1} = 1 - \alpha_{i,j}, \quad \text{if } A_{i,j} \leq x_{t,i} \leq A_{i,j+1}$$

$$\alpha_{i,j'} = 0, \quad \text{for } j' = 1, \dots, J_i \text{ and } j' \neq j, j+1$$

Here, $\alpha_{i,j}$ represents the similarity degree to which the input value $x_{t,i}$ matches the referential value $A_{i,j}$. After all the inputs are transformed into belief distributions, the activation weight of the k th belief rule can be calculated as follows [10].

$$w_k(\mathbf{x}) = \frac{\theta_k \prod_{i=1}^{M_k} (\alpha_{i,j}^k)^{\bar{\delta}_i}}{\sum_{l=1}^K \left[\theta_l \prod_{i=1}^{M_l} (\alpha_{i,j}^l)^{\bar{\delta}_i} \right]}, \text{ and } \bar{\delta}_i = \frac{\delta_i}{\max_{i=1, \dots, T_k} \{\delta_i\}} \quad (3)$$

Further, the belief degrees on the inference output can be generated through the aggregation of all activated belief rules using the following analytical evidential reasoning (ER) approach [12].

$$\{D_n\}: \beta_n(\mathbf{x}) = \mu \times \left[\prod_{k=1}^K \left(w_k(\mathbf{x})\beta_{n,k} + 1 - w_k(\mathbf{x}) \sum_{i=1}^N \beta_{i,k} \right) - \prod_{k=1}^K \left(1 - w_k(\mathbf{x}) \sum_{i=1}^N \beta_{i,k} \right) \right] \quad (4)$$

$$\{\mathbf{D}\}: \beta_D(\mathbf{x}) = \mu \times \left[\prod_{k=1}^K \left(1 - w_k(\mathbf{x}) \sum_{i=1}^N \beta_{i,k} \right) - \prod_{k=1}^K (1 - w_k(\mathbf{x})) \right] \quad (5)$$

where

$$\mu = \sum_{j=1}^N \prod_{k=1}^K \left(w_k(\mathbf{x})\beta_{j,k} + 1 - w_k(\mathbf{x}) \sum_{i=1}^N \beta_{i,k} \right) - (N-1) \prod_{k=1}^K \left(1 - w_k(\mathbf{x}) \sum_{i=1}^N \beta_{i,k} \right) - \prod_{k=1}^K (1 - w_k(\mathbf{x}))$$

β_D represents the remaining belief degrees unassigned to any known D_n . It has been proven that $\sum_{n=1}^N \beta_n(\mathbf{x}) + \beta_D(\mathbf{x}) = 1$. If all belief rules are complete, i.e., $\sum_{n=1}^N \beta_{n,k} = 1, k = 1, \dots, K$, then $\beta_D(\mathbf{x}) = 0$. The final inference output $f(\mathbf{x})$ by aggregating the K rules, which are activated by the actual input vector \mathbf{x} can be represented as

$$S(f(\mathbf{x})) = \{(D_n, \beta_n(\mathbf{x})), n = 1, \dots, N\} \quad (6)$$

Suppose that the utility of each consequent element D_n is given by $u(D_n)$. The numeric output of a BRB system is given as

$$f(\mathbf{x}) = \sum_{n=1}^N u(D_n) \beta_n(\mathbf{x}) \quad (7)$$

In uncertain nonlinear systems, belief rules may not all be complete, and then we have $\beta_D(\mathbf{x}) \geq 0$. As a result, $\beta_n(\mathbf{x})$ is used to denote the lower bound of the likelihood that the consequent is assessed to D_n . The upper bound of the likelihood is given by $(\beta_n(\mathbf{x}) + \beta_D(\mathbf{x}))$. Complementary to the belief distribution as denoted by (7), an utility interval can also be established for representing the uncertain output, which is characterized by the maximum, minimum, and average utilities of $S(f(\mathbf{x}))$ defined as follows, provided that $u(D_{n+1}) \geq u(D_n)$:

$$f_{\max}(\mathbf{x}) = \sum_{n=1}^{N-1} \beta_n(\mathbf{x}) u(D_n) + (\beta_N(\mathbf{x}) + \beta_D(\mathbf{x})) u(D_N) \quad (8a)$$

$$f_{\min}(\mathbf{x}) = (\beta_1(\mathbf{x}) + \beta_D(\mathbf{x}))u(D_1) + \sum_{n=2}^N \beta_n(\mathbf{x})u(D_n) \quad (8b)$$

It is evident that $f(\mathbf{x}) = f_{\max}(\mathbf{x}) = f_{\min}(\mathbf{x}) = f_{aver}(\mathbf{x}) = \frac{f_{\max}(\mathbf{x}) + f_{\min}(\mathbf{x})}{2}$, if $\beta_D(\mathbf{x}) = 0$. According to the Stone–Weierstrass theorem [13], it has been proven that BRB systems with $\sum_{n=1}^N \beta_{n,k} = 1, k = 1, \dots, K$ can approximate any continuous function on a compact set with arbitrary accuracy [16]. Allowing belief rules to be incomplete further enhances the flexibility of BRB systems. Therefore, it is intuitively reasonable that the utility interval $[f_{\min}, f_{\max}]$ can be used to approximate the lower and upper boundaries of the output domain of an uncertain nonlinear system.

3 Identification of BRB Systems Using Least Squared Error

In this section, we describe how BRB systems can be used to identify nonlinear systems with noises. The mean squared error (MSE), one of the most frequently used measures in supervised learning, is used as the training objective, which can be defined as follows.

$$\min_{\mathbf{P}} \xi(\mathbf{P}) = \frac{1}{T} \sum_{t=1}^T (f_{aver}(\mathbf{x}_t) - y_t)^2 \quad (9)$$

where $\mathbf{P} = < A_{i,j}, J_i, \beta_{n,k}, \theta_k, \delta_i, u(D_n), N >$ denotes the parameter vector. According to their physical meanings and functional requirements, all parameters must satisfy certain equality and inequality constraints [11][15]. $(\mathbf{x}_t, y_t), t = 1, \dots, T$ are available observed input-output data pairs. In the following a simple nonlinear function with noises is employed to demonstrate the capability of BRB systems.

Example 1. In this example, a simple nonlinear function is used to generate training datasets.

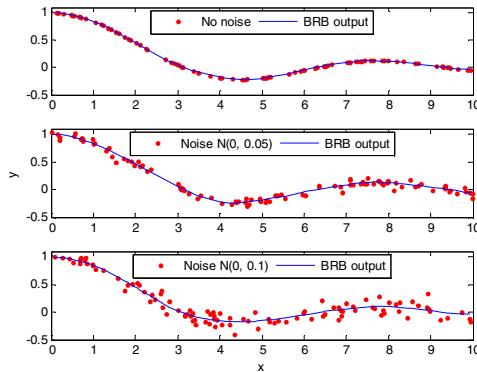
$$y = \frac{\sin x}{x}, \quad x \in [0, 10] \quad (10)$$

To construct a BRB system to simulate this system with uncertain noises $\xi = N(0, \delta)$, $\delta \in \{0, 0.05, 0.1\}$, 4 initial referential values for the input variable x are selected from the interval $[0, 10]$, and they are $\{0, 3, 7, 10\}$. As a result, an initial BRB system can be constructed as follow.

Table 1. Initial belief rule base for Example 1

R_k	θ_k	x	Consequents $\{D_1, D_2, D_3, D_4, D_5\} = \{-0.5, 0, 0.5, 1, 1.5\}$
1	1	0	$\{(D_1, 0), (D_2, 0), (D_3, 0), (D_4, 1), (D_5, 0)\}$
2	1	3	$\{(D_1, 0), (D_2, 0.9), (D_3, 0.1), (D_4, 0), (D_5, 0)\}$
3	1	7	$\{(D_1, 0), (D_2, 0.8), (D_3, 0.2), (D_4, 0.0), (D_5, 0)\}$
4	1	10	$\{(D_1, 0.1), (D_2, 0.9), (D_3, 0), (D_4, 0), (D_5, 0)\}$

Under different noise levels, 100 data sets are generated respectively for training the BRB identification system above. Fig. 1 shows the trained BRB outputs.

**Fig. 1.** BRB identification of nonlinear function with noises

Obviously, the trained BRB system can effectively identify the dynamics of the uncertain systems with noises. However, this certain output provides no information with the uncertain outputs if the uncertainty is incurred by uncertain physical parameters or incomplete knowledge.

4 Identification of BRB Systems Using l_∞ Norm

In this section, we describe how BRB systems can be used to identify nonlinear systems with uncertain physical parameters. The l_∞ -norm is used as the optimality criterion of minimizing the identification errors. The idea of the BRB system identification method for nonlinear systems with uncertain physical parameter is to find a BRB system satisfying

$$f_{\min}(\mathbf{x}) \leq y(\mathbf{x}) \leq f_{\max}(\mathbf{x}) \quad (11)$$

This implies that the outputs of a real uncertain model $y(\mathbf{x})$ can always be found in the utility interval defined in Eqs. (8a) and (8b). The main aim of designing a BRB system is to generate the utility interval on the whole input space as compact as possible in accordance with relevant constraints. This can be realized by solving the following *min-max* optimization problem.

$$\min_{\mathbf{P}} \max_{\mathbf{x}_t \in X} (|y_t - f_{\min}(\mathbf{x}_t)|, |f_{\max}(\mathbf{x}_t) - y_t|) \quad (12)$$

subject to

$$y_t - f_{\min}(\mathbf{x}_t) \geq 0, \quad t = 1, \dots, T \quad (12a)$$

$$y_t - f_{\max}(\mathbf{x}_t) \leq 0, \quad t = 1, \dots, T \quad (12b)$$

The *min-max* optimization problem (12) can be solved as the nonlinear programming problem of minimizing λ subject to the linear and nonlinear constraints (13a)-(13c) and the following inequalities.

$$y_t - (\beta_1(\mathbf{x}_t) + \beta_D(\mathbf{x}_t))u(D_1) - \sum_{n=2}^N \beta_n(\mathbf{x}_t)u(D_n) \leq \lambda, \quad t = 1, \dots, T \quad (13a)$$

$$-y_t + \sum_{n=1}^{N-1} \beta_n(\mathbf{x}_t)u(D_n) + (\beta_N(\mathbf{x}_t) + \beta_D(\mathbf{x}_t))u(D_N) \leq \lambda, \quad t = 1, \dots, T \quad (13b)$$

$$\lambda \geq 0 \quad (13b)$$

These three constraints guarantee that the utility interval of the BRB system enclose all training datasets.

Example 2. A nonlinear function with uncertain parameter is given as follows.

$$y(x) = \cos(x)\sin(x) + \gamma\cos(8x), 0 \leq \gamma \leq 0.2 \quad (14)$$

The function from the class is defined in the domain $\mathbf{U} = \{x \mid -1 \leq x \leq 1\}$ and the set of measured input set is $X = \{x_t \mid x_t = 0.021t, t = -47, -46, \dots, 47\} \subset \mathbf{U}$ [7].

To construct a BRB system, 5 initial referential values for the input variable x are uniformly selected from the interval [-1, 1], and they are {-1, -0.5, 0, 0.5, 1}. The outputs of the nonlinear function with the parameter $\gamma = 0, 0.05, 0.1, 0.15$, and 0.2 are given in the following matrix.

$$\begin{bmatrix} -0.4546 & -0.4619 & -0.4692 & -0.4765 & -0.4837 \\ -0.4207 & -0.4534 & -0.4861 & -0.5188 & -0.5515 \\ 0 & 0.0500 & 0.1000 & 0.1500 & 0.2000 \\ 0.4207 & 0.3881 & 0.3554 & 0.3227 & 0.2900 \\ 0.4546 & 0.4474 & 0.4401 & 0.4328 & 0.4255 \end{bmatrix} \quad (15)$$

In the matrix, each row represents the uncertain outputs for a given referential input. Through observing the measured outputs on these referential inputs, the following referential utilities can be defined for measuring the consequent elements of belief rules.

$$\{D_1, D_2, D_3, D_4, D_5\} = \{-0.8, -0.4, 0, 0.4, 0.8\}$$

As a result, a set of initial belief rules can be constructed as listed in Table 2.

Table 2. Initial belief rule base for Example 2

R_k	θ_k	x	Consequents $\{D_1, D_2, D_3, D_4, D_5\} = \{-0.8, -0.4, 0, 0.4, 0.8\}$
1	1	-1	$\{(D_1, 0.2), (D_2, 0.7), (D_3, 0), (D_4, 0), (D_5, 0), (\textcolor{blue}{D}, 0.1)\}$
2	1	-0.5	$\{(D_1, 0.3), (D_2, 0.5), (D_3, 0), (D_4, 0), (D_5, 0), (\textcolor{blue}{D}, 0.2)\}$
3	1	0	$\{(D_1, 0), (D_2, 0), (D_3, 0.5), (D_4, 0.4), (D_5, 0), (\textcolor{blue}{D}, 0.1)\}$
4	1	0.5	$\{(D_1, 0), (D_2, 0), (D_3, 0.1), (D_4, 0.6), (D_5, 0.1), (\textcolor{blue}{D}, 0.2)\}$
5	1	1	$\{(D_1, 0), (D_2, 0), (D_3, 0), (D_4, 0.8), (D_5, 0.1), (\textcolor{blue}{D}, 0.1)\}$

In Table 2, the sum of the belief degree on the individual consequent element D_n is less than 1, and the belief degree on the whole set of consequent elements \mathbf{D} is not equal to 0 as usual. This implies that the output of the nonlinear system represented by the belief structures is uncertain. To obtain a utility interval to enclose all the uncertain outputs of the nonlinear system, the measured input-output data set above can be used to train the initial belief rules. The optimization model of combining a BRB system with the optimization criterion of using l_∞ -norm is discussed above. The simulation results are presented in Fig. 2.

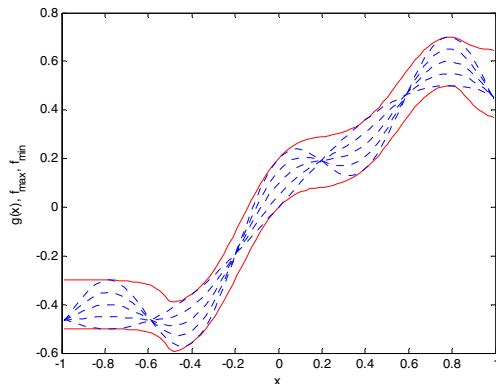


Fig. 2. Interval output of BRB identification

In Fig. 2, the dotted lines are the outputs of the nonlinear function with the above 5 different parameters, and the solid lines are the lower and upper bounds of the uncertain outputs respectively, which are generated by the trained BRB system identification model. It is obvious from Fig. 2 that the trained BRB system can

generate very tight boundaries around the family of the nonlinear functions. In the trained BRB system, the referential values of the input variable x are updated to be $\{-1.0000, -0.4521, 0.3202, 0.7118, 1.0000\}$. Table 3 lists the updated parameters in the trained belief rules.

Table 3. Trained belief rule base for Example 2

R_k	θ_k	x Consequents $\{D_1, D_2, D_3, D_4, D_5\} = \{-0.8000, -0.5500, 0.1654, 0.4395, 0.8000\}$
1	1.0000	-1.0000 $\{(D_1, 0.0000), (D_2, 0.8112), (D_3, 0.0000), (D_4, 0.0000), (D_5, 0.0614), (\textcolor{blue}{D}, 0.1274)\}$
2	0.2500	-0.4521 $\{(D_1, 0.7354), (D_2, 0.0000), (D_3, 0.0000), (D_4, 0.0000), (D_5, 0.1372), (\textcolor{blue}{D}, 0.1274)\}$
3	0.4252	0.3202 $\{(D_1, 0.0000), (D_2, 0.0000), (D_3, 0.5748), (D_4, 0.2946), (D_5, 0.0000), (\textcolor{blue}{D}, 0.1306)\}$
4	0.4614	0.7118 $\{(D_1, 0.0000), (D_2, 0.0019), (D_3, 0.1641), (D_4, 0.0000), (D_5, 0.6900), (\textcolor{blue}{D}, 0.1440)\}$
5	0.6293	1.0000 $\{(D_1, 0.0589), (D_2, 0.0407), (D_3, 0.0000), (D_4, 0.0044), (D_5, 0.7069), (\textcolor{blue}{D}, 0.1891)\}$

In this example, the identification results show that the BRB system is appropriate and effective in the modeling of the uncertainties for the family of nonlinear functions. Even better approximation performance can be obtained if we increase the number of referential values for the input variable according to the universal approximation property of BRB systems.

5 Conclusion

In this paper, two novel identification methods are presented through exploiting the strengths of BRB systems on uncertain knowledge modeling. In the proposed methods, the MSE and l_∞ -norm are used to measure the optimality of the identification model on a finite set of measured data respectively. Numerical studies validated the capability of the proposed BRB identification methods and showed that BRB systems have great potential in nonlinear uncertain system modeling and identification. This is only a pilot study, and further research will be conducted to develop a systematic BRB identification methodology which may be suitable for nonlinear systems with different types of uncertainty.

References

1. Ackermann, J.: Robust Control: Systems with Uncertain Physical Parameters. Springer, Heidelberg (1993)
2. Nelles, O.: Nonlinear System Identification. From Classical Approaches to Neural networks and Fuzzy Models. Springer, Heidelberg (2001)
3. Reinelt, W., Ljung, L.: Robust control of identified models with mixed parametric and non-parametric uncertainties. European Journal of Control 9(4), 373–380 (2003)
4. Isermann, R., Münchhof, M.: Identification of Dynamical Systems: An Introduction with Applications. Springer, Heidelberg (2011)
5. Sjöberg, J., Zhang, Q., Ljung, L., Berveniste, A., Delyon, B., Glorennec, P., Hjalmarsson, H., Juditsky, A.: Nonlinear black-box modeling in system identification: a unified overview. Automatica 31, 1691–1724 (1995)

6. González-Olvera, M.A., Tang, Y.: Black-Box Identification of a Class of Nonlinear Systems by a Recurrent Neurofuzzy Network. *IEEE Transactions on Neural Networks* 21(4), 672–679 (2010)
7. Škrjanc, I., Blažič, S., Agamennoni, O.: Interval Fuzzy Model Identification Using l_∞ – Norm. *IEEE Transactions on Fuzzy Systems* 13(5), 561–567 (2005)
8. Senthilkumar, D., Mahanta, C.: Identification of uncertain nonlinear systems for robust fuzzy control. *ISA Transactions* 49, 27–38 (2010)
9. Yang, J.B.: Rule and utility based evidential reasoning approach for multiple attribute decision analysis under uncertainty. *European Journal of Operational Research* 131(1), 31–61 (2001)
10. Yang, J.B., Liu, J., Wang, J., Sii, H.S., Wang, H.W.: A belief rule-base inference methodology using the evidential reasoning approach – RIMER. *IEEE Transactions on Systems, Man, and Cybernetics – Part A* 36(2), 266–285 (2006)
11. Yang, J.B., Liu, J., Xu, D.L., Wang, J., Wang, H.W.: Optimization models for training belief rule based systems. *IEEE Transactions on Systems, Man, and Cybernetics – Part A* 37(4), 569–585 (2007)
12. Wang, Y.M., Yang, J.B., Xu, D.L.: Environmental Impact Assessment Using the Evidential Reasoning Approach. *European Journal of Operational Research* 174(3), 1885–1913 (2006)
13. Cheney, E.W.: *Introduction to Approximation Theory*. AMS Chelsea Publishing (1998)
14. Zhou, Z.J., Hu, C.H., Xu, D.L., Yang, J.B., Zhou, D.H.: New model for system behaviour prediction based on belief-rule-based systems. *Information Sciences* 180, 4834–4864 (2010)
15. Chen, Y.W., Yang, J.B., Xu, D.L., Zhou, Z.J., Tang, D.W.: Inference analysis and adaptive training for belief rule based systems. *Expert Systems with Applications* 36(10), 12845–12860 (2011)
16. Chen, Y.W., Yang, J.B., Xu, D.L., Yang, S.L.: On the inference and approximation properties of belief rule based systems. *Information Sciences* 234, 121–135 (2013)

Author Index

- Ben Yaghlane, Boutheina 50
Böck, Ronald 13
Bordes, Jean-Baptiste 65
- Chen, Feng 114
Chen, Yu-Wang 209
Cheng, Enxiu 149
Cheng, Yong 202
- Davoine, Franck 65
Denœux, Thierry 65
- Fei, Xie 157
- Glüge, Stefan 13
- Haider, Sajjad 1
Hammami, Imen 79
Hamouda, Atef 79
Huynh, Van-Nam 37
- Jiaolong, Liu 157
Jing, Wang 157
Juárez-Ramírez, Reyes 167, 180
- Laâmmari, Wafa 50
Lanfen, Lin 157
Lawry, Jonathan 101
Lefèvre, Eric 79
- Li, Guanyi 101
Li, Shengfei 114
Li, Tianwen 137
Liao, Liang 192
Liu, Heng 192
Liu, Weiyi 137
Liu, Xin 192
- Ma, Feng 137
Ma, Tieju 23, 37
Meng, Wang 157
- Ocegueda-Miramontes, Violeta 167, 180
- Penghua, Yu 157
Pimienta-Romo, Rafael 167
- Qin, Zengchang 91
- Raza, Saleha 1
- Samet, Ahmed 79
Simon, Christophe 50
- Wang, Chuanqian 91
Wang, Jing 114
Wang, Lei 126
Wang, Tingting 114
Wang, Xiaojun 126
Wang, Yu 114
Wendemuth, Andreas 13
Williams, Mary-Anne 1
- Xiong, Junyi 91
Xu, Dong-Ling 209
Xu, Philippe 65
- Yan, Hong-Bin 23, 37
Yang, Bin 192
Yang, Chen 126
Yang, Jian-Bo 209
- Zhang, Baochang 91, 126
Zhang, YongJun 149