IJCSIS Vol. 11 No. 9, September 2013
ISSN 1947-5500

International Journal of Computer Science & Information Security


Cornell University
Library


Cogprints


Google scholar


.docstoc
find and share professional documents


ScientificCommons

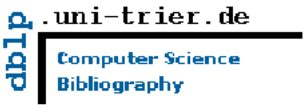

View my documents on
Scribd


BASE
Bielefeld Academic Search Engine


SCIRUS
search engine for science


SciRate.com


CiteSeerx beta


dblp .uni-trier.de
Computer Science
Bibliography


Q·Sensei


DOAJ
DIRECTORY OF
OPEN ACCESS
JOURNALS


EBSCO
HOST


ProQuest

IJCSIS

ISSN (online): 1947-5500

Please consider to contribute to and/or forward to the appropriate groups the following opportunity to submit and publish original scientific results.

CALL FOR PAPERS
International Journal of Computer Science and Information Security (IJCSIS)
January-December 2013 Issues

The topics suggested by this issue can be discussed in term of concepts, surveys, state of the art, research, standards, implementations, running experiments, applications, and industrial case studies. Authors are invited to submit complete unpublished papers, which are not under review in any other conference or journal in the following, but not limited to, topic areas.
See authors guide for manuscript preparation and submission guidelines.

Indexed by Google Scholar, DBLP, CiteSeerX, Directory for Open Access Journal (DOAJ), Bielefeld Academic Search Engine (BASE), SCIRUS, Cornell University Library, ScientificCommons, EBSCO, ProQuest and more.

Deadline: see web site
Notification: see web site
Revision: see web site
Publication: see web site

Context-aware systems
Networking technologies
Security in network, systems, and applications
Evolutionary computation
Industrial systems
Evolutionary computation
Autonomic and autonomous systems
Bio-technologies
Knowledge data systems
Mobile and distance education
Intelligent techniques, logics and systems
Knowledge processing
Information technologies
Internet and web technologies
Digital information processing
Cognitive science and knowledge
Agent-based systems
Mobility and multimedia systems
Systems performance
Networking and telecommunications
Software development and deployment
Knowledge virtualization
Systems and networks on the chip
Knowledge for global defense
Information Systems [IS]
IPv6 Today - Technology and deployment
Modeling
Software Engineering
Optimization
Complexity
Natural Language Processing
Speech Synthesis
Data Mining

For more topics, please see web site https://sites.google.com/site/ijcsis/

For more information, please visit the journal website (https://sites.google.com/site/ijcsis/)

Editorial
Message from Managing Editor

International Journal of Computer Science and Information Security *(IJCSIS – established since May 2009), presents research and development results of lasting significance in the theory, design, implementation, analysis, and application of computing and security. As scholarly open access, peer reviewed international journal with a primary objective to provide the academic community and industry a forum for ideas and for the submission of original research related to Computer Science and Security. The goal is to bring together researchers and practitioners from academia and industry to focus on computer science issues and advancement in these areas. Authors are solicited to contribute to this journal by submitting articles that illustrate research results, projects, surveying works and industrial experiences that describe significant advances in the Computer Science & Security. IJCSIS archives all publications in major academic/scientific databases; abstracting/indexing, editorial board and other important information are available online on homepage. Indexed by the following International agencies and institutions: Google Scholar, Bielefeld Academic Search Engine (BASE), CiteSeerX, SCIRUS, Cornell's University Library EI, Scopus, DBLP, DOI, ProQuest, EBSCO. Google Scholar reported a large amount of cited papers published in IJCSIS. IJCSIS supports the Open Access policy of distribution of published manuscripts, ensuring "free availability on the public Internet, permitting any users to read, download, copy, distribute, print, search, or link to the full texts of [published] articles".*

IJCSIS editorial board consisting of international experts solicits your contribution to the journal with your research papers, projects, surveying works and industrial experiences. IJCSIS is grateful for all the insights and advice from authors & reviewers.

We look forward to your collaboration. For further questions please do not hesitate to contact us at ***ijcsiseditor@gmail.com****.*

A complete list of journals can be found at:
http://sites.google.com/site/ijcsis/

IJCSIS Vol. 11, No. 9, September 2013 Edition

ISSN 1947-5500 © IJCSIS, USA.

Journal Indexed by (among others):

IJCSIS EDITORIAL BOARD

TABLE OF CONTENTS

3. Paper 31081315: An Efficient Parallel Based Diabetic Retinopathy Grading Algorithm (pp. 12-23)

Associate. Prof. Ghada F. Elkabbany, Informatics Dept., Electronics Research Institute, Cairo, Egypt
Associate. Prof. Heba A. Elnemr, Computers & Systems Dept., Electronics Research Institute, Cairo, Egypt

Abstract — Medical imaging has revolutionized the medicine by providing cost-efficient healthcare and effective diagnosis in all major disease areas. Diabetes is a chronic disease and a major public health challenge worldwide. Diabetes complications can be prevented or delayed by early identification of people at risk. There are several approaches carried out on this context. There are many methods are available for prediction but because natural process of this kind are very complex which involves large number of input variables so we need very large dataset for proper prediction, it also has disadvantage of high algorithmic complexity and extensive memory requirements of the required quadratic programming in large-scale tasks. For very large and complex problems it is better to divide data in parts which not only decrease the complexity but also provide the capability of handling the tasks in parallel. This work presents and evaluates a method for introducing parallelism into the diabetic retinopathy grading algorithm proposed in [1]. The aim is to improve its performance by utilizing parallel concepts which distribute the employed datasets into different nodes which reduces the computational complexity, processing power and memory requirements. To implement the parallel processing on DR grading algorithm presented in [1], different levels of parallelism are used. Multi-level of parallelization improves the system utilization and throughput. In the proposed parallel DR grading algorithm, when the number of nodes is large load imbalance occurs. Thus, static load balancing algorithm is applied to get better performance. The suggested parallel DR grading method is simple and can be used for large datasets. This method also provides the flexibility to be modified according to the dataset size, number of nodes and memory available on different units. We have tested the proposed algorithm and the results are very encouraging.

Keywords - Diabetic retinopathy; Clustering; Parallel processing; Texture feature extraction; Gray level co-occurrence matrix; Parallel techniques

4. Paper 31081319: Maximum Battery Capacity Routing to Prolong Network Operation Lifetime in Wireless MESH Network alongside the OLSR Protocol (pp. 24-29)

Ramezanali Sadeghzadeh, Faculty of Electrical and Computer Engineering, K.N. Toosi University of Technology, Tehran, Iran
Afsaneh Saee Arezoomand, Dept. of Electrical and Computer Engineering, Science and Research Branch, Islamic Azad University
Tehran, Iran
Mohammad Zare, Dept. of Electrical and Computer Engineering, Science and Research Branch, Islamic Azad University, Tehran, Iran

Abstract — A wireless mesh network (WMN) is a communications network made up of radio nodes organized in a mesh topology. Wireless mesh networks often consist of mesh clients, mesh routers and gateways. The mesh clients operate on batteries such as cell phone, laptop and .., while the mesh routers forward traffic to and from the gateways which may, but need not, connect to the Internet. To maximize the lifetime of mesh mobile networks, the power consumption rate of each node must be evenly distributed, it is essential to prolong each individual node (mobile) lifetime since the lack of mobile nodes can result in partitioning of the network, causing interruptions in communications between mobile nodes, and finally the overall transmission power for each connection request must be minimized. In this article we propose a new metric to find a proper route in wireless Mesh network and beside it we study OLSR protocol that it can be used in Ad hoc network.

Keywords- Wireless Mesh; Ad hoc Network; Energy Consumption; Power Control; OLSR protocol

5. Paper 31081321: A New Sentinel Approach for Energy Efficient and Hole Aware Wireless Sensor Networks (pp. 30-37)

Dame DIONGUE, PhD Student, Department of Computer Science, Gaston Berger University, BP : 234 Saint Louis, SENEGAL
Ousmane THIARE, Department of Computer Science, Gaston Berger University, BP : 234 Saint Louis, SENEGAL

Abstract — Recent advances in micro-sensor and communication technology have enabled the emergence of a new technology, Wireless Sensor Networks (WSN). WSN have emerging recently as a key solution to monitor remote or hostile environments and concern a wide range of applications. These networks are faced with many challenges such as energy efficiency usage, topology maintenance, network lifetime maximization, etc. Experience shows that sensing and communications tasks consume energy, therefore judicious power management can effectively extend network lifetime. Moreover, the low cost of sensor devices will allows deployment of huge number nodes that can permit a high redundancy degree. In this paper, we focus on the problem of energy efficiency and topology maintenance in a densely deployed network context. Hence we propose an energy aware sleep scheduling and rapid topology healing scheme for long life wireless sensor networks. Our scheme is a strong node scheduling based mechanism for lifetime maximization in wireless sensor networks and has a fast maintenance method to cover nodes failure. Our sentinel scheme is based on a probabilistic model which provides a distributed sleep scheduling and topology control algorithm. Simulations and experimental results are presented to verify our approach and the performance of our mechanism.

Keywords-component; energy conservation; lifetime maximization; topology maintenance; insert (key words)

6. Paper 31081322: Design of Generic Framework for Botnet Detection in Network Forensics (pp. 38-45)

Sukhdilpreet Kaur, Computer Science and Engineering, Punjabi University Regional Centre for Information Technology and Management, Mohali, India
Amandeep Verma, Computer Science and Engineering, Punjabi University Regional Centre for Information Technology and Management, Mohali, India

Abstract — With the raise in practice of Internet, in social, personal, commercial and other aspects of life, the cybercrime is as well escalating at an alarming rate. Such usage of Internet in diversified areas also augmented the illegal activities, which in turn, bids many network attacks and threats. Network forensics is used to detect the network attacks. This can be viewed as the extension of network security. It is the technology, which detects and also suggests prevention of the various network attacks. Botnet is one of the most common attacks and is regarded as a network of hacked computers. It captures the network packet, store it and then analyze and correlate to find the source of attack. Various methods based on this approach for botnet detection are in literature, but a generalized method is lacking. So, there is a requirement to design a generic framework that can be used by any botnet detection. This framework is of use for researchers, in the development of their own method of botnet detection, by means of providing methodology and guidelines. In this paper, various prevalent methods of botnet detection are studied, commonalities among them are established and then a generalized model for the detection of botnet is proposed. The proposed framework is described as UML diagrams.

Keywords- Network forensics, Botnets, Botnet detection methods, class diagrams, activity diagram.

7. Paper 31081325: Visualization for Information Retrieval in Regional Distributed Environment (pp. 46-63)

Mamoon H. Mamoon, Hazem M. El-Bakry, Amany A. Slamaa
Faculty of computer science & information system, Mansoura University, EGYPT

Abstract — Information retrieval (IR) is the task of representing, storing, organizing, and offering access to information items. The problem for search engines is not only to find topic relevant results, but results consistent with the user’s information need. How to retrieve desired information from the Internet with high efficiency and good effectiveness is become the main concern of internet user-based. The interface of the systems does not help them to perceive the precision of these results. Speed, resources consuming, searching and retrieving process also aren't optimal. The search engine's aim is developing and improving the performance of information retrieval system and gifting the user whatever his culture' level. The proposed system is using information visualization for interface

problems, and for improving other side of web IR system's problems, it uses the regional crawler on distributed search environment with conceptual query processing and enhanced vector space information retrieval model (VSM). It is an effective attempt to match renewal user's needs and get a better performance than ordinary system.

Keywords - Regional distributed crawler, VSM, conceptual weighting, visualization, WordNet, information visualization, web information retrieval.

8. Paper 31081327: A Comparative Study of Replication Techniques in Grid Computing Systems (pp. 64-73)

Sheida Dayyani, Department of Computer Engineering, Sheikh Bahaei University, Isfahan, Iran
Mohammad Reza Khayyambashi, Department of Computer Engineering, University of Isfahan, Isfahan, Iran

Abstract — Grid Computing is a type of parallel and distributed systems that is designed to provide reliable access to data and computational resources in wide area networks. These resources are distributed in different geographical locations, however are organized to provide an integrated service. Effective data management in today`s enterprise environment is an important issue. Also, Performance is one of the challenges of using these environments. For improving the performance of file access and easing the sharing amongst distributed systems, replication techniques are used. Data replication is a common method used in distributed environments, where essential data is stored in multiple locations, so that a user can access the data from a site in his area. In this paper, we present a survey on basic and new replication techniques that have been proposed by other researchers. After that, we have a full comparative study on these replication strategies. Also, at the end of the paper, we summarize the results and points of these replication techniques.

Keywords-comparative study; distributed environments; grid computing; data replication

9. Paper 31081332: Parallel Implementation of the Single Source Shortest Path Algorithm on CPU–GPU Based Hybrid System (pp. 74-80)

Dhirendra Pratap Singh, Dept. of Computer Science and Engineering, MANIT, Bhopal, India
Nilay Khare, Dept. of Computer Science and Engineering, MANIT, Bhopal, India

Abstract — Single source shortest path (SSSP) calculation is a common prerequisite in many real world applications such as traveler information systems, network routing table creation etc., where basic data are depicted as a graph. To fulfill the requirements of such applications, SSSP calculation algorithms should process their data very quickly but these data are actually very large in size. Parallel implementation of the SSSP algorithm could be one of the best ways to process large data sets in real time. This paper proposes two different ways of parallel implementation of SSSP calculation on a CPU-GPU (Graphics Processing Unit)-based hybrid machine and demonstrates the impact of the highly parallel computing capabilities of today's GPUs. We present parallel implementations of a modified version of Dijkstra's famous algorithm of SSSP calculation, which can settle more than one node at any iteration. This paper presents a comparative analysis between both implementations. We evaluate the results of our parallel implementations for two Nvidia GPUs; the Tasla C2074 and the GeForce GTS 450. We compute the SSSP on graph having 5.1 million edges in 191 milliseconds. Our modified parallel implementation shows the three-fold improvement on the parallel implementation of simple Dijkstra's algorithm.

Keywords - Graph Algorithm; Compute Unified Device Architecture (CUDA); Graphics Processing Unit (GPU); Parallel Processing.

10. Paper 31081340: Analysis of the Methodology Required for the Simulation of Handover Failure in GSM Network (pp. 81-86)

Syed Foysol Islam, Faculty of Engineering, University of Development Alternative (UODA), Dhaka, Bangladesh
Fahmi Ahmed, Faculty of Engineering, University of Development Alternative (UODA), Dhaka, Bangladesh

Abstract — This research paper shows the methodology needed for the simulation of call drop & handover failure in GSM network tele-traffic through OMNeT++ simulation tool under Windows platform. It measures design conditions and minimum quality standards should provide for operation and simulates call drop and hand over failure in GSM tele-traffic. The simulator has been programmed in OMNeT++, is a discrete event simulator focused on research of wired or wireless networks.

Keywords - Call drop; Handover; Wireless network; Simulator;

11. Paper 31061311: Web Usage Mining through Efficient Genetic Fuzzy C-Means (pp. 87-90)

Deepak Kumar Niware, Department of Computer Science & Engg., TIT, Bhopal (INDIA)
Dr. Setu Kumar Chaturvedi, Department of Computer Science & Engg., TIT, Bhopal (INDIA)

Abstract - In process of knowledge discovery from any weblog dataset, most widely and extensively used clustering algorithm for this purpose is Fuzzy c-means (FCM) algorithm because the data of web-log is unsupervised dataset. Due to sensitivity of FCM, it can be easily trapped in a local optimum, and it is also depends on initialization. In this paper we present use of Genetic algorithm in Fuzzy cmeans algorithm to select initial center point for clustering in FCM. The purpose of this paper is to provide optimum initial solution for FCM with the help of genetic algorithm to reduce the error rate in pattern creation.

Keywords: Fuzzy C-means, Genetic Algorithm, Web log mining, Web usage mining, Web mining.

12. Paper 31081301: Clustering Technique Based Outlier Detection Method for Document Classification (pp. 91-96)

Deepti Jain, SATI Vidisha, Vidisha (M.P.) India
Dr. Bhupendra Verma, Director TIT Bhopal, Bhopal (M.P.) India
Dr. R. C. Jain, Director SATI Vidisha, Vidisha (M.P.) India

Abstract - In this paper, we propose a clustering based technique to capture outliers for document classification and apply Kmeans clustering algorithm to divide the dataset into clusters. The points lying near centroid of the cluster are not probable candidate for outlier and prune out such points from each cluster then calculate a distance based outlier score for remaining points. The computations calculate to the outlier score reduces considerably due to the pruning of some points. Based on the outlier score declare the top n points with the highest score as outliers after that classification technique is applied for categorization. The experimental results using actual dataset demonstrate that even though the number of computations is fewer, the proposed method performs better than the obtainable method.

Keywords: outlier; Cluster; Distance-based; Classification.

13. Paper 31081318: Hybrid Storage Architecture: A survey (pp. 97-102)

Tejaswini Apte, Symbiosis Institute of Computer Studies and Research, Pune, INDIA
Dr. Maya Ingle and Dr. A.K. Goyal, Devi Ahilya Vishwavidyalaya, Indore, Indore, INDIA

Abstract - Database design requirement for large scale OLAP applications differs from small-scale database programs. Database query and update performance is highly dependent on the storage design techniques. Two storage design techniques have been proposed in the literature namely; a) Row-Store architecture and b) Column-Store architecture. This paper studies and combines the best aspect of both Row-Store and Column-Store architectures to better serve an ad-hoc query workload. The performance is evaluated against TPC-H workload.

General Terms: Performance, Design Keywords: Statistics

Bhuvana R., Research Scholar, Department of Computer Science, Vels University, Chennai, INDIA-600117.
Dr. Purushothaman S., Professor, PET Engineering College, Vallioor, INDIA-627117,
Rajeswari R., Research scholar, Mother Teresa Women's University, Kodaikanal, INDIA-624101.

Abstract - Mining of depression data such as depressed mood, feelings of guilt, suicide, insomnia early, insomnia middle, insomnia late, work and activities, retardation, Psychomotor, agitation, anxiety, anxiety somatic, somatic symptoms, somatic symptoms general, genital symptoms, genital symptoms, insight, diurnal variations, depersonalization and decreolization, paranoid symptoms, obsessionals and compulsive symptoms have been collected based on the Hamilton rating scale for depression. This paper presents the implementation of neural network methods for depression data mining and diagnosis patients by using radial basis function (RBF) and Echo state neural network (ESNN). The output of RBF is given as input to ESNN network. A systematic approach has been developed to efficiently mine the depression data for proper diagnosis of the patients.

Keywords: Hamilton Rating Scale Depression data, radial basis function (RBF), echo state neural network (ESNN)

D. Nagarajan, Department of Information Technology, Math Section, Salalah college of Technology, Salalah, Sultanate of Oman.
M. Raji, Department of Mathematics, Asan Memorial College of Engineering and Technology, Chengalpattu, India.

Abstract — Traffic signals are very important vital factor for reduce the traffic pollution in our world. The past three decades researches much attention about the traffic pollution. There are many opportunities to use clever traffic engineering to reduce the impacts of traffic on public transportation. Often these combine traffic signals with short sections of exclusive public transport lanes. The aim of the paper is to reduce the traffic pollution using traffic signal by Markov chain and genetic algorithm.

Keywords- traffic system; continuous time markov chain; genetic algorithm.

*Ali akbar tabibi, Research and Science University of Bushehr, Student Masters, Department of Computer Engineering, Software,*7514963161 *Bushehr Iran*
Seyed javadmirabedini, Islamic Azad University, Central Tehran Branch, Tehran Iran

Abstract - Organizational architecture is composed under a process called *organizational architecture process*. This process is complicated and architecture can use its framework as a modulator of structure to control complicacy and apply the method as a behavior director. In architecture, behavior is prior to structure, and a structure may have different behaviors. But which behavior (method) best suite architecture and thus meet the concerned needs? Evaluation of architecture is needed to answer this question. As an instance, this article aims to demonstrate validity of architecture behavior on intelligent fuel card using colored Petri networks. As result, it revealed up that the given solution led to identify traffic points and thus helped the architecture designers in choosing the right method.

Keywords: Organizational Architecture, Evaluation of Architecture, Colored Petri

Bashir Alam (1), M.N Doja (1), Mansaf Alam (2) and Shweta Malhotra (1)
(1) Department of Computer Engineering Jamia Millia Islamia , New Delhi, India
(2) Department of Computer Science Jamia Millia Islamia , New Delhi, India

Abstract — Cloud computing is a new emerging concept recently introduced in the world. Cloud services on the first hand provides many advantages like pay-as-u-go nature, faster deployment of IT resources and the way of future but on the other hand challenges/ issues of cloud overweight the advantages of cloud. Among all the challenges of cloud, the upmost challenge that the world is facing with cloud is "Security" as clients outsource their personal, sensitive data to the cloud over the internet which can be very dangerous if not secured properly. In this paper we have analyzed security issues of cloud from different aspects along with some implemented solutions. Security of cloud can be categorized by service models provided by service providers, data life cycle security issues and it can be categorized by data security, virtualization security and software/application security. We have also analyzed some implemented solution model based on cryptography and shamir's secret sharing algorithm to some of the security issues.

Keywords- Software as a service (SAAS) Platform as a service (PAAS); Infrastructure as a service (IAAS); Service level agreement (SLA), Multi cloud Database model (MCDB), NetDB2-Multi Share(NetDB2-MS).

18. Paper 31081307: Software Engineering Framework using Agile Dynamic System Development Method for Efficient Mobile Application Development (pp. 126-134)

Mr. N. Balasupramanian, Mr. RamKumar Lakshminarayanan , Dr. RD. Balaji,
Lecturer, Department of IT, Higher College of Technology, Muscat, Sultanate of Oman

Abstract - The mobile industry is changing the technologies very often to attract the customer to a greater extent; whether it is application platforms, devices, technology, features, network models or exploration of application use cases, the speed of change for any one of these technologies means that businesses or opportunities have to think carefully before investing in creating their own applications. Now-a-days, the mobile application development is targeted of introducing many new tools, techniques and methodologies for the application development. This paper provides the development team members a right direction to apply appropriate software engineering framework implementing agile method for the development of mobile application and this paper also gives a comparative study between the XP and DSDM agile methods.

Key Words- Going Mobile, Application Development, Software Engineering, Agile, Framework, XP-Extreme Programming, DSDM-Dynamic System Development Method

Countering Wrapping Attack on XML Signature in SOAP Message for Cloud Computing

Hadi Razzaghi Kouchaksaraei, Alexander G. Chefranov

Department of Computer Engineering, Eastern Mediterranean University
Gazimagusa- Mersin 10- Turkey
Hadi.razzaghi87@gmail.com
Alexander.chefranov@emu.edu.tr

***Abstract*— It is known that the exchange of information between web applications is done by means of the SOAP protocol. Securing this protocol is obviously a vital issue for any computer network. However, when it comes to cloud computing systems, the sensitivity of this issue rises, as the clients of system, release their data to the cloud.**

XML signature is employed to secure SOAP messages. However, there are also some weak points that have been identified, named as XML signature wrapping attacks, which have been categorized into four major groups; Simple Ancestry Context Attack, Optional element context attacks, Sibling Value Context Attack, Sibling Order Context.

In this paper, two existing methods, for referencing the signed part of SOAP Message, named as ID referencing and XPath method, are analyzed and examined. In addition, a new method is proposed and tested, to secure the SOAP message.

In the new method, the XML any signature wrapping attack is prevented by employing the concept of XML digital signature on the SOAP message. The results of conducted experiments show that the proposed method is approximately three times faster than the XPath method and even a little faster than ID.

Keywords: Cloud Computing, SOAP message, XML digital signature, Wrapping attack.

I. INTRODUCTION

Cloud computing is a new technology [1], which provides greatly ascendable resources such as bandwidth, hardware and software, to be utilized as a service for consumers, over the Internet. This concept has attracted wide attention in all kind of industries recently [2]. One of the most significant advantages of using of this technology is that consumers can save the cost of hardware deployment, software license and system maintenance. Consequently, the price of providing and using the systems will be reduced significantly.

However, besides being absolutely beneficial, there are still particular unsolved problems [2], in order to implement this concept. It can be said that the most important challenges in cloud computing are security and trust. Since the consumer's data has to be released to the cloud, the system requires high security and safety over them. The data in clouds could be very personal and sensitive and must not be unveiled to an unauthorized person. In cloud computing, data are threatened during the transition as well. This problem reduces the reliability of the cloud systems [3].

A popular protocol, which is used to exchange the data in cloud systems, is Simple Object Access Protocol (SOAP) [4] based on Extensible Markup Language (XML) [5]. Securing data in SOAP messages is one of the main concerns related to security in cloud systems. It can be threatened by XML Signature wrapping attack, which causes the unveiling of sensitive data [6]. This attack is based on altering the structure of the original message from the genuine sender. Although some remedies have been proposed to counter this attack (ID referencing and XPath methods), none of them has been able to counter the attack completely [6], as they sign a particular part of an XML document.

The solution provided in this research, uses a new method, namely SESoap, to provide integrity for the messages exchanged in a cloud system by SOAP. In this technique, which is less complicated, more reliable and faster than the ID referencing and XPath methods, the entire SOAP message is signed by XML digital signature, instead of signing a part of that. It also counters all known wrapping attacks and makes similar attacks impossible.

Layout of this article is as follows. In the next section, basic definitions and explanations related to SOAP message and XML signature are given. In the 3rd section, XML signature wrapping attack and its four different categories are explained briefly. The 4th section covers some of the previous researches, which are relevant to this topic. Proposing and describing the SESoap method, its analysis, and their results have been given in the 5th section. Finally brief conclusions and achievements of this research have been given in 6th section.

II. TECHNICAL BACKGROUND

A. SOAP Message

SOAP [7], is a protocol to provide communication between applications. It works as a format for sending messages via Internet and also collaborates with the firewalls [8], [9], [7].

1) SOAP Building Blocks: As it is also mentioned above, SOAP message's language is based on XML [8]. Moreover, it can be explained that the building block of SOAP is in fact a typical XML document, which consists of these items:

1) Envelope: this element recognizes the XML document as a SOAP message.
2) Header: this element includes the header information of a SOAP message.
3) Body: this element includes the actual SOAP message
4) Fault: Errors that occurred while processing message are included in this element [8], [10].

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

3) Skeleton of a SOAP Message: A typical skeleton of a SOAP message is shown in Fig.1.

```
<?xml version="1.0"?>
<soap:Envelope
xmlns:soap="http://www.w3.org/2001/12/soap-envelope"
soap:encodingStyle="http://www.w3.org/2001/12/soap-encoding">

<soap:Header>
...
</soap:Header>

<soap:Body>
...
  <soap:Fault>
  ...
  </soap:Fault>
</soap:Body>

</soap:Envelope>
```

Fig. 1. Skeleton of a SOAP message [10]

B. XML signature

XML signature is a technique, which is used to deliver reliability, integrity and message authentication, for various types of data [11]. By providing integrity to data, it is meant that once the data is signed; it cannot be altered later, without invalidating the signature. This technique is executed by employing asymmetric cryptography. The roles for signing a document are as follows [12].

$$M = D_{P_C}[E_{R_C}[M]] = D_{R_C}[E_{P_C}[M]]$$

In the formula, a message M is signed by private key and a public key is used to verify the signature. The reverse operation is allowed as well. Asymmetric encryption uses two keys in order to encrypt and decrypt a message, M, which are named private (Rc) and public (Pc) keys. XML digital signature employs private key and public key to sign a message and validate the document, respectively. When signing the message, signature will be attached to the original document, and will be sent to the receiver. It should be noted that the document, is not hidden, since hiding the message is not the aim of XML digital signature. Since asymmetric encryption is time consuming, a hash function (f (M)) is calculated over the document and the result, which is called digest value, is considerably smaller than the document itself. The result of hash function is then encrypted by private key. Consequently, the time passed for encrypting data is reduced significantly. Fig. 2 shows the structure of an XML signature.

```
<Signature>
  <SignedInfo>
    <CanonicalizationMethod />
    <SignatureMethod />
    <Reference>
       <Transforms>
       <DigestMethod>
       <DigestValue>
    </Reference>
    <Reference /> etc.
  </SignedInfo>
  <SignatureValue />
  <KeyInfo />
  <Object />
</Signature>
```

Fig. 2. Structure of an XML signature [12]

III. XML SIGNATURE WRAPPING ATTACK

XML signature wrapping attacks are possible because of the fact that the signature does not convey any information to where the referenced element is placed [13]. This attack was introduced for the first time, in 2005 by McIntosh and Austel [14], stating different kind of this attack, including Simple Context, Optional Element, Optional Element in security header (sibling value) and Namespace injection (Sibling order) [14]. This attack happens in SOAP message, which transfers the XML document, over the Internet.

A. Simple Ancestry Context Attack

In Simple Ancestry Context Attack, a request's SOAP body is signed by a signature, which is placed in the security header of the request. The recipient of the message, checks if the signature is correct and legalizes trust in the signing credential. Lastly, the recipient controls to realize whether the required element was actually signed, by bringing the "id" of the SOAP body to the ID reference, in the signature [15].

A typical example of this attack is shown in Fig. 3. The mechanism of this attack can be briefly explained in this way that, the SOAP body gets swapped with a malicious SOAP body. The original SOAP body is placed in a <wrapper> element, which is situated in the SOAP header and when the signature is validated, the XML signature confirmation algorithm, begins searching for the element, which has the id of "CMPE", as it is stated in the <Reference> element. Finally, <soap:Header> Element wrapped within the <wrapper> element, will be found by the algorithm. Signature verification will be implemented on the <soap:Header>, within the <wrapper> element. The verification will be positive, because it includes the original SOAP body, which is signed by the sender. The SOAP message will be passed to the logic of the application. In the application logic procedure, only the SOAP body, which is straightly positioned under the SOAP header, will be processed. In other words, all other SOAP body elements will be just ignored [15]. Fig. 3 shows how this attack works.

```
<soap:Envelope ...>
 <soap:Header>
  <wsse:Security>
    ...
    <ds:Signature>
      <ds:SignedInfo>
        ...
        <ds:Reference URI="#CMPE">
          ...
        </ds:Reference>
      </ds:SignedInfo>
        ...
      </ds:Signature>
    </wsse:Security>

        <Wrapper
     soap:mustUnderstand="0"
     soap:role=".../none" >
    <soap:Body wsu:Id="CMPE>
      <getQute Symbol="IBM">
    </soap:Body>
    </Wrapper>
  </soap:Header>
  <soap:Body wsu:Id="newCMPE">
    <getQuote Symbol="MBI"/>
  </soap:Body>
</soap:Envelope>
```

Fig. 3. Typical Simple Ancestry Context Attack [16]

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

B. Optional Element Context Attacks

In Optional Element Context Attacks, the signed data is contained in the SOAP header and it is arbitrary. Comparing this attack to the Simple Context Attack, which is explained above, reveals that the main problem is not the place of the signed data in the SOAP header [9]. In fact, the optional nature of signed data is the main issue [14]. The <ReplyTo> element, which specifies where to send the reply, can be given as an example, which is shown in Fig. 4. The mechanism of this attack can be explained as follows; it can be seen that the element of <wsa:ReplyTo> is placed in the <wrapper> element, while, the element of <wrapper> is also positioned underneath the <wsse:security>. In addition, by means of soap:mustUnderstand="0", in <wrapper>, this element has become optional and by using soap:role=".../none", it is destined that the SOAP node (application logic) should not process this header element. These modifications in the SOAP message, result in the <wsa:ReplyTo> to become completely disregarded by the application's logic. Having these attributions, when the signature gets legalized, the verification algorithm of XML signature begins to search for the element, which has the id of "theReplyTo" (specified in the <Reference>) and <wsa:ReplyTo>, which is in the <wrapper> element, will be found. At this stage, signature confirmation will be done on the <wsa:ReplyTo>, in the <wrapper>, and because it is including the original <wsa:ReplyTo>, signature confirmation will be positive. Consequently, SOAP message body and the descendants, which are understood, will be handed to the application logic while the <wrapper>, will not be passed to it. Thus, the application logic will ignore the <wsa:ReplyTo> element and as the result, the reply will not go to the address specified in <wsa:ReplyTo> and the original message sender will get the reply [9].

C. Sibling Value Context Attack

Sibling Value Context Attack covers the following scenario. In this attack, the security header includes a signed element, which is in fact an alternative sibling of <Signature>. A common model for this attack can be the element of <Timestamp>, which together with <Signature>, are direct descendants of SOAP security header. The difference between this attack and the previously discussed attacks is in the signed data, which in this attack is the sibling of <Signature> [16]. The main aim of this attack is to ignore the sibling of the signature element.

D. Sibling Order Context

According to McIntoch and Austel, 2005[14], this attack is dealing with the protection of the sibling elements that are individually signed.

Their semantics are related to their order relative to one another, from reordering by an adversary. More work is required to define appropriate countermeasures that do not prevent the addition and removal of siblings that do not impact the ordering semantics [14].

IV. Known Countermeasures to Wrapping Attacks

The requirements of a service-side security policy, in order to detect an attack were shown by McIntosh and Austell, 2005 [14]. These necessities are being improved by each attack, which is able to bypass the previous provided security policy. In continuance, some of the improvements in the policy will be explained.

```
<soap:Envelope ...>
 <soap:Header>
  <wsse: Security>
   ...
   <ds:Signature>
    <ds:SignedInfo>
     ...
     <ds:Reference URI="#CMPE">
      ...
     </ds:Reference>
     <ds:Reference URI="#theReplyTo">
      ...
     </ds:Reference>
    </ds:SignedInfo>
     ...
    </ds:Signature>
   </wsse:Security>

          <Wrapper
    soap:mustUnderstand="0"
    soap:role=".../none" >
   <wsa:ReplyTo wsu:Id="theReplyTo>
    <wsa:Address>http://cmpe.emu.edu.tr/</wsa:Address>
   </wsa:ReplyTo>
   </Wrapper>

 </soap:Header>
 <soap:Body wsu:Id="CMPE">
  <getQuote Symbol="IBM"/>
 </soap:Body>
</soap:Envelope>
```

Fig. 4. Typical Optional element context attack [14]

1) In the wsse:security header element, a signature "A" , XML signature, should be placed, having a clear soap:role attribute and value of "…/ultimateReceiver".

2) From signature "A", the element, identified by /soap:Envelope/soap:Body, must be referenced.

3) In the case of having any elements, which are matching with

/soap:envelop/soap:Header/wsse:Security[@role="…/ultimateReceiver"] wsu:Timestamp and /soap:Envelop/soap:Header/wsa:ReplyTo, it should be noted that these elements must be referred through an absolute path, Xpath expression, from signature "A".

4) Verification key of signature "A" must be issued and provided by a trusted Certificate Authorities (CAs) and the certificate of X.509v3, respectively [14].

The first example of XML signature wrapping attack, which was indicating that the controls suggested by McIntosh and Austell [14] are not satisfactory to notice XML signature wrapping attack, was shown by Gruschka and Lo Iacono, in 2009 [17]. It is also claimed in their research that the timestamp has to be referenced by an extra XPath expression, which is not fulfilled in Fig. 4. Although, it can be added easily, it should be noted that the XPath references result in further problems. It is known that XPath expressions are more

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

difficult to be evaluated, comparing to IDs, this issue is especially important in the context of streaming SOAP message. Another more important issue is that employment of XPath references may indicate security issues, so they are not suggested by basic security profile [6].

In a new method [18], which was proposed in 2006 and is named as inline method, a new element called SOAP account was introduced. Some characteristic information are gathered together and inserted in the SOAP account element [17]. Protection of some key features of SOAP message structure is aimed in this technique. The properties, which are aimed to be protected, are listed as below.

1) Number of header element descendants
2) Number of soap:envelop, descendent elements
3) Amount of references in every signature
4) The descendants and antecedents of every signed item

By means of this approach, with the above properties, if in an attack, each of these properties is changed, the attack will be easily identified [18].

The main problem with this method is that it does not provide a general protection, from XML signature wrapping attack. In other words, if an attacker manages to change the SOAP message structure in a way that the inline method structure properties does not get changed, this technique can be easily dodged [19].

In addition, fastXPath method was proposed by Gajek et al., in 2009 [9]. This method is employed to increase the speed of XPath function, and to point to the signed subtree. However, this method also could not solve the identified issues about XPath expression [20]. A comparison between runtime of different methods, ID, fastXPath and XPath methods, have been also done in their investigation. The comparison's relevant graph is shown in Fig. 5.

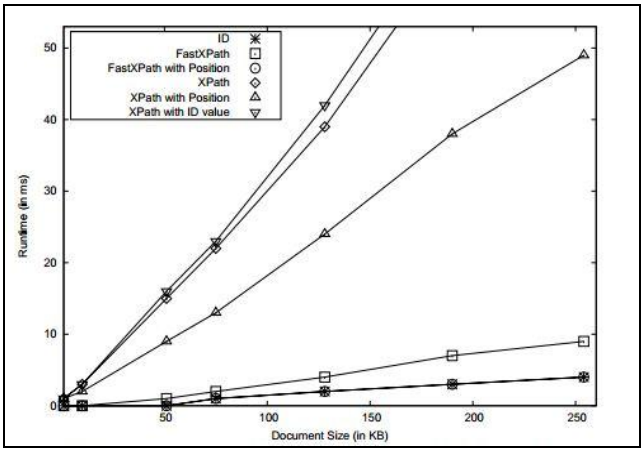


Fig. 5. Runtime comparison of different referencing methods [17]

V. SIGNING ENTIRE SOAP (SESOAP) METHOD

Since most of the XML signature wrapping attacks are done through changing the structure of the original SOAP message, sent by the genuine sender [17], it is logical to propose a protecting method, which aims to protect the structure of the sent message, from attacker. To fulfil this aim, the digital signature can be used to guarantee the integrity of message.

The method of this paper, i.e. Signing Entire SOAP (SESoap) method, is to apply the digital signature structure over entire SOAP envelop element, which results in securing the whole document. Consequently, an attacker will not be able to change the location of elements or remove or add any element to the original document. In the case of modification in any part of the document, the signature cannot be verified. The skeleton of SESoap method is shown in Fig. 7.

It should be noted that the element of SOAP:signature, contains the result of signing the entire content of soap:envelop, except the element of soap:signature itself. To explain better, the structure of SOAP after applying the SESoap method is shown in Fig. 6.

A. Simple Element Context Attack Countering

In simple Context attack, a wrapper alters the location of the Soap body and adds a new Soap body to threaten the SOAP document [14]. It is quite clear that by using digital signature over entire document, any alteration or adding any element to the signed document will be totally prevented.

```
<soap:envelope>
        <soap:header>
        </soap:header>

        <soap:body>
        </soap:body>

        <soap:signature>
        </soap:signature>
</soap:envelope>
```

Fig. 6. Skeleton of SESoap method

B. Optional Element Context Attack Countering

In Optional Element Context attack, a wrapper adds some information to optional element to application logic of a program could not parse that element [14]. Again, the same as the previous attack, when a wrapper tends to add something to the document, the attack is prevented by SESoap.

C. Sibling Value Context Attack Countering

The two previous types of attacks are possible to be prevented by means of XPath method [14]; however XPath is susceptible against this attack [6]. As it has been explained in the previous section, Time stamp element, which is an optional sibling element of signature element, can be threatened by wrapper. But for wrapping on this element, the wrapper again must modify some parts of document [14]. Consequently, as modifications are prevented in SESoap method, Sibling Value Context attack will not be allowed to occur.

D. Sibling Order Attack Countering

This attack relies on changing the order of individual sibling elements [14]. Therefore, since reordering is also not

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

possible in SESoap, again no wrapper can be successful in implementation of this attack.

E. Conducted Experiments

SESoap method has been implemented by using C#.net, in order to determine how fast it is, comparing to the previous methods of ID referencing and XPath. These examinations have been performed by means of Laptop, having 2.00 GHz Core2Duo CPU, and 1.00 GB memory, in Windows7 operating system. The time for finding the element in SESoap is zero, because this technique does not search for any specified element inside the SOAP document. Experiments were conducted on file sizes used in [17] and also on more than ten times greater size (up to 3.15 MB). The graph for comparing these time durations is shown in Fig. 7.

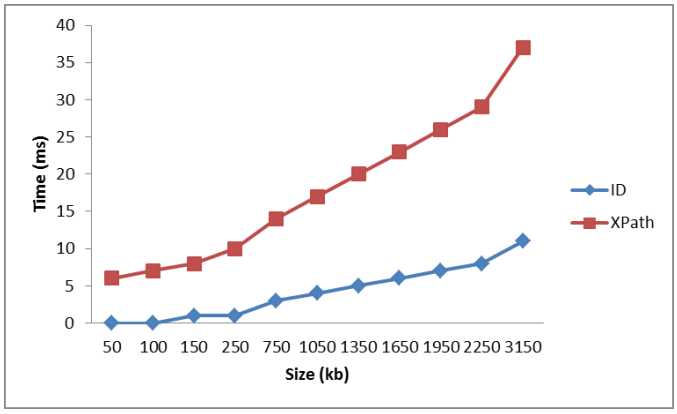


Fig. 7. Time durations of the ID and XPath methods

In the next step, the time durations for hashing the specified element inside the SOAP document, have been estimated. Fig. 8 shows the result of the hashing specified element.

In addition, the consumed time for encrypting data, in all the three methods are the same, because in the digital signature, encryption function applies on the signed info element of signature. The sizes of the signed info element in all the methods are equal. As the result, the consumed times for encrypting the signed info elements are the same. In this study the time consumed for all three methods was 3.0004 milliseconds. In this study, two codes have been used to measure the time, in Code1 each function (Finding element, hash function and encryption function) has been done separately and in Code2 the whole operations have been done as one component. The total times consumed to sign the soap message in each of the three methods, using Code1, are shown in Fig. 9 [21].

According to these results, ID is faster than XPath, in finding an element. On the other hand, ID and XPath methods are faster, comparing to SESoap method, in hashing the specified element. Moreover, as the numbers show, the total consumed time to sign a SOAP document by SESoap method is approximately three times faster than the XPath method and even a little faster than ID. Consequently, it can be claimed that, the SESoap method is operating more sufficiently, than the other two methods, considering both aspects of security and time.

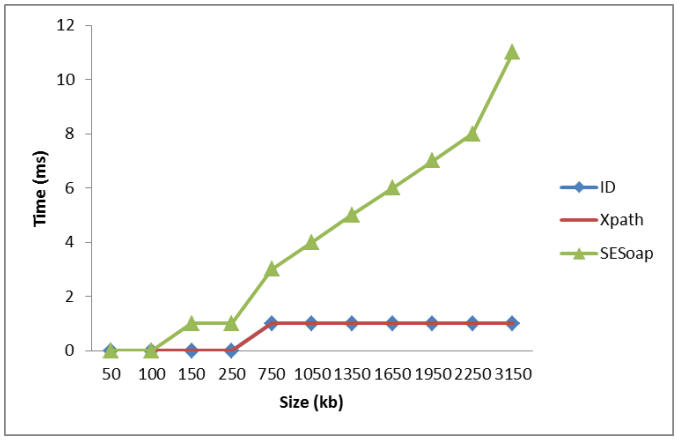


Fig. 8. Time durations for hashing the specified element

Moreover, the total time durations in order to sign soap message, using Code 2, is shown in Fig. 10.

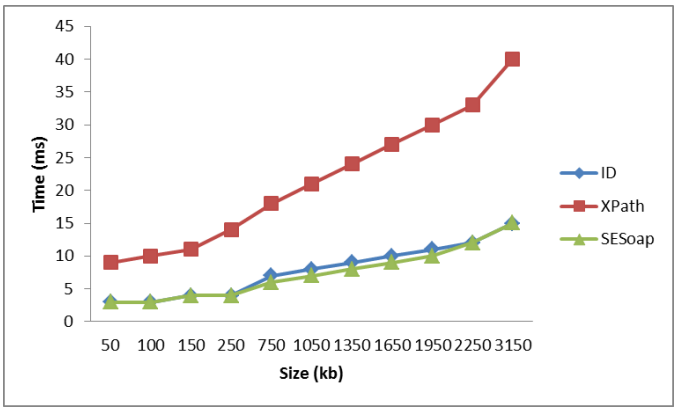


Fig. 9 Total time durations consumed to sign the soap message, using Code 1

These results are more complying with the previous research [17], but as it can be obviously noticed, the results of that research are less efficient than what is done in this study.

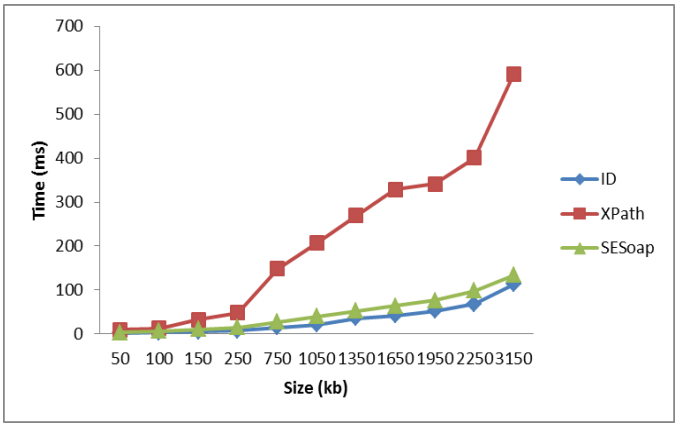


Fig. 10 Total time durations consumed to sign the SOAP message, using Code 2

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

VI. Conclusion

The primary goal of this study was to secure SOAP message, which is employed to exchange information between web applications of cloud computing systems. Having this aim, a new method, SESoap, has been proposed. The concept of this method is using Digital Signature technique to immune the information inside a SOAP message from modification by an adversary.

The results obtained from implementation of SESoap method indicate that this method is slower than the other examined methods, for hashing the information. The reason of this observation is that, comparing to the other examined methods, in this method, the hash function is applied over a greater size of data. On the other hand, for finding element in SOAP message, SESoap does not consume any time and the total time duration for signing the message, is approximately three times faster than the XPath method and even a little faster than ID.

References

[1] P. Mell and T. Grance, "COMPUTER SECURITY," September 2011. [Online]. Available: http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf. [Accessed April 2013].

[2] S. Bhayal, "A study of security in cloud computing," ProQuest Dissertation Publishing, 2011.

[3] D. Jamil and H. Zaki, "Security Issues in Cloud Computing and Countermeasures," International Journal of Engineering Science and Technology, vol. 3, no. 4, pp. 2672-2676, 2011.

[4] W3C, "SOAP Specifications," W3C, 2007. [Online]. Available: http://www.w3.org/TR/soap/. [Accessed April 2013].

[5] W3C, "Extensible Markup Language (XML)," W3C, 2012. [Online]. Available: http://www.w3.org/XML/. [Accessed April 2013].

[6] N. Gruschka and L. Lo Iacono, "Vulnerable Cloud: SOAP message security validation revisited," in 2009 IEEE International Conference on Web Services, Los Angeles, 2009.

[7] "SOAP – Simple Object Access Protocol for Message Negotiating and Transmission," The Two Way Web, [Online]. Available: http://www.thetwowayweb.com/soapmeetsrss. [Accessed May 2013].

[8] "SOAP (Simple Object Access Protocol) Definition," 2013. [Online]. Available: http://www.techterms.com/definition/soap. [Accessed May 2013].

[9] W3Schools, "SOAP Introduction," W3Schools, [Online]. Available: http://www.w3schools.com/soap/soap_intro.asp. [Accessed May 2013].

[10] w3schools, "SOAP Syntax- SOAP Building Blocks," 2013. [Online]. Available: http://www.w3schools.com/soap/soap_syntax.asp. [Accessed May 2013].

[11] R. Salz, "Understanding XML Digital Signature," July 2003. [Online]. Available: http://msdn.microsoft.com/en-us/library/ms996502.aspx. [Accessed May 2013].

[12] M. Bartel, J. Boyer, B. Fox, B. LaMacchia and E. Simon, "XML Signature Syntax and Processing Version 1.1," World Wide Web Consortium (W3C), April 2013. [Online]. Available: http://www.w3.org/TR/2013/REC-xmldsig-core1-20130411/. [Accessed April 2013].

[13] C. O. h. Eigeartaigh, "Open Source Security: XML Signature Wrapping attacks on Web Services," October 2012. [Online]. Available: http://coheigea.blogspot.com/2012/10/xml-signature-wrapping-attacks-on-web.html. [Accessed May 2013].

[14] M. McIntosh and P. Austel, "XML Signature Element Wrapping Attacks and Countermeasures," IBM Research Report, NewYork, 2005.

[15] "XML Signature Wrapping - Simple Context -," WS-Attacks.org, 2011. [Online]. Available: http://clawslab.nds.rub.de/wiki/index.php/XML_Signature_Wrapping_-_Simple_Context. [Accessed April 2013].

[16] "XML Signature Wrapping - Optional Element in Security Header -," WS-Attacks.org, 2010. [Online]. Available: http://clawslab.nds.rub.de/wiki/index.php/XML_Signature_Wrapping_-_Optional_Element_in_Security_Header. [Accessed May 2013].

[17] C. Mainka, M. Jensen, L. I. Luigi and J. Schwenk, "XSPRES: ROBUST AND EFFECTIVE XML SIGNATURES FOR WEB SERVICES," in 2nd International Conference on Cloud Computing and Services Science, Porto, 2012.

[18] M. A. Rahaman, A. Schaad and M. Rits, "Towards secure SOAP message exchange in a SOA," in 3rd ACM workshop on Secure web services, New York, 2006.

[19] S. Gajek, L. Liao and J. Schwenk, "Breaking and fixing the inline approach," in ACM workshop on Secure web services, New York, 2007.

[20] S. Gajek, M. Jensen, L. Liao and J. Schwenk, "Analysis of Signature Wrapping Attacks and Countermeasures," in ICWS 2009. IEEE International Conference on Web Services, Bochum, 2009.

[21] Microsoft, "XmlDsigXPathTransform Class (System.Security.Cryptography.Xml)," Microsoft, 2013. [Online]. Available: http://msdn.microsoft.com/en-us/library/system.security.cryptography.xml.xmldsigxpathtransform(v=vs.85).aspx. [Accessed 30 June 2013].

An Approach for Software Security Evaluation Based on ISO/IEC 15408 in the ISMS Implementation

Tahereh Nayerifard*
Computer Engineering Department
Islamic Azad University, North Tehran Branch
Tehran, Iran
tahere.nayerifard@gmail.com

Nasser Modiri
Computer Engineering Department
Islamic Azad University, Zanjan Branch
Zanjan, Iran
nassermodiri@yahoo.com

Sam Jabbehdari
Computer Engineering Department
Islamic Azad University, North Tehran Branch
Tehran, Iran
s_jabbehdari@iau-tnb.ac.ir

***Abstract*—Security software is focused on identifying potential hazards and can have a negative impact on the software and also damage the whole system. If risks are identified early in the software engineering process, Software design problems are detected, and the potential hazards are eliminated or controlled.**

Value of the investment on hardware components and software programs, the value of data organization, individual data values, threats, computer crimes, are the main reasons to understand the Importance of security and why security measures are necessary. Since the systems are under constant threat and on the other hand, absolute security cannot be seen, it is obvious that whenever there is a security problem of the advancement of technology. From Hence, in order for raising the level security in the software, at all stages of the development of software products, security assessments should be considered.

In this paper, we tried to security evaluate all the activities of Software Development Life Cycle based on the third part of the ISO/IEC 15048, to increase the level of security in the SDLC. In fact, using this standard, the adoption of security activities in order to assess the life cycle activities is proposed. Continued research in applying the principles of ISMS, security assessment activities have improved with exposure in PDCA cycle, thus the complete security evaluation on the life cycle of software development activities will be carried out. Therefore, the goal is to create a method based on the principles of safety engineering, that represent the evaluation of the activities involved SDLC under the Common Criteria standard. Since the guidelines of the standards, ISO/IEC 12207, ISO/IEC 15408 and ISO/IEC 27034 is used, this approach worked quite flexible and adaptable to changing technology, organizational structure, changing business objectives and organization security policy changes.

Keywords- ISO/IEC 15408 ،PDCA ،SDLC ،Security Evaluation ،Software Security

I. INTRODUCTION

The importance of software security that is felt when this system failure led to loss of lives and property. Thus, the reliability of these systems is essential. To ensure the reliability, evaluating and measuring are required in all phases of software development and then, to know how to improve the product quality and security. Assessment and, in particular, evaluate the security is an issue that is very important. Regardless of the limitations, evaluation is an integral part of the software development and has been widely deployed in every phase of the SDLC [1]. The software does not work correctly; it can have been devastating effects on an organization. Without evaluating the software, cannot be aware of problems of interoperability, quality and security of the information. This has led to many problems, including as following:

a) Losing Time: The reason for this is that the transaction can take a long time to process, and can be an employee who is unable to work due to an error or deficiencies.

b) Capital Loss: This could include the loss of customers' rights that are due to non-compliance with legal requirements resulting financial penalties.

c) Damage to business reputation: if an organization due to software problems, not able to provide services to its clients, customers lose trust and faith in the organization.

d) Injury or death: the issue of safety critical systems, if not working properly can cause injury or death (e.g. the flight software in traffic control.

II. SOFTWARE SECURITY ISSUE

More research on software usability and security, rather than focusing on the issues of users and their needs, while producing secure software, vendors and developers should also be considered. However, according to published reports, as shown in Figure 1, the number of security vulnerabilities found in various applications, is increasing every day, and this represents a weakness in developing process of the software.

* Corresponding author

[1] Software Development Life Cycle

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

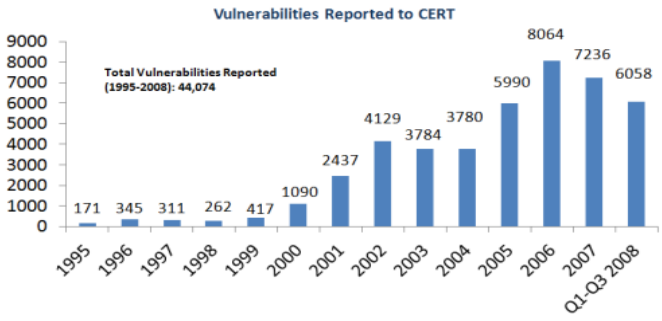


Figure 1. Statistics software vulnerabilities reported to the CERT center until late 2008 [1]

On the other hand, the expansion of the software used in sensitive environments and confidence in their work, loss or damage resulting from any security vulnerability, for customers and users and in return for software developers will have a huge cost. Clearly, those software developers have a key role to play in the creation of security. In many cases, the lack of familiarity of developers with the project security fields, the product will cause serious vulnerabilities.

With the development of information and communication systems, attacks and security threats against such systems have also increased and today, considering the security aspects in system development is one of the key issues. The role of information technology in human life is efficient only if the security of this area be assurance. IT security failures not only reduce today's technology confidence, but also it will be a conversion factor to threat and economic and social human catastrophe problems. The other important issue, satisfy customers and employers to payment the costs of security implementing and maintaining the system in secure mode. In this regard, the approach that considers the prerequisites for achieving safety and security requirements from the beginning, and standards-based moves, can help reduce the concerns of employers investing style. Therefore, if we can solve the security problems from the outset, the security problems in the final stages will be lower and will be removed or controlled with very low cost.

III. SOFTWARE ENGINEERING, QUALITY, SECURITY AND THEIR RELATIONSHIP

The two non-functional capabilities of the software, is software quality and security. Although it is true that security is usually regarded as an important indicator of the reliability of the software, however, achieving security in software systems, according to the centralized of the information, data and systems, it is necessary that the software quality be observed at first. To achieve this non-functional objective, it is essential that by the structures, models or frameworks, we have a broad and comprehensive view of software quality. Clearly, software quality that has such a large scale couldn't be controlled easily and requires that we have a proper definition of the subject and take advantage of known standards and methods of producing secure software development.

Developing software without a reliable framework and without proper processes and activities, it is impossible to achieve quality. Hence to get the security evaluation, the software development activities by software engineering team, using Software engineering standards and methods, measurement quality activities related to software development activities undertaken in each phase of SDLC by quality team, using standard and methods known in the context of quality, and Finally, the security evaluation activities over the two groups, using security standards, will help producers and owners of the software to gain secure security evaluation. Figure 2 shows the dependence of the activities of software development, quality and security.

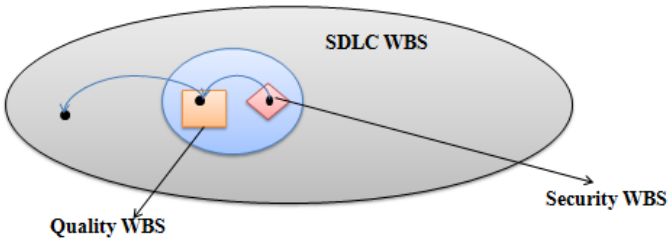


Figure 2. Dependence Work Breakdown Structure of SDLC, Quality and Security

A. *Security in SDLC*

As in the world of engineering to make each product, there are some stages, software engineering is no exception to this rule. In other words, in order to make software products, there is a cyclical process of software development is divided into stages regulated. The life cycle of software development, each stage is defined as activities that must be performed at each step. In this life cycle, inputs and outputs that must be given to each section, and it should be quite are clear and for each step, the control points are defined as the amount by which progress will be determined in terms of quantity and quality.

SDLC is a systematic approach to the creation of software or application. This cycle typically includes requirements, analysis, design, coding, test, implementation and post-implementation phases. Application development refers to a software development process used by an application developer to build application systems. This process is commonly known as the Software Development Life Cycle methodology and encompasses all activities to develop an application system and put it into production, including requirements gathering, analysis, and design, construction, implementation, and maintenance stages. Examples of the SDLC methodology include e.g., waterfall, iterative, rapid, spiral, RAD, Xtreme and many more. [2, 3]

A SDLC is a well-defined, disciplined, and standard approach used in developing applications, which provides:

- A methodical approach to solving business and information technology problems
- A means of managing, directing, monitoring and controlling the process of application/software building, including:
  - A description of the process - steps to be followed

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

- Deliverables reports/programs/documentation/etc

Benefits of using a SDLC methodology include:

- Has a proven framework
  - Consistency and uniformity - methods and functions
  - Results/Deliverables
- Facilitates information exchange
- Defines and focuses on roles and responsibilities
- Has a predefined level of precision to facilitate a complete, correct and predictable solution.
- Enforces planning and control

ISO/IEC 12207 is a standard that establishes a common framework for SDLC. Concepts from the standard ISO/IEC 12207 can help the software director and the business in general to achieve greater success with their employees.

In the computer world, there are high volumes of attacks, threats, hazards and enemies. Despite the wide range of hazards in the computer field, this shows the importance of software engineering should also be considered in terms of security. In fact, we must understand the threats and to deal with them should be invested on software engineering. The important point here is, this is a security issue that must be addressed in the context of software engineering. The reason is that the security problem is continuous and permanent and addressing it in the software engineering causes this process can be continuously tracked and managed.

Most developers are not sensitive to the threat or application software designs and consider security as a Solution after completing the design and construction the software. In order to avoid security problems and for dealing with security threats and attacks, should be prevented from cross-sectional dispersion Proceedings and actions in this area should be structured, which is discussed in the context of software engineering. So, to create the desired level of security and defies the threats in the software during its life cycle, security actions must be implemented in software engineering and for software development always should step in the software engineering framework.

IV. ISO/IEC 15048 PHILOSOPHY

The CC philosophy is that the threats to security and organizational security policy commitments should be clearly articulated and the proposed security measures should be demonstrably sufficient for their intended purposes. Furthermore, those measures should be adopted that reduce the likelihood of vulnerabilities, the ability to exercise (i.e. intentionally exploit or unintentionally trigger) a vulnerability, and the extent of the damage that could occur from a vulnerability being exercised. Additionally, measures should be adopted that facilitate the subsequent identification of vulnerability and the elimination, mitigation, and/or notification that vulnerability has been exploited or triggered [4].

To show requirements and ensure IT security operations, under the standard ISO/IEC 15408 the following two concepts are used [5]:

a) Protection Profile infrastructure (PP)

The PP allows collecting and implementation completeness and reusability security requirement. PP can be use by a customer for detecting and realizing secure product, which meets their needs.

b) Security Target infrastructure (ST)

The ST shows security requirement and secure operation for evaluation system or special product, which is called TOE[2], ST is a base for evaluation according to the standard ISO/IEC 15408 and use with who evaluate on TOE.

The main concept of protection profiles (PP), packages of security requirements and the topic of conformance are specified and the consequences of evaluation; evaluation results are described. This part of the CC gives guidelines for the specification of Security Targets (ST) and provides a description of the organization of components throughout the model.

V. ISMS[3] AND PDCA CYCLE

Plan/Do/Check/Act Cycle was established by Japanese in 1951 based on doming cycle. This cycle consists of four stages: Plan: determining of objectives and required process for presentation of results according to customer's requests and or organization policies. Do: implementation. Check: monitoring and measurement of process and product according to policies, objectives and requirements or request related to product and reporting of results. Act: doing activities in order to improve process performance. This cycle is based on scientific methods, and feedback plays a basic role in that so the main principle of this systematic method is iteration. When a hypothesis is being denied the next execution of the cycle can expand knowledge, and these iterations make become closer to the aim. The PDCA Cycle is the underlying method/strategy which underpins the ISO/IEC 27001 approach. PDCA is the core to the ISO/IEC 27001 implementation of ISMS (Information Security Management System), and is documented within the standard itself [6].

VI. THE PROPOSED APPROACH

The results of studies carried to indicate that there are conditions that must be established in order to achieve the desired level of success in evaluating security. In conclusion, this can be seen as software engineering and software quality. The proposed approach in this study is to evaluate security software based on the activities of each phase. In fact, the control points for security evaluation will not be at the end, or the parts of a phase. However any of the activities associated with each phase, after assessing the quality activities done on it, the security will be evaluated. Figure 3 illustrates the security evaluation approach is proposed.

[2] Target of Evaluation

[3] Information Security Management System

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

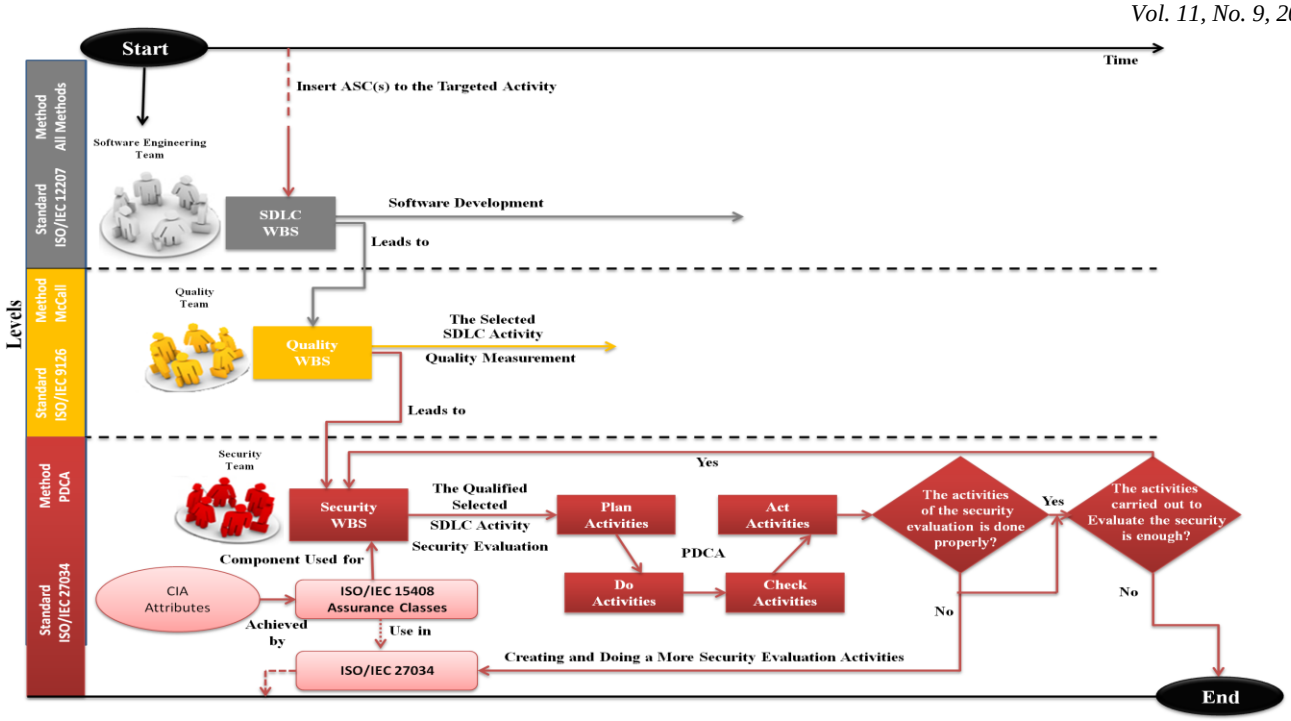


Figure 3. Security evaluation of the proposed approach based on ISMS

After the PDCA cycle and security feedback activities, investigate whether evaluations are appropriate and adequate. If the answer is negative, further activities generate using the CC and Security WBS will be completed, and security evaluation process will be carried out on the related activity of SDLC. This will be done so that the security evaluation provides the desired results. The completion of the security evaluation activities will continue until the software is being developed. Complete security evaluation, and identifies these activities associated with each of the SDLC activities will be part of the organization capital and is a very valuable.

ISO/IEC 27034, series of standards known as ISMS family, provides guidance to assist organizations in integrating security into the processes used for managing their applications. The purpose of ISO/IEC 27034 is to assist organizations in integrating security seamlessly throughout the life cycle of their applications [7]. This standard can be placed between the third and first level of the proposed approach and the CC. Thus, in addition of using of security assurance classes and components from CC to evaluate the targeted SDLC activity (our TOE in this approach), these components have the ability to be implemented as Application Security Controls (ASC) [7] on the targeted SDLC activity using ISO / IEC 27034. Therefore, in addition to security assessment, security controls imposed on SDLC activities, therefore these activities can be improved and so will be led to secure activities in the software development life cycle.

A. *Features of the proposed approach*

1) Reduce costs and damages resulting from security problems: This approach is recommended emphatically for software which security issue is very important. Because it will prevent excessive of costs and damages consequent duplication due to lack of adequate security acceptance. Although security assessments for each of the SDLC activities can be costly, but the full set of security evaluation activities after the development of a software, will return this cost. Because y take advantage of the activities in the development of future applications is also possible.

2) Obtain a valid security certificate: CC is the security standards for assessment and assurance of security. An important feature of this standard is certificating the products for business security levels defined in this standard that has the important global status. One of the reasons it was chosen as the security standard in proposed approach is that will be able to get a valid security certificate.

3) Improve performed security evaluation using PDCA Cycle: Implement according to the principles of ISMS, provides the ability to receive feedback from previous security assessments and thus to improve results and to reach appropriate conditions, more security evaluation activities will be selected and executed. Furthermore, it causes the security evaluation activities to be completed and become assets for the organization.

VII. CONCLUSION

Our goal in this paper is a comprehensive survey of the major features of the software, in order to secure and proper security evaluation. To enhance the reliability and availability of this approach on the level of commercial and military, etc. and to minimize the possibility of error, it used the international standards to promote for the intended purpose. In fact, the procedure is establishing a coherent relationship

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

between standards and the proper use of them in order to utilize the characteristics of each of these standards and enhance their effectiveness by combining them.

This approach attempt is to secure software development life cycle activities based on international standards. As a result, the security level of the SDLC activities will be determined and consequently, we have eliminated or reduced the security problems in these activities. Also, due to the resolution of security problems based on the proposed cycle, in the subsequent periods of security assessments, fewer security evaluation activities will be required, thus this reduces the time and cost and all activities carried out over time to produce and develop the software will be safe.

REFERENCE

[1] http://www.cert.org last visited on 27/08/2013

[2] ISO/IEC 12207, Systems and software engineering -- Software life cycle processes, 2008

[3] Application Development Standard: Standards for SDLC, Government of Ontario IT Standard (GO-ITS), Document Number: 54, V1.0, 2007

[4] ISO/IEC 15408, Information technology -- Security techniques -- Evaluation criteria for IT security – Part 3: Security assurance components, 2008

[5] ISO/IEC 15408, Information technology -- Security techniques -- Evaluation criteria for IT security – Part 1: Introduction and general model, 2009

[6] ISO/IEC 27001, Information technology -- Security techniques -- Information security management systems -- Requirements, 2008

[7] ISO/IEC 27034, Information technology -- Security techniques -- Application security -- Part 1: Overview and concepts, 2011

AUTHORS PROFILE

Tahereh Nayerifard received her B.Sc Degree in Computer Engineering from Islamic Azad University (Hamadan Branch), Hamadan, Iran in the year 2009. Currently she is pursuing M.Sc in Computer Engineering from Islamic Azad University (Tehran North Branch),Tehran, Iran under the guidance of Dr Modiri. She is presently working on Software Security Solutions.

Dr. Nasser Modiri received his M.S. Degree from the University Of Southampton, U.K, and Ph. D. degree from the University of Sussex, U.K. in 1986 and 1989, respectively. In 1988 he joined The Networking Centre of Hemel Hempstead, and in 1989 he worked as a Principal Engineer at System Telephone Company (STC) Telecommunications Systems, U.K. Currently, Dr. Modiri is the president of Ayandehgan Rayaneh Co. developing web-based software and designer and implementer of information technologies services for Intranet networks while teaching actively MSc courses in network designing, software engineering and undertaking many MSc projects. He is currently developing applications for Virtual Universities, Virtual Parliaments, Virtual Organizations, ERP, GPS+GSM, GPRS, RFID, ISO/IEC 27000, ISO/IEC 15408 technologies.

Dr. Sam Jabbehdari currently working as an assistant professor at the department of Computer Engineering in IAU (Islamic Azad University), North Tehran Branch, in Tehran, since 1993. He received his both B.Sc. and M.S. degrees in Electrical Engineering Telecommunication from K.N.T (Khajeh Nasir Toosi) University of Technology, and IAU, South Tehran branch in Tehran, Iran, in 1988, through 1991 respectively. He was honored Ph.D. Degree in Computer Engineering from IAU, Science and Research Branch, Tehran, Iran in 2005. He was Head of Postgraduate Computer Engineering Department IAUNorth Tehran Branch during 2008-2010. He also has written “Advanced Topics in Computer Networks” book in Persian Language (Tehran, Classic, 2009). His current research interests are Scheduling, QoS, MANETs, Wireless Sensor Networks and Grid Computing Systems.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

An Efficient Parallel Based Diabetic Retinopathy Grading Algorithm

Ass. Prof. Ghada F. Elkabbany
Informatics Dept., Electronics Research Institute
Cairo, Egypt

Ass. Prof. Heba A. Elnemr
Computers & Systems Dept., Electronics Research Institute
Cairo, Egypt

***Abstract*—Medical imaging has revolutionized the medicine by providing cost-efficient healthcare and effective diagnosis in all major disease areas. Diabetes is a chronic disease and a major public health challenge worldwide. Diabetes complications can be prevented or delayed by early identification of people at risk. There are several approaches carried out on this context. There are many methods are available for prediction but because natural process of this kind are very complex which involves large number of input variables so we need very large dataset for proper prediction, it also has disadvantage of high algorithmic complexity and extensive memory requirements of the required quadratic programming in large-scale tasks. For very large and complex problems it is better to divide data in parts which not only decrease the complexity but also provide the capability of handling the tasks in parallel. This work presents and evaluates a method for introducing parallelism into the diabetic retinopathy grading algorithm proposed in [1]. The aim is to improve its performance by utilizing parallel concepts which distribute the employed datasets into different nodes which reduces the computational complexity, processing power and memory requirements. To implement the parallel processing on DR grading algorithm presented in [1], different levels of parallelism are used. Multi-level of parallelization improves the system utilization and throughput. In the proposed parallel DR grading algorithm, when the number of nodes is large load imbalance occurs. Thus, static load balancing algorithm is applied to get better performance. The suggested parallel DR grading method is simple and can be used for large datasets. This method also provides the flexibility to be modified according to the dataset size, number of nodes and memory available on different units. We have tested the proposed algorithm and the results are very encouraging.**

Keywords- Diabetic retinopathy; Clustering; Parallel processing; Texture feature extraction; Gray level co-occurrence matrix; Parallel techniques

I. INTRODUCTION

Medical image processing has become an applied research area and has been an interdisciplinary research field attracting expertise from applied mathematics, computer sciences, engineering, statistics, physics, biology and medicine. There are so many different medical image modalities presented like CT, PET, MRI etc. These Modalities are having different characteristics and used as per requirements. The aim of digital medical image processing is to improve the pictorial information in order to perform subsequently other tasks such as classification, feature extraction or pattern recognition. Since the size of these images are very large, the analysis of these modalities take so much time to process sequentially and thus give result after some time. So if we divide this sequentially processing to efficient parallel processing then we can find good results in very reasonable time. Hence, time and/or money can be saved using parallel computing. Furthermore, parallel computing provides concurrency and by this we can use non-local recourses very efficiently. It also removes the limit of serial computing.

This work presents a framework of diabetic retinopathy (DR) grading [1] using parallel computing in order to reduce the execution time. DR is an eye problem that can cause blindness. It occurs when high blood sugar damages small blood vessels in the back of the eye, called the retina. The early detection of stages of DR will be highly beneficial in effectively controlling the progress of the disease [2]. For the past several years, many automatic detection and grading of DR techniques have been developed and discussed [3] and [4]. However, most of the methods are only of theoretical interest because the time complexity of these methods is too high for realistic handling of huge amounts of existing medical images.

The rest of the paper is organized as follows: Section 2 presents the related work, Section 3 introduces a brief description of parallel computing systems. Section 4 describes the proposed parallel DR grading algorithm. In section 5 the discussion of results over an extensive datasets are presented. Finally, the conclusion is discussed in section 6.

II. RELATED WORK

Digital processing of medical images has helped physicians and patients during past years by allowing examination and diagnosis on a very precise level. Nowadays possibly the most important support that can be offered for modern healthcare is the use of high performance computing architectures to analyze the huge amounts of data that can be collected by modern acquisition devices. As we can see that medical imaging requires lots of memory space and time to process so by utilizing parallel techniques we can find efficient and fast result. The main idea of parallel image processing is to divide the problem into simple tasks and solve them concurrently, in such a way the total time can be divided between the total tasks (in the best case) [5]. Sanjay *et al.*, present parallel implementation of different sequential image processing algorithm utilizing current multi-core architectures available in commercial processors. The focus of this implementation was to improve the performance of

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

segmentation, denoising, histogram processing. Another survey has been done by Y. Kadah et al. [6], this special issue contains eleven papers covering various imaging modalities including MRI, CT, X-ray, US, and optical tomography. The papers demonstrated the potential of parallel computation in medical imaging and visualization in a wide range of applications including image reconstruction, image denoising, motion estimation, deformable registration, diffeomorphic mapping, and modeling. Shrivastava et. al [7] proposes a parallel SVM to correctly predict the future possibility of diabetes for any person based on a survey dataset which relates the different body parameters with diabetic and non diabetic persons. The parallel SVM is utilized to distribute the survey datasets into n different sets for n different machines which reduces the computational complexity, processing power and memory requirements for each machine. Texture features extraction methods are key functions in various image processing applications such as medical images, remote sensing, and content-based image retrieval. The most common way to extract texture features is the use of Gray Level Co-occurrence Matrix (GLCM). The GLCM contains the second-order statistical information of spatial relationship of the pixels of an image. However, the computation of both GLCM and extracting texture features are very time consuming. Many researchers have been working on accelerating the process of computation the GLCMs and texture features extraction algorithms on FPGAs platforms [8]-[11]. As shown in **[**10**]** the implementation on the FPGAs had some drawbacks: first, some implementations required large external memory banks, while some processing is performed by a host machine. Second, other implementations included symmetry and sparseness matrices, which is not a general implementation to support all kinds of images. Finally, these implementations calculate GLCMs without implementation considerations for improving the performance of the Haralick texture features. Additionally, some of them used small image sizes. Some researchers displayed the Cell [11]-[14] and Graphics Processing Units (GPUs) [15]. Sugano and Miyamoto [14] have implemented good feature extraction method for tracking on the Cell processor. While Gipp et al. [15] accelerated the computation of the GLCMs and Haralick texture features using GPUs for biological applications. In this paper, we have chosen the cluster computing platform for our parallel implementation and performance improvement. The reasons behind this selection compared to the FPGAs, Cell and GPUs are as follows. First, it is available and not expensive. There are many experiments in the implementation of different applications on this platform. Second, the communication times are not significant. Finally, it is hardware independent.

This paper presents a novel method for fast parallel computation of DR grading algorithm presented in [1]. The aim is to improve its performance by utilizing parallel concepts which distribute the employed datasets into different nodes which reduces the computational complexity, processing power and memory requirements. In the proposed parallel DR grading algorithm, when the number of nodes is large load imbalance occurs. Thus, static load balancing algorithm is applied to get better performance. Load balancing is dividing the amount of work that a node has to do between two or more nodes so that more work gets done in the same amount of time and, in general, the application is computed faster. The proposed parallel DR grading method is simple and can be used for large datasets. This method also provides the flexibility to be modified according to the dataset size, number of nodes and memory available on different units.

III. PARALLEL COMPUTING

The computing power required by applications is increasing at a tremendous rate. Hence, the researchers have therefore been towards devising ever more powerful computer systems to handle these complex problems. The usage of parallel computing is one of the most promising means by which we can bridge the gap between needs and available resources. Architecture of parallel systems is broadly divided into two categories: shared memory and distributed memory. Shared memory (tightly coupled) systems use a common or global memory shared by various processors and have centralized control. On the other hand, distributed memory (loosely coupled or local memory) systems, involves connecting multiple independent processing elements (PEs) each contains a processor and its local memory. There is no sharing of primary memory, but each processor has its own memory. While multi-threads programming is used for parallelism on shared memory systems, the typical programming model on distributed memory system is message passing. Clusters are considered to be mixed configuration of both shared memory and distributed memory. [16] and [17]. That is to say, a cluster consists of a set of loosely connected PEs that work together so that in many respects they can be viewed as a single system, this PEs communicates in most cases through shared memory. Different clusters can be connected together through Local Area Network (LAN) [18]-[20]. In this work we implement the DR grading algorithm presented in [1] on a multi-cluster parallel system. In the next section the parallel based DR grading algorithm will be described in details.

IV. PARALLEL BASED DR GRADING ALGORITHM

This section presents the parallel implementation of the DR grading algorithm proposed in [1]. This algorithm proceeds in four stages; image preprocessing, statistical texture feature extraction, feature selection and classification stage. It was trained and tuned on 84 retinal images of which 62 images contain different signs of diabetic retinopathy, from the DIARETDB0 database [21]. The images in the dataset were classified by ophthalmologists based on the lesion type (exudates, Microaneurysms (red small dots) and Hemorrhages) exists [1]. The image categories were formed to confirm that each diabetic retinopathy finding type is included. The DIARETDB0 database was divided into four categories. Images having no lesions are considered normal, whereas images that have lesions like exudates, microaneurysms and

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

hemorrhages are considered abnormal. The database categorization is presented in table I.

TABLE I. DIABETIC RETINOPATHY CATEGORIES

Diabetic retinopathy group	Lesion type
Group 1	Red Small dots
Group 2	Red Small dots, hemorrhages, hard exudates, soft exudates.
Group 3	Red Small dots, hemorrhages, hard exudates.
Group 4	Normal

In parallel computing the problem must be divided into simple tasks and solves them concurrently. Parallel processing cannot be applied to all problems, in other words we can say that not all the problems can be coded in a parallel form. Refers to how the problems are divided up, there are two types of parallelization: (i) *fine-grained parallelization* which divides a problem into a large number of smaller tasks, usually of short duration, (ii) *Coarse-grained parallelization*, in which larger and longer independent tasks are computed in parallel. Finer granularity increases the amount of work that can be done simultaneously and so is potentially faster, but at the price of requiring more resources for communication between processors [5], [22] and [16]. To implement the parallel processing on DR grading algorithm presented in [1], different levels of parallelism are used. This combination will improve the system utilization and throughput. The purpose of this work is to quantify the design of such system analytically and bring parallelism into the sequential code of the algorithm proposed in [1].

TABLE II. THE CONSIDERED DATABASES

The database	The included groups	Number of images
db_1	Group1 and Group 2	17
db_2	Group1 and Group 3	22
db_3	Group1 and Group 4	19
db_4	Group2 and Group 3	22
db_5	Group2 and Group 4	25
db_6	Group3 and Group 4	27

A. *Problem description*

We assume that:

1. The parallel implementation operates on a computer cluster equipped with "*N*" homogenous nodes/clusters "C", where C={C_1,C_2,..., C_N}. Inside each node there are "*M*" homogenous PEs, where PE={PE_0,PE_1,..., PE_{M-1}} and PE_0 is the master one as shown in figure 1.
2. We have six datasets {db_1,db_2 , …db_6}, where each dataset consists of fundus images of two groups as presented in table II [1]. Each has "G_k" fundus images each represented by {R,G,B} images, where *k* is the database number, as shown at figure (2-a). The number of images belongs to each database are variable. In our experimental example, in the testing phase we assume that {G_1=17, G_2=22, G_3=19, G_4=22, G_5=25, and G_6=27}.
3. The preprocessing stage is accomplished as presented in [1]. Each image "g_k: g_k varies from 1 to G_k", belongs to database "db_k" is transformed to Hue, saturation and intensity (HSI) space. In this work we concerned with the "I" image only[1]. The "I" band image is filtered using the median filter, and then histogram equalization is applied. Afterward, weiner filter is utilized. Finally, the "I" band image is added to and subtracted from the weiner filtered image producing two images. Thus, for each sample image "img_{ki}" images are generated, where "*ki*" is {1, 2 or 3}.
4. Texture feature extraction is achieved as proposed in [1] during two steps. In the first step, four texture images are computed using the GLCM. These images are combined and thus other five images are determined. Next, statistical features of the obtained texture images are then extracted. Three window sizes are utilized to create the texture images (to be noted that utilizing one or two window sizes were found sufficient to discriminate between some groups [1]). Therefore, the window size selected to the database "db_k" is "W_k" and equal {1,2, or 3} corresponding to the window sizes (w = 25, 75, or 125) respectively as shown at figure (2-b).

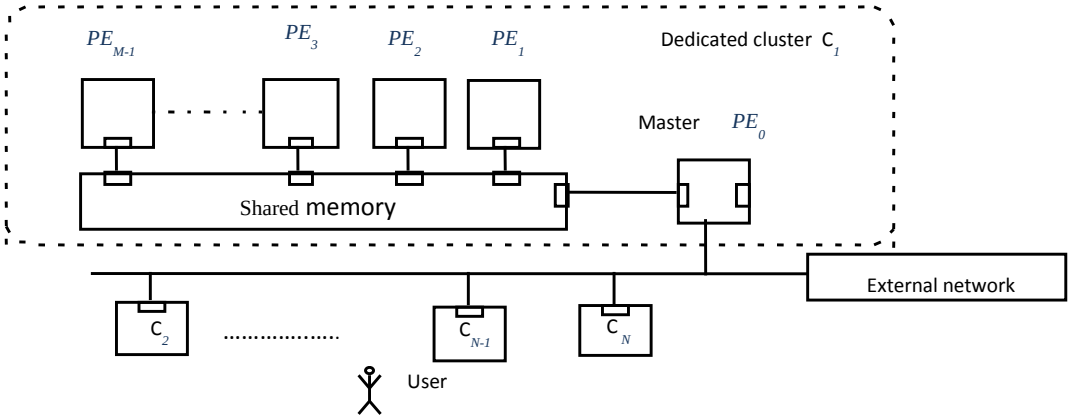
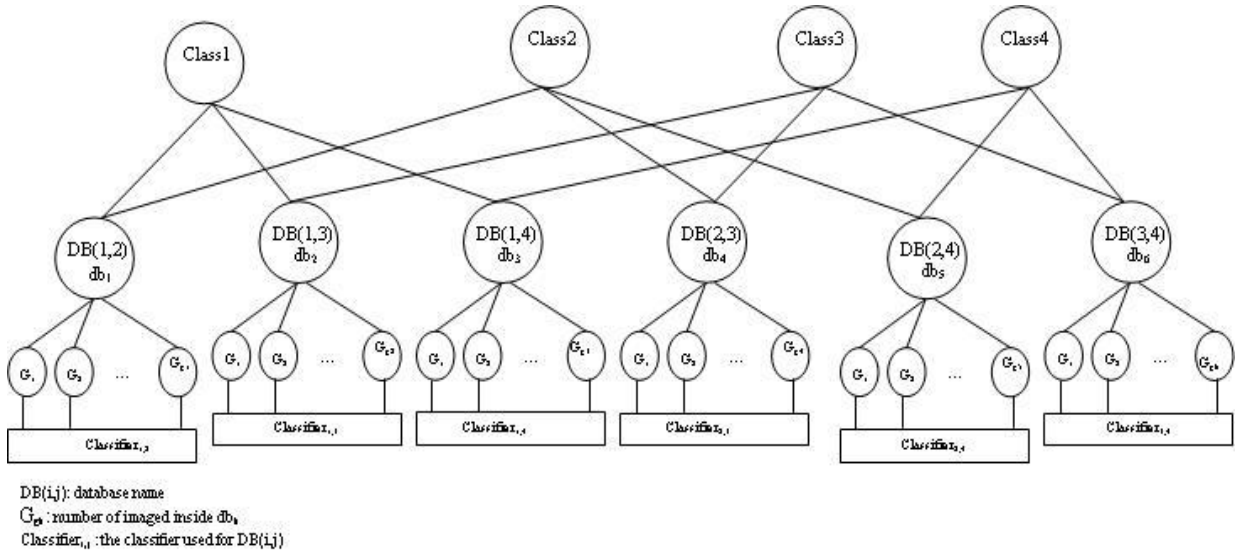
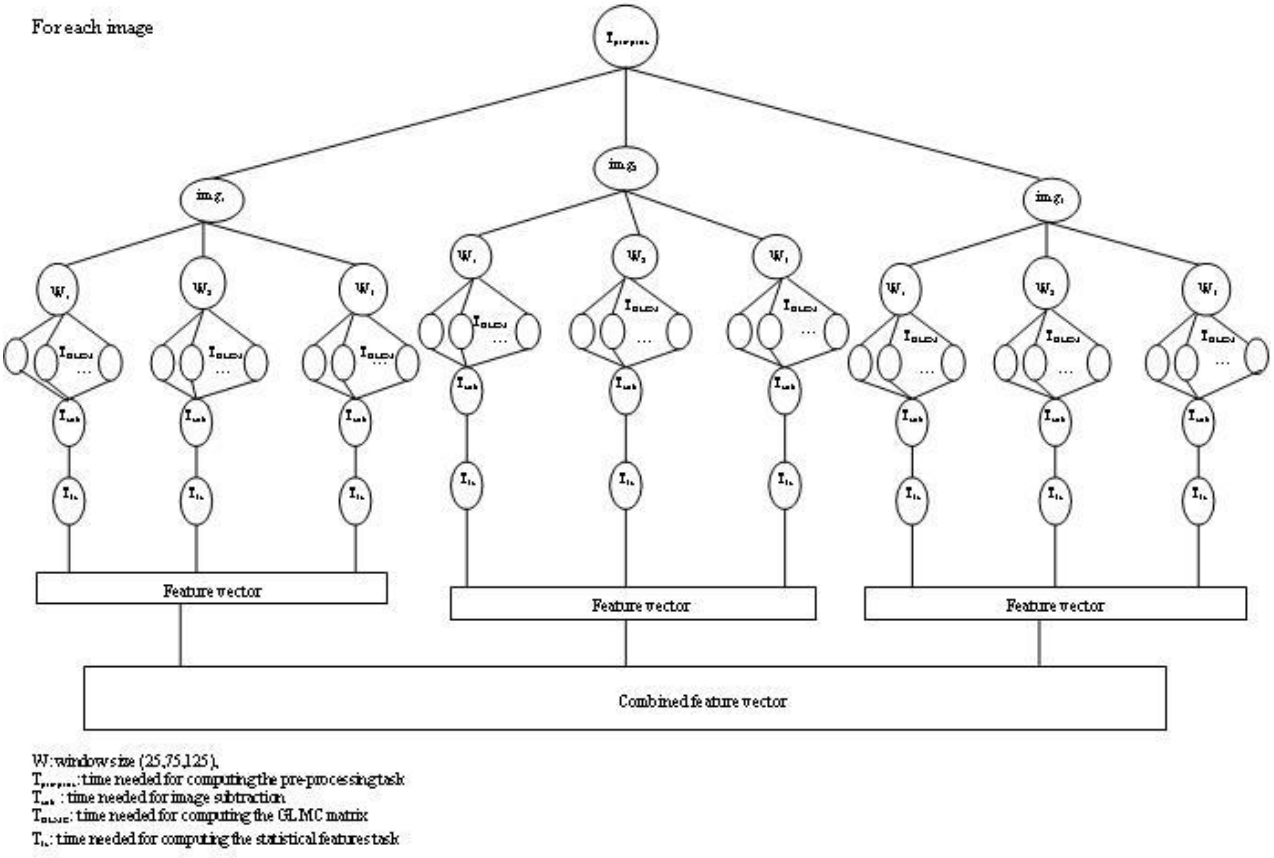


Figure 1. An *N*-Cluster system configuration

http://sites.google.com/site/ijcsis/
ISSN 1947-5500



(a) A DAG representation of the DR grading algorithm



(b): DAG representation of single image

Figure 2. [1] A Directed Acyclic Graph (DAG) representation

[1] Most parallel systems describe different applications by Directed Acyclic Graphs *DAG*s. A *DAG* is a directed graph that contains no cycles, the vertex weights represent task processing time and the edge weights represent data dependencies as well as the communication time between tasks. In addition, a collection of tasks that must be ordered into a sequence, subject to constraints that certain tasks must be performed earlier than others, may be represented as a *DAG* with a vertex for each task and an edge for each constraint [M. Cosnard, E. Jeannot, and T. Yang, "*Compact DAG Representation and its Symbolic Scheduling*", Journal of Parallel and Distributed Computing, vol. 64, pp. 921-935, 2004].

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

5. For each sample image "g_k" there are three tasks to be computed:
   - T_1: pre-processing task which generates at most three new images {img_1, img_2, img_3}
   - T_2: for each image "img_{ki}" there are more than one subtask to be computed
     - $T_{2\text{-}1}$: for each window size "W_k"
       - compute the nine texture images using GLCM
       - compute the statistical features for each texture image
     - $T_{2\text{-}2}$: generate a feature vector for each image "img_{ki}"
   - T_3: combine the feature vectors obtained from "img_{ki}" to generate a feature vector for "g_k"

The above steps must be repeated for all sample images belongs to each database.

Parallelization of DR grading algorithm presented in [1] will be discussed in the next sub-section. The complete step by step description of the proposed algorithm is given below.

B. The proposed algorithm

As mentioned at the previous section, parallelization can be done either in coarse-grained or fine-grained levels. In this work to implement the parallel processing on DR grading algorithm presented in [1], different levels of parallelism are used. In coarse grained level of parallelization different databases can be computed in parallel. To exploit parallelism, first these databases are assigned priorities and placed in a list ordered in decreasing values of priority. Whenever a cluster/node is available, highest priority database is selected from the list and assigned to it. In our work we assume that database which needs largest execution time takes the highest priority (Tested databases are sorted as follows {db1,db3,db2,db6,db4, and db5}). For "*N*" clusters there are two cases:

First: when the number of clusters *"N"* less than the number of databases, each node assigned a database from the database list sequentially, and the remaining (6-N) databases are assigned to the lightly loaded nodes ("*M*" PEs of each node share the computation of different images as discussed later at case 2).

Second: when the number of clusters *"N"* is equal to or larger than the number of databases, when (*N mod 6*) =0, the same number of nodes (*N/6*) is assigned to each database. On the other hand, if *(N mod 6)* ≠0, all databases are assigned the same number of nodes (*N/6*) and the remaining nodes are assigned to the first (*N mod 6*) databases of the list. Each dataset consists of "G_k" fundus images as presented in section 4.1. These images can be computed in parallel. To calculate the number of nodes assigned to an image "g_k: $1 \le g_k \le G_k$" of a database "db_k" (assuming that the number of nodes assigned to "db_k" is "NC_k" and all images need the same computation time) there are two cases:

- Case1: ($NC_k \ge G_k$) and (NC_k mod G_k) = 0, the number of nodes assigned to each image is $\left(NCg_k = \frac{NC_k}{G_k} \right)$ and ($M * NCg_k$) PEs cooperate to compute this image. In case of, (NC_k mod $G_k \neq 0$), each image is assigned $\left| \frac{NC_k}{G_k} \right|$ nodes, and the remaining nodes will helps the first (NC_k mod G_k) nodes (this case is not found at the test examples).
- Case 2: ($NC_k < G_k$) and (G_k mod NC_k) = 0, each node "C_i" ($1 < i \le NC_k$) computes $\left(G_i = \frac{G_k}{NC_k} \right)$ images. If (G_k mod NC_k) $\neq 0$, each node "C_i" computes $\left| \frac{G_k}{NC_k} \right|$ images and the remaining images will be computed by the first (G_k mod NC_k) nodes. Inside node "C_i", if the number of PEs, $M < G_i$ each PE computes $\frac{G_i}{M}$ images (in case of (G_i mod M) $\neq 0$ the remaining images will be computed by the first (G_i mod M) PEs). On the other hand, when $M \ge G_i$ and (M mod G_i) =0, each image "g_j: $1 \le j < G_i$" can be computed by $\left(\frac{M}{G_i} \right)$ PEs. For an image "g_j" which generates new "*kj*: 1 to 3" images (as mentioned above), new generated images can be computed in parallel as follows:

  (2.1) $\left(\frac{M}{G_i} \right)$ = 1, in this case image "g_j" is computed by a single PE, i.e. all generated images of image "g_j" will be sequentially computed by a single PE.

  (2.2) $\left(\frac{M}{G_i} \right)$=2 and *kj* =2, here each one the two PE computes one generated image "img_{kij}" in parallel. On the other hand, if $\left(\frac{M}{G_i} \right)$=2 and *kj* =3, each one of the two PE computes one generated image in parallel and then both PEs share the computation of the third one together (as discussed later in case (2.5)).

  (2.3) $\left(\frac{M}{G_i} \right)$= 3 and *kj* =3, in this case each generated image can be computed by one PE in parallel. On the

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

other hand, when $\left(\frac{M}{G_i}\right)$= 3 and *kj* =2, each one of the first PEs assigned one generated image, and the third one share one of them in its work (as discussed later in case (2.5)). Moreover, when $\left(\frac{M}{G_i}\right)$=3 and *kj* =1, the three PEs cooperate to compute this generated image (as discussed later in case (2.5)).

(2.4) $\left(\frac{M}{G_i}\right) \geq kj$ and $\left(\frac{M}{G_i} \bmod kj = 0\right)$, the number of PEs that cooperate to compute each generate image is $\left(Nimg_{kj} = \left(\frac{M}{G_i} * \frac{1}{kj}\right)\right)$. That is to say more than one PE computes a single generated image (as discussed later in case (2.5))

(2.5) $\left(\frac{M}{G_i}\right) \geq ki$ and $\left(\frac{M}{G_i} \bmod kj \neq 0\right)$, in this case the number of PEs that cooperate to compute each generate image is $\left(Nimg_{kj} = \left(\frac{M}{G_i} * \frac{1}{kj}\right)\right)$, and the remaining PEs can help the first $\left(\frac{M}{G_i} \bmod kj\right)$ PEs. As mentioned at section 4.1 each generated image "img_{kii}" is convolved with "W_k =1,2, or 3" windows in parallel (fine grained level of parallelization is applied). Assuming that the number of PEs that cooperate to compute image "img_{kij}" that was generated from image "g_i" of database db_k is "$Nimg_{kij}$", we can exploit parallelism as follows:

(2.5.1) $Nimg_{kij}$ =1, all windows must be computed sequentially by a single PE.

(2.5.2) $Nimg_{kij}$ = W_k =2, each one of the two PE computes one window in parallel. On the other hand, $Nimg_{kij}$ =2 and W_k =3, each one of the two PE computes one window in parallel and then both PEs share the computation of the third one (as discussed later in case (2.5.5)).

(2.5.3) $Nimg_{kij}$ = W_k =3, in this case each window "w_{kl}" can be computed by one PE in parallel. On the other hand, when $Nimg_{kij}$ = 3 and W_k =2, each one of the first PEs assigned one window, and the third one share one of them in its work (as discussed later in case (2.5.5)). Moreover, When $Nimg_{kij}$ = 3 and W_k =1, the three PEs cooperate together to compute this window (as discussed later in case (2.5.5)).

(2.5.4) $Nimg_{kij} \geq W_k$ and ($Nimg_{kij}$ mod W_k =0), each window "w_{kl}" can be computed by $\left(NCw_{kijl} = \frac{Nimge_{kij}}{W_k}\right)$ processing elements (that is to say more than one PE computes a single window as discussed later in case (2.5.5)).

(2.5.5) $Nimg_{kij} \geq W_k$ and ($Nimg_{kij}$ mod W_k ≠0), each window "w_{kl}" can be computed by $\left(NCw_{kijl} = \frac{Nimge_{kij}}{W_k}\right)$ processing elements, and the remaining PEs can help the first ($Nimg_{kij}$ mod W_k) PEs. Here more than one PE compute a single window. In this case parallelization is applied where texture images computation using GLCM can be done partially in parallel. In this work parallel computations are achieved by image partitioning on different processing elements (partition iterations loop on different PEs [23]). In image partitioning, the image is divided into different groups consisting of rows or columns in which each group is assigned to one single PE. Each group contains an equal number of rows or columns. Assume that, for image "img_{ki}" at "w_{kl}", image partitioning can be done on "NP_{kijl}" PEs as follows:

- NP_{kijl} < [(R-w-1)]: each PE computes {mod[(R-w-1) / NP_{kijl}]} loops and {mod[(R-w-1)/[(R-w-1) – (NP_{kijl} -1)]]} PEs will compute one additional loop.
- NP_{kijl} = (R-w-1): each PE computes only one loop.
- NP_{kijl} > (L-w-1): each PE computes only one loop and (NP_{kijl} - (R-w-1)) PEs will be idle.

C. LOAD BALANCING

As mentioned in table II, different databases have different number of images. Furthermore, each image may generate different number of new images according to the predefined dataset. This leads to one or more nodes have very few tasks to handle while other nodes have many; in this case; a problem of *load imbalance* occurs. This imbalance reduces the overall performance of the parallel systems. To improve system performance, the load must be redistributed among all

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

nodes; this process is called *load balancing*. A good load balancing algorithm should minimize the total execution time, while limiting the communication overhead for data transfer. Load balancing algorithms (also called *load re-distribution*) can be classified as either *static* or *dynamic*. Static algorithms allocate the workload to different nodes during compile time (that is to say, static load balancing routine is executed once). In contrast, dynamic algorithms distribute the load during runtime. Static load balancing is usually referred to as the *re-scheduling* problem. That is to say, static algorithms are based on a prior knowledge of the problem structure (all the information that governs load distribution and redistribution decisions is known before run-time). In contrast, dynamic algorithms distribute the load among processors during runtime based on the behavior of the application [24]. In our work static load balancing (SLB) is the suitable way to balance the load. That is because the complete knowledge about the problem structure and the number of images utilized is known at the compilation stage. There are four types of static load balancing: Round Robin algorithm, Randomized algorithm, Central Manager algorithm, and Threshold algorithm [25]. To improve the system performance, we apply the threshold static load balancing algorithm [26] on the proposed parallel based DR grading algorithm. Balancing is started by classifying each node as either being under-loaded, normal-loaded, or overloaded. A threshold value is used to partition the states of nodes into these categories. Two threshold parameters t_under and t_upper can be used to describe these levels. (i) Under-loaded: load < t_under, (ii) Normally-loaded: t_under ≤ load ≤ t_upper, and (iii) Overloaded: load > t_upper. To achieve fairness, some researchers choose the average execution time executed by all nodes as the key to classify the status of different nodes [27]. In our work we assume that: (i) t_upper = t_avg.+ ε and (ii) t_under = t_avg.- ε, where "ε" is a small constant value = 15% of the average value.

Load can be exchanged between the overloaded and under-loaded nodes. Load redistribution can be done between nodes/clusters and between PEs inside each cluster. In case of "$N<6$", only PEs inside each cluster exchange the load to minimize the synchronization and communication times. On the other hand when "$N\geq6$", tasks can be exchanged between different clusters in order to improve the system performance. Different nodes/PEs can (i) Exchange complete images (between clusters or between PEs), or (ii) Exchange sub-images (between PEs inside each cluster). After load rescheduling tasks can be computed as shown in section 4.2 (case 2). Next section describes the discussion of the results obtained with the parallel implementation of the DR grading algorithm presented in [1].

V. IMPLEMENTATION AND DISCUSSION OF RESULTS

This section presents the implementation of the DR grading algorithm proposed in [1] on different clusters architecture with and without using load balancing. In our experiments we applied 44 fundus images from the DIARETDB0 database [21]. These images are classified into four groups as illustrated in table I [1]. Those four classes are utilized to generate six databases each database consists of images from two groups as presented in table II. It is clear from table II that the number of images belong to the different databases are varied. In this paper, the system performance utilizing computer cluster architecture is compared to its performance when using Intel Processor (Core i7). To evaluate the performance of a parallel system, we must first choose some criteria. These criteria are called *metrics*. Characteristics or properties of a good performance metric should be specific, measurable, acceptable, and realizable. Many performance metrics have been proposed to quantify the parallel systems. Among of them are execution (parallel) time, speedup, efficiency, processor utilization, communication overheads, and etc. *Execution time (parallel time)* T_{par} which is referred to the total running time of the program, is the most obvious way of describing the performance of parallel programs. T_{par} is the time interval between the beginning of parallel computation and the time since the last processing elements (*PE*s) finishes execution. It is the sum of the computation time, and the communication time (overhead time). The computation time is the sum of essential and excessive computation. While, the communication time is the total time needed to send and receive data between nodes/PEs. Another parallel metric is *Speedup "Sp"*. *Speedup* is defined as the ratio of the time taken to execute a problem on a single processing element (*called serial time* T_s) to the time required to solve the same problem on a parallel system T_{par}, and $S_P = \frac{T_s}{T_{par}}$. Moreover, *Efficiency* "*Ep*" is another measure for performance evaluation. This measure is very close to *Speedup*, it is the ratio of speedup to the total number of processing elements "*M*", and $E_P = \frac{S_p}{M}$ [27]. As communication overhead becomes an increasing factor and can exceed the total parallel time T_{par}, the computation to communication ratio must be considered as a performance measurement. For our application, there is no communication time needed for the coarse-grained level of parallelism, that is to say, there is no communication among different databases' images (independent tasks). While; for the fine-grained level of parallelization; the communication between PEs inside each node which cooperate to compute the process of computing the texture images using the GLCM is estimated to be about "one mille-second" assuming fully connection inter-processor communication (this time can be neglected). In addition, the communication time needed to send the number of the generated features from different PEs to the master one (inside each node) is also negligible. As explained in section 4, DR grading algorithm proposed in [1] proceeds in four stages; image preprocessing, statistical texture feature extraction, feature selection and classification stage. The feature selection stage will not be taken into consideration since we are going

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

to use the selected features proposed in [1]. The process of computing the texture images using the GLCM can be computed in parallel; while the obtained statistical features form each texture image must be computed serially. That is because the synchronization and communication times between different nodes are dominant compared to the computational time. The entire above stages must be repeated for each image inside each database (course-grained parallelism). Without using parallel processing the time needed to run six independent tasks (six different databases) is as follows: the average time needed for a single run against a given database with average number of images = 22, and two different window sizes, is about 16.5 hours on a Laptop with Intel Processor (Core i7), 2.20 GHz clock, and 8 GB RAM. This implies that running six independent experiments requires about 98 hours. Figure (3) presents the performance of the proposed parallel based DR grading algorithm on different number of clusters/nodes (from two to twenty). The number of PEs inside each node is varied from two to thirty-two to observe their effect on the DR grading based algorithm performance. Figures (4) and (5) present a comparison between balancing and non-balancing implementations when using different number of nodes with different number of PEs. From these figures the following observations can be noted:

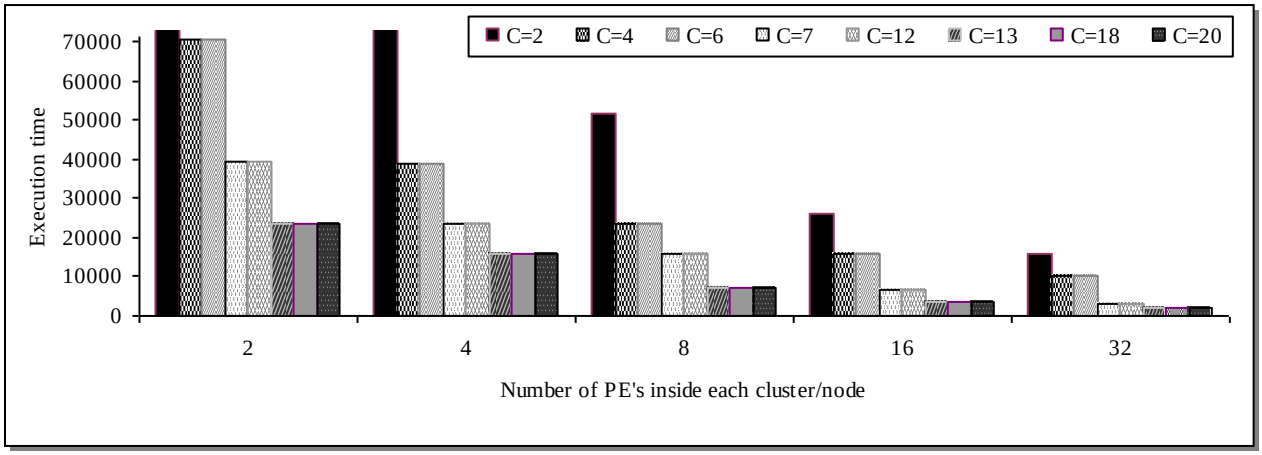
- The results shown in figure (3) depict that the total execution time significantly reduces as the number of nodes (all nodes have the same number of PEs) increases, see figure (3-a). In addition, upon increasing the number of nodes, the speedup increases, and consequently the efficiency decreases as illustrated in figures (3-b) and (3-c). The reason for the efficiency reduction is load imbalance.
- As shown in figure (3-a), without load balancing when increasing the number of nodes from 4 to 20, for the iterations (N=4,…to 6), the execution time is constant and decreases at "N=7", this value remains constant until "N=12". That is because the number of nodes that cooperate to compute the largest database (which has the largest number of images) is constant which leads to constant execution time. Generally, when increasing the number of nodes that assigned to the largest database, the execution time decreases at (N mod 6=1) and remains constant until reaches (N mod 6 =0). On the other hand, while the speedup behaves like the execution time, that is to say Sp increases at (N mod 6=1) and remains constant until reaches (N mod 6 =0), the system efficiency decreases when increasing the number of nodes as shown in figures (3-b), (3-c). Figure (3-c) shows that at each (N mod 6 =1) the efficiency increases because the number of nodes assigned to the largest database increases. This behavior can be improved by applying load balancing as shown in figures (4) and figure (5). The results presented in these figures validate this claim, these figures show that with increasing the number of nodes the execution time decreases, while the system efficiency increases that is because the computational time distributed between different nodes.
- Figures (5-a) and (5-b) depict the execution time and efficiency of different cluster architecture (from two to thirteen) for computation of the DR grading algorithm. Figure (5-a) shows that load balancing reduces the execution time required compared to its corresponding value without balancing. In case of C=2, there is no need for SLB between the two clusters because for both clusters t_under ≤ load ≤ t_upper. In this case balancing can be done only between different PEs inside each cluster. Figure (5-b) shows that applying SLB increases the efficiency compared to its corresponding value without balancing.
- Without applying SLB the increase of the total number of processors "$N*M$" will decrease the system efficiency as shown in figure (5-b). Applying load redistribution will improve the system performance. To obtain a reasonable efficiency (equals to 75% after balance), we will be satisfied with an improvement degree ($(T_S-T_{par})/T_s$) equals 98% which can be reached at "N=13 and M=16". That is to say the time needed by the algorithm in case of parallel processing is about 56 minutes only (after balance) when using 13 nodes each has 16 PEs which leads to a reduction of about 98% of the processing time without using parallel processing with efficiency equals to 75%.

VI. CONCLUSIONS

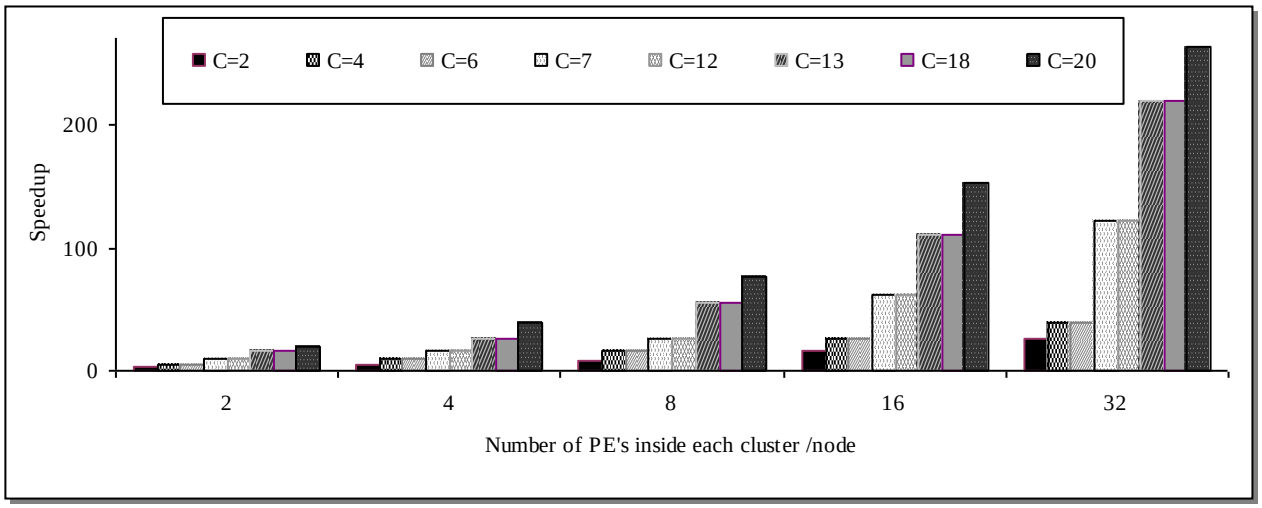
This paper presents a novel algorithm for fast parallel computation of diabetic retinopathy (DR) grading algorithm presented in [1]. The aim is to improve its performance by utilizing parallel concepts which distribute the employed datasets into "N" different sets for "N" different nodes which reduces the computational complexity, processing power and memory requirements. Each sample image is preprocessed then several images are generated. These images are split into a number of parts and each part is sent to a separate computing node to create texture images using GLCM. Afterward, statistical features are extracted from these texture images. This improves the system utilization and throughput. The number of nodes is not related to the size of the problem, which reduces the design area to a minimum compared to other schemes. From the practical work we can conclude that: sources of parallelism exist in all phases of the algorithm proposed in [1] (DBs, texture images and statistical feature). Our experiments show that the process of generating texture images using GLCM is the most promising phase to exploit parallelism because it contains huge amount of computations (multiplications and operations) compared with those in the other phases. The choice of parallel architecture is another factor that affects the performance of the proposed scheduling algorithm. In our work we choose the cluster computing architecture because it is available and not expensive. In addition, its communication times are not significant, and it is hardware independent. Furthermore, when the number of nodes increases the problem of load imbalance appears, and then load balancing algorithm should be applied. In this work, threshold static load balancing is applied on the proposed parallel DR grading based algorithm. It is clear from this analysis that parallel implementation of the DR grading algorithm proposed in [1] reduces its computational times by a factor of "98%" over non-parallel implementations.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

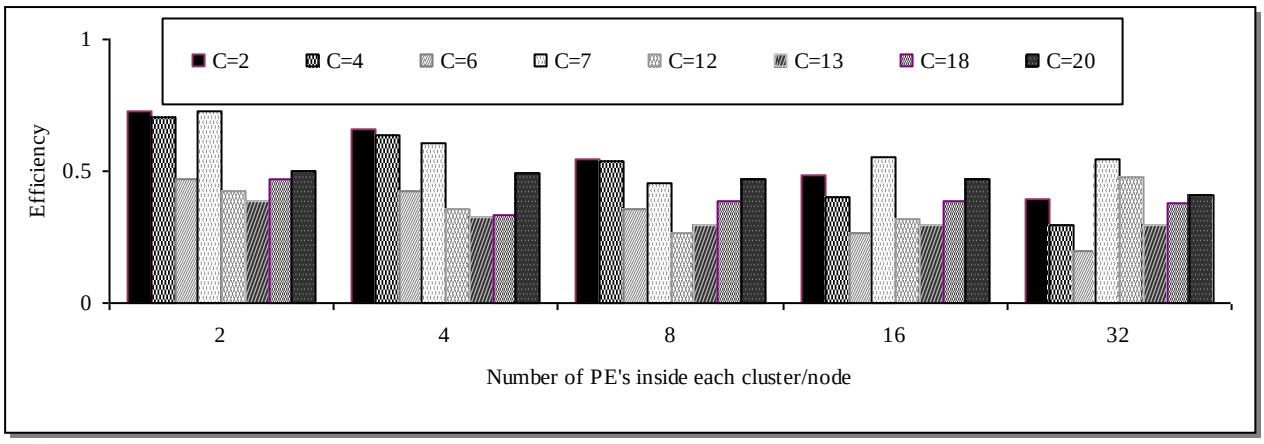
Researchers usually use a small portion of the data set to validate their algorithm, due to the computational complexity of the algorithm. Once a computer cluster is available, we will be able to test the proposed algorithm in a much larger data set. Hence, the overall system performance can be improved.



a) Execution time



b) Speedup



c) Efficiency

Figure 3. Parallel Based DR Grading Algorithm Performance Without Using Load Balancing

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

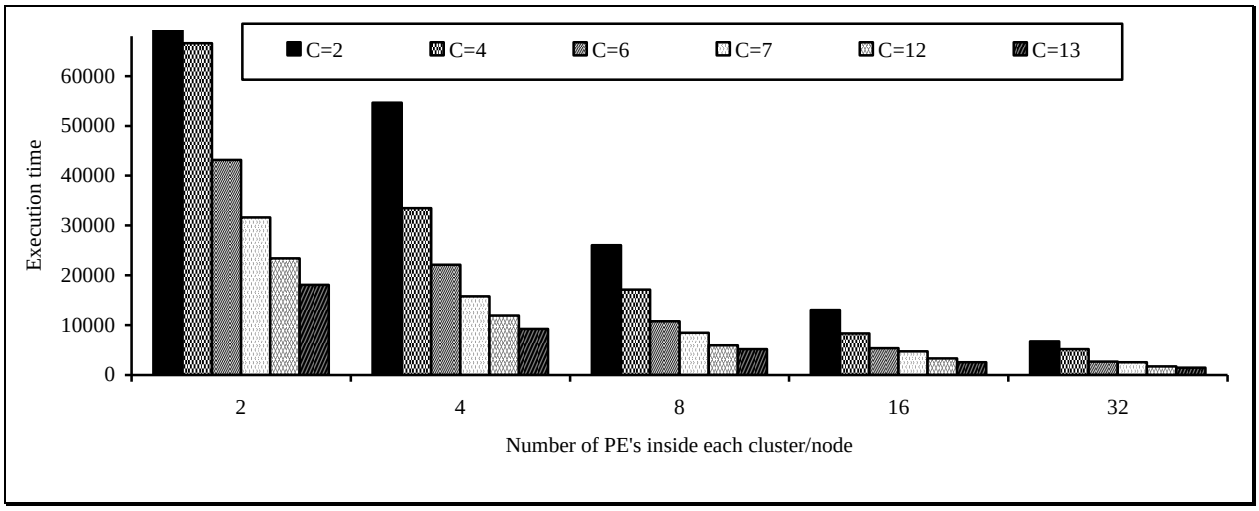
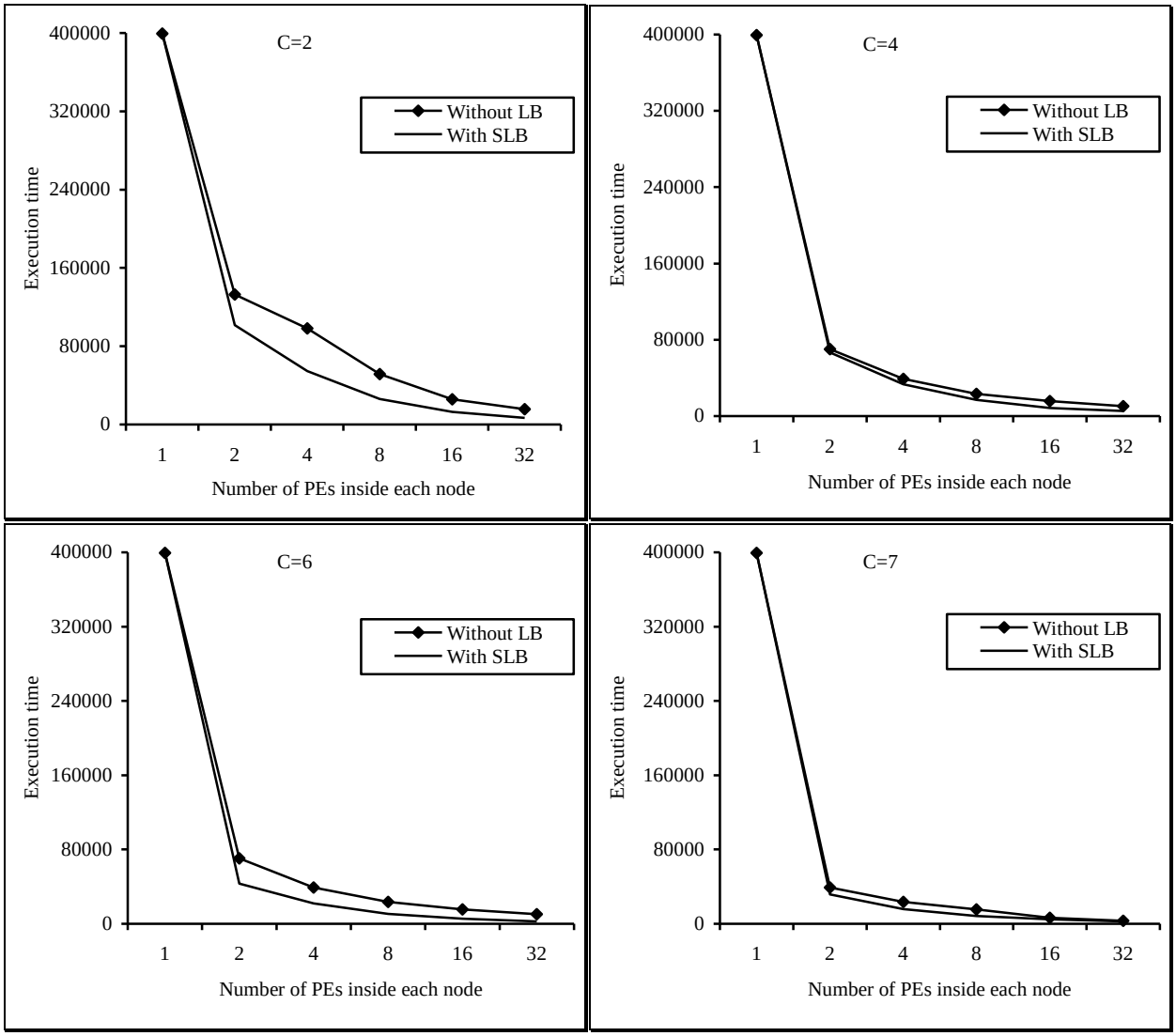
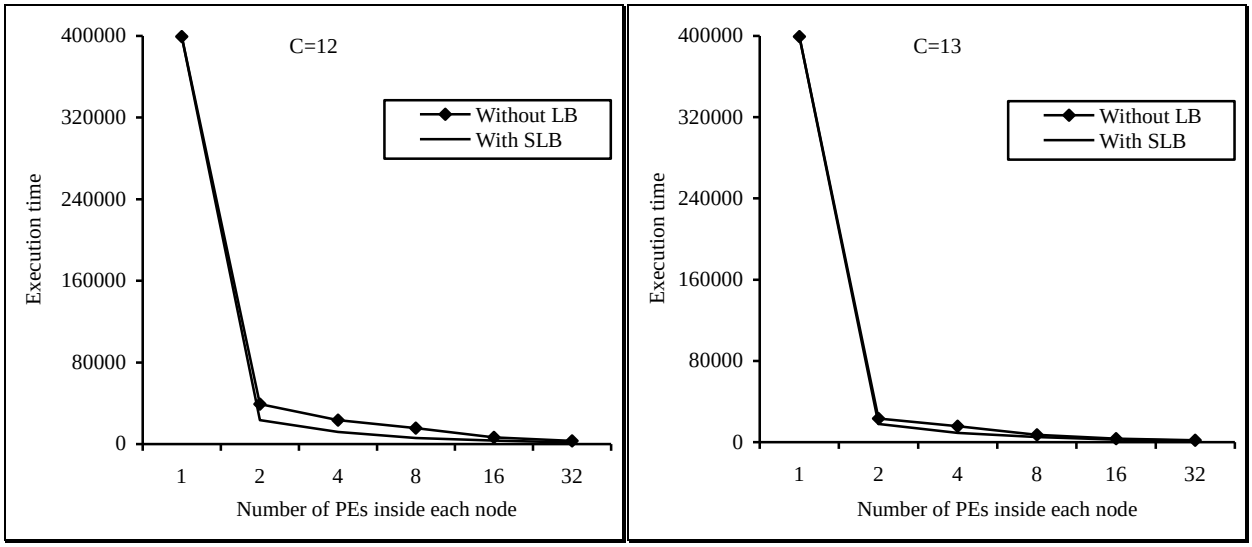


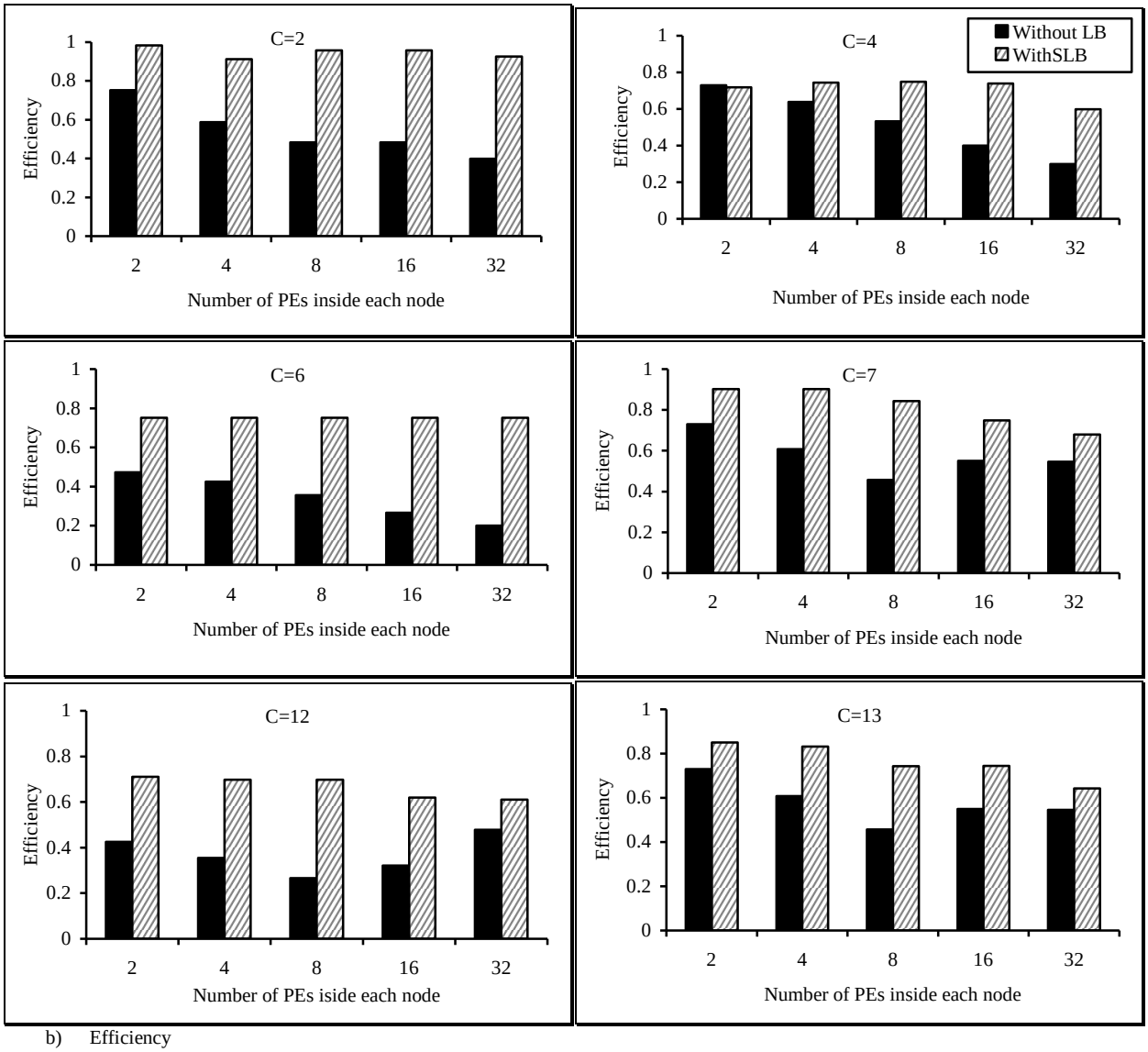
Figure 4. Parallel Based DR Grading Algorithm Execution time when Using Static Load Balancing (SLB)



http://sites.google.com/site/ijcsis/
ISSN 1947-5500



a) Execution time



b) Efficiency

Figure 5. The performance improvement when using SLB

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

REFERENCES

[1]. H. A. Elnemr and A. A. Hefnawy, "A Significance test-based feature selection method for diabetic retinopathy grading", CiiT Int. J. Digital Image Processing, Accepted May, 2013.

[2]. P. Mitchell, S. Foran, T. Wong, B. Chua, I. Patel and E. Ojaimi, "Guidelines for the management of diabetic retinopathy", National Health and Medical Research Council, Commonwealth of Australia, no. 4176, 2008.

[3]. R.J. Windera, P.J. Morrowb, I.N. McRitchiea, J.R. Bailie and P.M. Hart," Algorithms for digital image processing in diabetic retinopathy," Computerized Medical Imaging and Graphics, Elsevier Ltd, vol.33, issue 8, pp. 608-622, Dec. 2009.

[4]. K. Narasimhan, Neha. V. C., and K. Vjayarekha, "A Review of automated diabetic retinopathy diagnosis from fundus image," J. Theoretical and Applied Information Technology, vol. 39, no.2, 2012.

[5]. S. Saxena, N. Sharma, and S. Sharma, "Image processing tasks using parallel computing in multi-core architecture and its applications in medical imaging", Int. J. Advanced Research in Computer and Communication Engineering, vol. 2, issue 4, April 2013.

[6]. Y. Kadah, Kh. Abd-Elmoniem, and A. Farag, "Parallel computation in medical imaging applications", Editorial, Hindawi Publishing Corporation, International J. Biomedical Imaging, vol. 2011, 2 pages.

[7]. N. Shrivastava, P. Saurabh and B. Verma,"An efficient approach parallel support vector machine for classification of diabetes dataset", In. J. Computer Applications, vol. 36, no.6, pp.19-24, Dec. 2011.

[8]. DK. Iakovidis, DE. Maroulis, and DG. Bariamisa, "FPGA architecture for fast parallel computation of co-occurrence matrices", Microprocessor and Microsystems J., vol. 31, no. 2, pp 160–165, March 2007.

[9]. A. Reza Akoushideh and A. Shahbahrami, "Accelerating texture features extraction algorithms using FPGA architecture", Int. Conf. on Reconfigurable Computing and FPGAs, Cancun, Quintana Roo, Mexico, pp. 232-237, 13-15 Dec., 2010.

[10]. A. Shahbahrami, T. Pham, and K. Bertels, "Parallel implementation of gray level co-occurrence matrices and Haralick texture features on cell architecture", J. Supercomputers, no. 59, pp 1455–1477, 2012.

[11]. L. Sieler, C. Tanougast C, A. Bouridane, "A scalable and embedded FPGA architecture for efficient computation of grey level co-occurrence matrices and Haralick textures features", Microprocessor and Microsystems J., vol. 34, no. 1, pp 14–24, Feb. 2010.

[12]. MA. Tahir, A. Bouridane, and F. Kurugollu, "An FPGA based coprocessor for GLCM and Haralick texture features and their application in prostate cancer classification", J. Analog Integrated Circuits Signal Process, vol. 43, no.2, pp 205–215, 2005.

[13]. D. Bariamis, DK. Iakovidis, and DE. Maroulis, "Dedicated hardware for real-time computation of second-order statistical features for high resolution images", Lecture Notes in Computer Science, Springer, Berlin, vol. 4179, pp 67–77, 2006.

[14]. H. Sugano, and R. Miyamoto, "Parallel implementation of good feature extraction for tracking on the cell Processor with OpenCV interface (IIH-MSP '09)", 5th IEEE Int. Conf. on Intelligent Information Hiding and Multimedia Signal Processing, koyoto, Japan, pp 1326–1329, 12-14 Sept., 2009.

[15]. M. Gipp, G. Marcus, N. Harder, A. Suratanee, K. Rohr, R. Konig, and R. Manner, "Haralick's texture features computed by GPUs for biological applications", IAENG Int. J. Computer Science (IJCS), vol. 36, no. 1, Feb. 2009.

[16]. E. Olmedo, Jorge de la Calleja, Antonio Benitez, and Ma Auxilio Medina," Point to point processing of digital images using parallel computing", Int. J. Computer Science Issues (IJCSI), vol. 9, issue 3, no. 3, May 2012.

[17]. Y. Tanaka, M. Mastsuda, M. Ando, K. Kubota, and M. Sato, 'COMPas: A Pentium pro PC-based SMP cluster and its experience", IPPS Workshop on Personal Computer Based Networks of Workstations, LNCS 1388, pp. 486–497. 1998.

[18]. M. Tudruj, "Embedded cluster computing through dynamic reconfigurability of inter-processor connections", IWCC 2001, LNCS 2326, Springer-Verlag Berlin Heidelberg, pp. 77–91, 2002.

[19]. A. Gupta, "Cluster schedulers", B.Tech, 6th Semester, 04010102, Department of Computer Science and Engineering Indian Institute of Technology Guwahati; 5-9.

[20]. M. M. Al-Jarrah, S. A. Amin and R. S. Naom, "Parallel implementation of systolic array design for developing medical image rotation", Computer Engineering and Intelligent Systems vol.4, no.2, pp 49-56, 2013.

[21]. T. Kauppi, V. Kalesnykiene, J. K. Kamarainen, L. Lensu, I. Sorri, H. Uusitalo, H. Kalviainen, and J. Pietila., "Diaretdb0: evaluation database and methodology for diabetic retinopathy algorithms", Technical Report, Lappeenranta Univ. of Technology, Lappeenranta, Finland, 2006.

[22]. P. Pacheco, An introduction to parallel programming, Morgan Kaufmann, 2011.

[23]. Y. Yan, C. Jin, and X. Zhang, "Adaptively scheduling parallel loops in distributed shared-memory systems", IEEE Trans. on parallel and distributed systems, vol. 8, no. 1, pp 70-81, Jan. 1997.

[24]. G. Elkabbany, N. Wanas, S. Shaheen, and N. Hegazi, "A dynamic load balancing framework for real-time applications in message passing systems", Inter. J. of Parallel Programming, vol. 39, issue 2, pp 143-184, 2011.

[25]. S. Abu Elanin, and M. Kitakami, "Performance analysis of static load balancing in grid", Int. J. Electrical & Computer Sciences IJECS-IJENS, vol. 11, no. 03, pp. 57-63, 2011.

[26]. S. Sharma, S. Singh, and M. Sharma, "Performance analysis of load balancing algorithms", Academy of science, engineering and technology, issue 38, pp. 269-272, Feb. 2008.

[27]. S. Moharil, S. Lee, "Load balancing of parallel simulated annealing on a temporally heterogeneous cluster of workstation", 21st Int. Symposium on Parallel Distributed Processing (IPDPS'07), pp. 1–8. Long Beach, CA, 26–30 March, 2007.

[28]. A. Chhabra, and G. Singh, "Analysis and integrated modeling of the performance evaluation techniques for evaluating parallel systems", Int. J. Computer Science and Security (IJCSS), vol. 1, issue 1, pp. 1-10, May/June 2007.

AUTHORS PROFILE

Ghada Farouk ElKabbany is an Assistant Professor at Electronics Research Institute, Cairo-Egypt. She received her B. Sc. degree, M. Sc. degree and Ph. D. degree in Electronics and Communications Engineering from Faculty of Engineering, Cairo University, Egypt in 1990, 1994 and 2007 respectively. Her research interests include: High Performance Computing (HPC), robotics, image processing, and computer network security.

Heba Ahmed Elnemr is an Assistant Professor at Electronics Research Institute, Cairo-Egypt. She received her B. Sc. degree, M. Sc. degree and Ph.D. degree in Electronics and Communications Engineering from Faculty of Engineering, Cairo University, Egypt in 1990, 1995 and 2003 respectively. She has supervised several masters and Ph. D. students in the field of image processing. Her research interests include: pattern recognition, signal processing, biometrics and image processing.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

Maximum Battery Capacity Routing to Prolong Network Operation Lifetime in Wireless MESH Network alongside the OLSR Protocol

Ramezanali Sadeghzadeh
Faculty of Electrical and Computer Engineering, K.N. Toosi University of Technology
Tehran, Iran
Sadeghz@eetd.knut.ac.ir

Asaneh Saee Arezoomand
Dept. of Electrical and Computer Engineering, Science and Research Branch, Islamic Azad University
Tehran, Iran
Afsanesaee@yahoo.com

Mohammad Zare
Dept. of Electrical and Computer Engineering, Science and Research Branch, Islamic Azad University
Tehran, Iran
AriooBarzan@Hotmail.Com

***Abstract*— A wireless mesh network (WMN) is a communications network made up of radio nodes organized in a mesh topology. Wireless mesh networks often consist of mesh clients, mesh routers and gateways. The mesh clients operate on batteries such as cell phone, laptop and .., while the mesh routers forward traffic to and from the gateways which may, but need not, connect to the Internet. To maximize the lifetime of mesh mobile networks, the power consumption rate of each node must be evenly distributed, it is essential to prolong each individual node (mobile) lifetime since the lack of mobile nodes can result in partitioning of the network, causing interruptions in communications between mobile nodes, and finally the overall transmission power for each connection request must be minimized. In this article we propose a new metric to find a proper route in wireless Mesh network and beside it we study OLSR protocol that it can be used in Ad hoc network.**

Keywords- Wireless Mesh; Ad hoc Network; Energy Consumption; Power Control; OLSR protocol

I. INTRODUCTION

Wireless mesh architecture is a first step towards providing cost effective and dynamic high-bandwidth networks over a specific coverage area [1], [2].

Mesh networks may involve either fixed or mobile devices. The solutions are as diverse as communication needs, for example in difficult environments such as emergency situations, tunnels, oil rigs, battlefield surveillance, high speed mobile video applications on board public transport or real time racing car telemetry. An important possible application for wireless mesh networks is VoIP. By using a Quality of Service scheme, the wireless mesh may support local telephone calls to be routed through the mesh.

The term 'wireless mesh networks' describes wireless networks in which each node can communicate directly with one or more peer nodes. It is a multi-hop wireless network [8] and consists of two types of nodes: mesh routers and mesh clients. Mesh routers have minimal mobility and form the backbone of WMNs, some of them are called gateway nodes and connected with a wired network. Client mesh networks comprise of energy-limited, mobile devices such as laptops and IP Phones. The mesh clients have mobility requirements as well as energy constraints, thus making communication challenging.

Dependence of power-consumption constraints on the type of mesh nodes. Mesh routers work by the endless power energy from backbone Internet and mesh clients by the limited battery, so we focus on the routing of the mesh clients. The lack of mobile clients can result in partitioning of the network, causing interruptions in communications between mobile clients. Since most mobile clients today are powered by batteries, efficient utilization of battery power is more important than in cellular networks. It also has an important influence on the overall communication performance of the network.

II. POWER-EFFICIENT ROUTING PROTOCOLS

In this section, we present a brief description of the relevant energy-aware routing algorithms proposed recently.

A. Minimum Battery Cost Routing (MBCR)

This metric reduce the total power consumption of the overall network. But, it has a critical disadvantage, it does not reflect directly on the lifetime of each client. If the minimum total transmission power routes obtain via a specific client, the battery of this client will be exhausted quickly, and this client will die of battery exhaustion soon. So, the remaining battery capacity of each client is a more accurate metric to describe the lifetime of each client [10].
Let $f_i(c_i^t)$ be the battery cost function of a client n_i. Now, suppose a node's willingness to forward packets is a function of its remaining battery capacity. As proposed, one possible choice for f_i is

$$f_i(c_i^t) = \frac{1}{c_i^t} \quad (1)$$

Where c_i^t is the battery capacity of a client n at time t.

The battery cost R_j for route j, consisting of D nodes, is

$$R_j = \sum_{i=0}^{D_j-1} f_i(c_i^t). \quad (2)$$

Therefore, to find a route with the maximum remaining battery capacity, we should select a route i that has the minimum battery cost.

$$R_i = min\{R_j | j \in A\} \quad (3)$$

where A is the set containing all possible routes.

Since only the summation values of battery cost functions is considered, a route containing nodes with little remaining battery capacity may still be selected, which is undesirable.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

B. Minimum Total Transmission Power Routing

In the MTPR mechanism, the total transmission energy for the route is calculated as: $P(r_d) = \sum_{i=0}^{d-1} T(n_i + n_{i+1})$. Where a function $T(n_i, n_j)$ denoting the energy consumed in transmitting over the hop (n_i, n_j) and generic route is $r_d = n_0, n_1, \ldots n_d$, where n_0 is the source node and n_d is the destination node.

The optimal route r_o satisfies the following condition:

$$P(r_o) = \min_{r_j \in r_*} P(r_j) \quad (4)$$

Where r_* is the set of all possible routes.

C. Min-Max Battery Cost Routing (MMBCR)

Equation (2) can be modified to make sure that no node will be overused, as indicated in [4]. Battery cost R_j for route j is redefined as

$$R_j = \max_{i \in route\_j} f_i(c_i^t) \quad (5)$$

Similarly, the desired route i can be obtained from the equation

$$R_i = \min\{R_j | j \in A\}$$

As MMBCR always tries to avoid the route with nodes having the least battery capacity among all nodes in all possible routes, the battery of each host will be used more fairly than in previous metrics. But, there is no guarantee that minimum total transmission power paths will be selected under all circumstances.

III. OUR PROPOSED NEW ROUTING MODEL

Nodes in Wireless Mesh Network (WMN), especially in client topology, are battery driven. Therefore, they suffer from limited energy level problems. In such an environment there are two important reasons that result in partitioning of the network: 1) Node dying of energy exhaustion, and 2) Node moving out of the radio range of its neighboring node. Hence, to achieve the best route in WMNs, node stability is essential. According to previous discussions, our goal is to maximize the lifetime of each node and use the battery fairly. But, in this new model we consider both hop-counts and remaining battery capacity together to achieve a proper route. In other words, battery capacity defines as a cost function with spot a number of hops. WangBo proposed [5] new energy consumption model that considers the hops too. So, energy consumption of each node will be use fairly and energy distribution implement better over more hops. To represent new model are four different conditions:

- Equal number of hops with different total cost.
- Different number of hops with equal total cost.
- Different number of hops with different total cost.
- Equal number of hops with equal total cost.

$f_i(c_i^t)$ is the battery cost function (BCF) of a client n_i that defines as Equation (1) in MBCR, i.e. $f_i(c_i^t) = \frac{1}{c_i^t}$. Where c_i^t is the battery capacity of a client n_i at time t. But, in new model the battery cost R_j is different from equation (2). Since we consider the number of hops the battery cost R_j, consisting of M nodes, is redefined as follows:

$$R_j = \frac{1}{H} \sum_{i=0}^{M_j - 1} f_i(c_i^t) \quad (6)$$

where H denote the number of hops. So, to find a route with the maximum remaining battery capacity, the favorite route i can be obtained from the equation

$$R_i = \min\{R_j | j \in A\} \quad (7).$$

This model prevents the nodes in the routing with consideration their cost which have a less battery capacity and balance the energy consumption of each node.

IV. PERUSE THE NEW MODEL WITH AN EXAMPLE

In order to illustrate the new model, we give an example (Fig. 1) to express it.

In figure 1 if source node is *S* and destination node is *D*, there are three routes between them. First, battery cost R_j is calculated for three routes. Then, according to equation (7) the best route is elected. To compare these three routes we consider three different conditions as mentioned in previous section. Since we consider critical states in the example, we compare two steps of conditions. 1) Route 1 and route 2 show the state 2, i.e. Different number of hops with equal total cost. If the prior metric implemented to select the route, the route with minimum hops was selected i.e. route 1 while the node 3 in this route has a little energy capacity, which is undesirable. But, whit this new metric route 2 is elected that have enough energy capacity and distributed energy has done well. Although route 2 has an end-to-end delay higher than route 1, route 2 restrains interruptions in communications between mobile clients. 2) Route 2 and route 3 show the state 3, i.e. Different number of hops with different total cost. Again if the prior metric implemented like MBCR, the route 3 was selected that has a minimum total energy, it can consume more power to transmit user traffic from a source to a destination, which actually reduces the lifetime of all nodes. But with new metric after calculate the R_j, route 2 is selected. In addition, more hops can reduce the total transmission power consumption and balance the energy consumption of each node.

Most of the previous metrics have surveyed on equal hops. But, this new metric has studied on different hops to find a proper route which has shown in figure 1.

V. THE STRUCTURE OF OUR SIMULATOR

Different routing protocols have been proposed for mesh wireless networks. Some use conventional routing metrics such as minimum hop, while others consider new routing metrics such as power consumption. To better understand

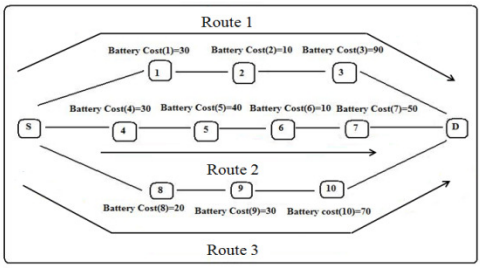


Figure 1: Example of the algorithm

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

their performance in terms of power efficiency, we perform simulations. Fig. 2 demonstrates these steps.

VI. Simulation Settings

Simulations were done using the OPNET v14.5 Simulator and Matlab R2007b to analyze the performance of the proposed metric. OPNET is excellent simulation software. However, there are a number of simple tasks that are often not so simple to do in OPNET. So, we combine the power of Matlab as a backend to OPNET simulations using the Matlab Engine. The proposed metric was implemented in DSR and its performance was compared to the standard DSR protocol that uses the minimum hop metric and OLSR protocol applied to another scenario separately.

A. Network scenario

We first evaluate the various mechanisms in network scenario. The first network consists of 25 mobile nodes equally distributed over a 500 _ 500 meters area. The Rx Group Configuration node is added to speed up the simulation. It is configured to eliminate all receivers that are over 300 meters away (See Fig. 3). Our objective of second scenario (see Fig.4) is to collect OLSR related statistics and analyze them as the network dynamics changes. OLSR is a protocol and uses Multi- point Relay (MPR) optimization for controlled flooding and operations. We will study the network performance as number of MPR nodes change. This is important for OLSR [12-14] deployments. This network has 50 nodes configured to run OLSR. The nodes in the network are grouped in clusters. Nodes in the center cluster are mobile. They move along their trajectories at 50 seconds and stop at ~60 seconds. IP demands are configured between pair of nodes.

B. Simulation Parameters

Table I and II summarize the simulation parameters for first and second scenario, respectively.

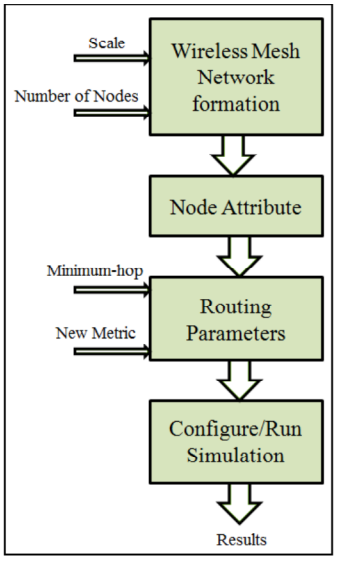


Figure 2: Wireless Mesh Network simulation model

Table I. Simulation parameters for first scenario

Simulation Parameters	Values
Network Area	500m x 500m
Number of Nodes	25
Data Rate (bps)	5.5 Mbps
Operation Mode	802.11b
Simulation Time	10 min

We concentrate on two different situations: a completely static environment and a dynamic environment.

Table II. Simulation parameters for second scenario

Simulation Parameters	Values
Network grid	500×500 meters
Number of Nodes	50
Data Rate (bps)	11 Mbps
Transmission range	300 meters
Operation Mode	Direct sequence
Simulation Time	10 min
trajectory	olsr_move
Parameters set for OLSR	Default values

VII. Routing Discovery Using Dynamic Source Routing

We choose the Dynamic Source Routing (DSR) [11] protocol as a candidate protocol. This section briefly describes the functionality of the dynamic source routing protocol.

- When node S wants to send a packet to node D, but does not know a route to D, node S initiates a route discovery
- Source S floods Route Request (RREQ)
- Each node appends own identifier when forwarding RREQ
- Every node maintains a neighbor information table, to keep track of multiple RREQs
- Destination D on receiving the first RREQ, sends a Route Reply (RREP)
- RREP is sent on a route obtained by reversing the route appended to receive RREQ
- RREP includes the route obtained by reversing the route appended to receive RREQ
- RREP includes the route from S to D on which RREQ was received by node D

Table III shows the Characteristics of DSR protocol.

Table III. Characteristics of DSR

Characteristic	DSR
Routing Philosophy	Reactive
Type of Routing	Source routing
Frequency of Updates	As needed
Worst case	Full flooding
Multiple routes	Yes

A. Route Request (RREQ) Packet for our New Metric

The RREQ packet of the DSR [11] is extended as RREQ of the new metric adding two extra fields, BCF and CL_P. Fig. 5 shows these fields.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

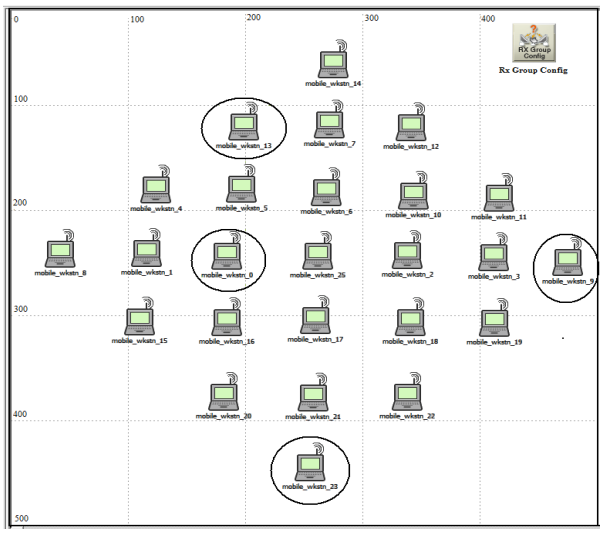


Figure 3: First Network Scenario

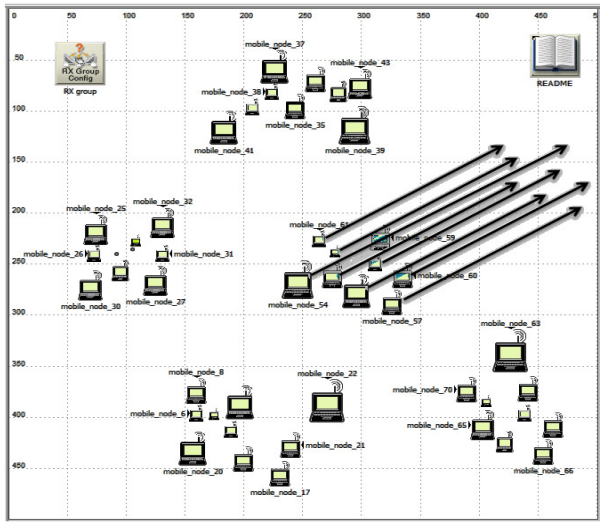


Figure 4: Second Network Scenario

SA	DA	T	ID	TTL	BCF	HOPs	P	CL_P

Figure 5: The RREQ packet

SA (Source Address) field carries the source address of node. DA. (Destination Address) field carries the destination address of node. T (Type) field indicates the type of packet, TTL (Time To Live) field is used to limit the lifetime of packet, by default, it contains zero. BCF (Battery Cost Function) field carries inverse battery capacity of each node. HOP field carries the hop count, initially, this field contains zero value. P (PATH) field carries the path accumulations, when packet passes through a node; its address is appended at end of this field. CL_P field calculates the battery cost (like equation 6) at the end of route.

B. *Destination Node*

In this case with new metric, when the destination receives multiple RREQs it selects the path has the minimum digit that has calculated in CL_P power, i.e. the path with the maximum remaining battery capacity.

VIII. PERFORMANCE STUDY

In first scenario, we epitomize our study on estimating the halt-time, of nodes. The halt-time (Expiration time) expresses how long a node has been active before it halts due to lack of battery capacity. The halt-time of nodes directly affects the lifetime of an active route and possibly of a connection. Then we evaluate the traffic routing received/sent by each node with different transmit power in static and dynamic environments.

In the dynamic environment we used the "random waypoint" model to simulate nodes movement. The motion is characterized by the maximum speed. Each node starts moving from its initial position to a random target position selected inside the simulation area. The node speed is uniformly distributed between 0 and the maximum speed.

In second scenario, we study four states:

A. *MPR count statistics*

This statistic shows the number of nodes selected as MPRs in the network. Initially (0-50 seconds), "MPR count" increases and then converges to 1. In the transient phase, each node has partial information about network topology. Hence each node tries to select MPR based on partial topology information. As nodes receive topology information, MPRs are re-elected and finally converges at steady state. Note that node_56 becomes MPR by default (due to its willingness parameter) and since it is the only required MPR, rest of nodes do not become MPR. Since the transmission range is 300 meters, one hop is required for communication for the nodes at opposite ends of the network. Node_56 being in center cluster has more "reachability" than nodes in edge clusters. For time period after 50 seconds, the nodes in middle cluster starts moving and reach the upper right edge. Since there is no single best candidate for MPR (in terms of reachability), each node finds different MPRs to reach two hop neighbor.

That is why we see higher MPR count. The MPR that is selected in each cluster is the node with its willingness parameter set to high (Node_19, node_28, node_25, node_64). Since there are 5 clusters, 5 MPRs will remain in the steady state (Fig. 6).

B. *TC Traffic sent (bits/sec)*

Topology control (TC) messages are periodically sent out only by MPR nodes in the network. For time period 0-50 seconds, there was only 1 MPR node. After 50 seconds, the number of MPRs in network increases, hence TC Traffic Sent increases (See Fig.6).

C. *Hello Message Sent*

Hello message are periodically sent by each node in the network. It contains the list of neighbors and their quality.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

Statistics "Hello Message Sent (Fig.7)" shows the number of hello messages sent in the network. It does not change even after 50 second time as each node continues sending hello message. However, the "Hello Message Sent (bits/seconds)" statistics changes with node movement. The number of neighbors for each node decreases when the nodes in center cluster moves away, hence the size of each hello message reduces.

D. MPR status

This statistic indicates (Fig.17) if a node is elected as an MP. It generates a square wave graph with values 1 and 0, where value 1 indicates the time when this node becomes an MPR.

IX. Simulation Results

In our simulations, two different route selection schemes are considered: 1) Minimum Hop (MH) and 2) New Metric (NM).

Note: Since a client can forward packets only when its battery capacity is above zero, the value of the cost function will always be finite. Fig. 8 demonstrates the expiration time of nodes and of connections. The expiration times are sorted in ascending order. In Minimum Hop approach, the times of some first nodes exhausting theirs battery are much earlier than that of some last nodes since this metric does not take the battery capacity of each node into consideration and selects route with minimum hop. So, there is no guarantee to extend the lifetime of nodes. But, expiration sequences for New Metric preliminary nodes have the longer lifetime than that of the first nodes in MH because NM chooses the path that has the nodes with proper remaining battery capacity.

Fig. 9 displays the end to end delay of system. Delay of our network is compared with minimum hop which has a more delay of MH metric. The reason of it is, since new metric needs to obtain power's information about its own neighbourhood and it is down periodically in system. So, it takes a few times to gather this information and increase the end to end delay system. To save the information temporary, this method needs a routing table causes the overhead of routing (see Fig.10). But, in this scenario prolonging network operation is more important than other issue.

Fig. 11 and fig. 12 illustrate the routing traffic received and routing traffic sent for static environment for some nodes in different positions respectively (see Fig.3). Transmit power for each node is increased and the average value of routing traffic is recorded. As these figures have shown, the routing traffic decreases when the transmit power of each node increases. Because, when the transmit power of node is little node cannot route the packet better therefore retransmission mechanism occur and cause the traffic.

Fig. 13 represents the average routing traffic received/sent for 25 nodes in the network. (Global statistics)

Fig. 14, fig. 15 and fig. 16 illustrate the routing traffic received, sent and global for dynamic environment respectively.

X. Conclusion

In this paper, we first presented previous work on power-aware routing. Then we proposed a new energy consumption model, chiefly for the mesh clients (because mesh clients have limited battery resources and must consume battery power more efficiently to prolong network operation lifetime), to be used to predict the lifetime of nodes according to current traffic conditions. The main goal of this new metric is to extend the lifetime of each node. Alongside this new work, we simulate the network scenario by using OLSR protocol and we study their statistics. Using OPNET simulator and MATLAB, we implemented OLSR protocol in wireless mesh network and new proposed metric and we compared it with minimum hop mechanism. Finally, we studied the routing traffic vs. transmit power in static and dynamic environment and observed the traffic increased when transmit power decreased.

References

[1] I.F. Akyildiz and X. Wang," A Survey on Wireless Mesh Networks," in IEEE communication Magazine, September 2005.

[2] J. Jun and M. L. Sichitiu, "The Nominal Capacity of Wireless Mesh Networks," in IEEE Wireless Communications, October 2003.

[3] C.-K. Toh, "Maximum battery life routing to support ubiquitous mobile computing in wireless ad hoc networks," IEEE Communications Magazine, vol. 39, no. 6, pp. 138–147, June 2001.

[4] S.Singh, M. Woo, and C. Raghavendra, "Power-aware routing in mobile ad hoc networks," in MobiCom '98: Proceedings of the 4th annual ACM/IEEE international conference on Mobile computing and networking.New York, NY, USA: ACM Press, 1998, pp. 181 190.

[5] WangBo and Li Layuan, "Maximizing Network Lifetime in Wireless Mesh Networks," IEEE Wireless Communications, 2008.

[6] D.K.Kim, J. Garcia-Luna-Aceves, K. Obraczka, J.-C. Cano, and P. Manzoni, "Routing mechanisms for mobile ad hoc networks based on the energy drain rate," IEEE Transactions on Mobile Computing, vol. 2,no. 2, pp. 161–173, 2003.

[7] C.-K. Toh, "Maximum Battery Life Routing to Support Ubiquitous Mobile Computing in Wireless Ad Hoc Networks," IEEE Comm. Magazine, June 2001.

[8] Held, Gilbert, Wireless mesh networks, Taylor & Francis Group, USA, March 2005.

[9] K. Scott and N. Bambos, "Routing and Channel Assignment for Low Power Transmission in PCS," Proc. IEEE Int'l Conf. Universal Personal Comm., 1996.

[10] S. Singh, M. Woo, and C. S. Raghavendra, "Power- Aware Routing in Mobile Ad Hoc Networks," Proc. MobiCom '98, Dallas, TX, Oct. 1998.

[11] David B. Johnson, David A. Maltz, & Josh Broch, "DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks", (2001), Ad Hoc Networking, Addison- Wesley , pp. 139-172

[12] T. Clausen, P. Jacquet, "RFC3626: Optimized Link State Routing Protocol (OLSR)"

[13] A. Qayy , L. Viennot, A. Laouiti, "Multipoint relaying: An efficient technique for flooding in mobile wireless networks", 35t Hawaii International Conference on System Sciences (HICSS'2001)

[14] T. Clausen, C. Dearlove, J. Dean, C. Adjih, \RFC5444: Generalized Mobile Ad Hoc Network (MANET) Packet/Message Format", Std. Track, http://www.ietf.org/rfc/rfc5444.txt

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

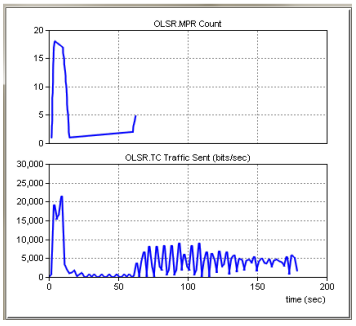


Figure 6. MPR count and TC Traffic Sent

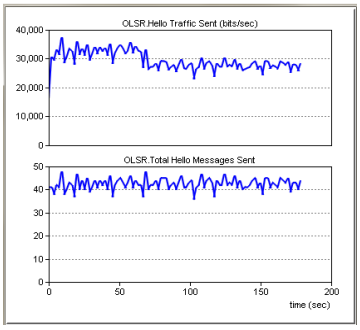


Figure 7. Hello Message Sent

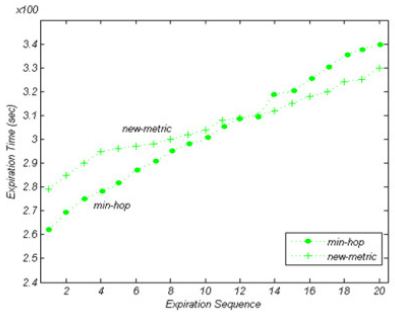


Figure 8. Expiration time vs. expiration sequence

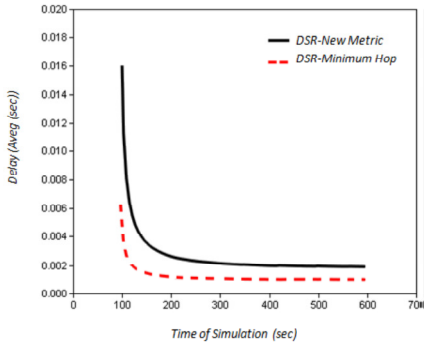


Figure 9. The end-to-end delay

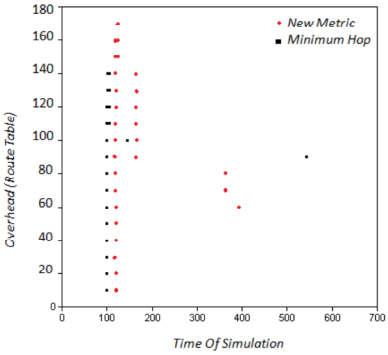


Figure 10. Overhead of routing

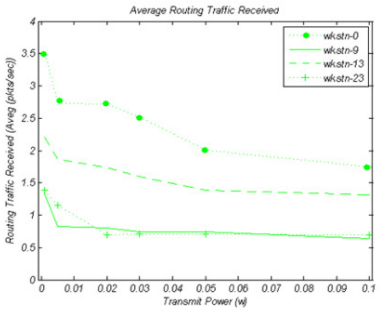


Figure 11. Average Routing Traffic Received for some static nodes vs. Transmit Power

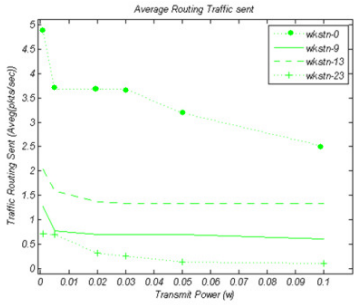


Figure 12. Average Routing Traffic Sent for some static nodes vs. Transmit Power

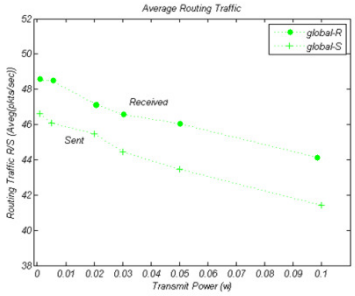


Figure 13. Average Routing Traffic Received/Sent for all static nodes vs. Transmit

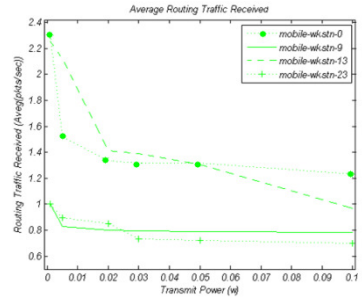


Figure 14. Average Routing Traffic Received for some mobile nodes vs. Transmit Power

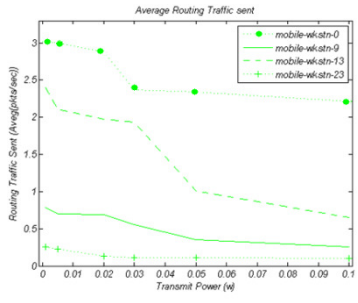


Figure 15. Average Routing Traffic Sent for some mobile nodes vs. Transmit Power

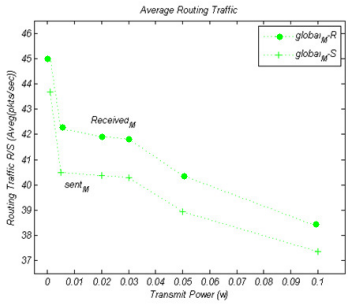


Figure 16. Average Routing Traffic Received/Sent for all mobile nodes vs. Transmit Power

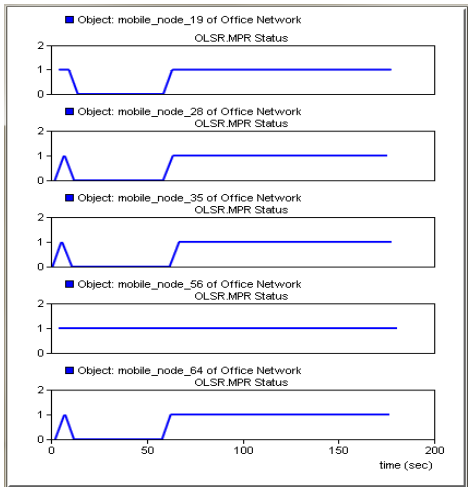


Figure 17. OLSR MPR Status

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

A New Sentinel Approach for Energy Efficient and Hole Aware Wireless Sensor Networks

Dame DIONGUE
Ph.D Student, Department of Computer Science
Gaston Berger University
Saint Louis, Senegal .

Ousmane THIARE
Department of Computer Science
Gaston Berger University
Saint Louis, Senegal .

Abstract—**Recent advances in micro-sensor and communication technology have enabled the emergence of a new technology, Wireless Sensor Networks (WSN). WSN have emerging recently as a key solution to monitor remote or hostile environments and concern a wide range of applications. These networks are faced with many challenges such as energy efficiency usage, topology maintenance, network lifetime maximization, etc. Experience shows that sensing and communications tasks consume energy, therefore judicious power management can effectively extend network lifetime. Moreover, the low cost of sensor devices will allows deployment of huge number nodes that can permit a high redundancy degree. In this paper, we focus on the problem of energy efficiency and topology maintenance in a densely deployed network context. Hence we propose an energy aware sleep scheduling and rapid topology healing scheme for long life wireless sensor networks. Our scheme is a strong node scheduling based mechanism for lifetime maximization in wireless sensor networks and has a fast maintenance method to cover nodes failure. Our sentinel scheme is based on a probabilistic model which provides a distributed sleep scheduling and topology control algorithm. Simulations and experimental results are presented to verify our approach and the performance of our mechanism.**

Keywords-component; energy conservation; lifetime maximization; topology maintenance; insert (key words)

I. INTRODUCTION

Recent technological advances in microelectronics have favored the development of tiny and intelligent embedded devices called sensor nodes that can detect and send relevant information relatively to a given environment. This has led to the emergence of a new technology, Wireless Sensor Networks. A typical Wireless Sensor Network consists of a huge number of tiny sensor with sensing, processing and transmission capabilities [1]. These last decades, wireless sensor technology holds the lead of the stage in several sectors such as environmental monitoring, military surveillance [2], medical diagnosis [3][4], building automation [5][6], industrial automation tasks, etc. In most cases, the area of interest (wireless sensor network's deployment area) is harsh or even impossible to access for human intervention. Therefore, the deployment is most often done by airplane dropping and this may lead to unfair repartition of sensor nodes through the monitored region.

Beside problems related to random deployment, Wireless Sensor Networks are also suffering to many challenges such as data aggregation, routing, security, energy management, topology management, etc. The two later issues are attracted more and more interest from researchers and are typically addressed in this paper. Energy consumption and topology changes are of critical importance regarding Wireless Sensor Networks because the sensor node lifetime is closely related to its battery power and once deployed, they are usually inaccessible to be replaced nor recharged, due to harsh environment. However, the protocol designers should take into consideration these constraints and allow sensor nodes to have sufficient autonomy to organize themselves and cooperate with each other to save their energy. In some types of applications, random deployment is most often used and it does not always guarantee better coverage and rational use of energy. This type of deployment, may issue to energy or coverage holes problems due to unfair repartition of sensor nodes.

In this paper, we focus on node scheduling and propose an energy aware sleep scheduling and fast topology maintenance algorithm for lifetime maximization in wireless sensor networks. Our scheme is based on the Sentinel concept and need to operate in highly dense networks. The proposed scheme consists of two parts, the sleep scheduling procedure that uses nodes redundancy and dynamic probe rate adjustment to take better advantage of the redundancy, and the fast recovery procedure to take into account nodes failure. Since Sentinel scheme operate in very dense networks, it must be coupled with an effective recovery procedure. So that, if a sentinel node fails, whether in the shortest time a spare to take over and maintains the hole. Unlike the scheme proposed in [10] where authors assume an active messaging status for active nodes, here we propose that working nodes use passive messaging to limit the overhead charge. Another major challenge in wireless environment is the problem of collisions. In some case, collision may occur and cause activation of multiple nodes in a single area. To solve this problem, we use a disabling procedure, called activity withdrawal algorithm, between active nodes based on proximity and activity duration weight. When we have two active conflicting nodes, the disabling procedure permit to select the older one to ensure the monitoring task and put the other node in sleep mode.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

The remainder of this is organized as fellow. Section II describes some related works in the literature. Section III details our model description and the scheduling problem definition. Section IV makes an overview to the proposed scheme. Section V shows the simulations and experimental results. Finally, section VI provides conclusion and future works.

II. Related Work

A. Energy conservation

Wireless sensor devices are very constrained in term of battery power. Sensor nodes are non rechargeable battery operating devices and generally deployed in often inaccessible environment like forests for fire or pollution detection, sea for tracking some species, battlefields for enemies tracking, etc. Then, the only way to keep alive the network for longer time is to efficiently manage the battery power usage. However, many mechanisms, algorithms and protocols have been proposed in routing, clustering, data agregation, security, mobility, and especially coverage and connectivity areas.

Virmani and al. propose an energy efficient data agregation protocol based on nodes clustering [7]. Their protocol relies on the reduction of the distance between communication nodes. In the same vein, Murthy and al. Proposed a crosslayered clustering protocol [8]. We find that most of the works on lifetime maximization deal at the same time with the coverage problems. In [9][10], the authors use the distribution of the interest area into several cover sets (disjoint and/or nondisjoint) to efficiently rationalize the energy usage.

Other works focus on lifetime maximization based on energy efficient coverage and state management mechanisms. Achieving this assumes that nodes cooperate with each other to make distributed decisions on the choice of active subset; hence the need to synchronize the whole network activities [11][12]. This approach requires some processing and communication cost at each node. However, it is more scalable and more flexible for nodes failures. Ye and al. proposed in [12] a probing environment and adaptive sensing mechanism. They assume to activate the minimum set of nodes, over a highdensity sensor network, that can provides the monitoring of the interest area and put all the redundant nodes in sleep mode. In PEAS [12], authors proposed energy conservation by maintaining all working nodes by a minimum distance c. The asleep nodes may wake up after a random period and check their vicinity (for a radius c) by sending broadcast messages. They will enter on-duty mode only if they receive no replies from working nodes; otherwise they will stay on off-duty mode. Their solution offered a crucial benefit in term of energy consumption and guarantee an asymptotic network connectivity. But authors assume that working nodes never go back to sleep, which may result in redundant working nodes when collisions occur at the probe requesting/replying steps.

B. Topology Maintenance

A Wireless Sensor Network well-functioning strongly depends on: (i) a good coverage of the interest area to retrieve relevant information, (ii) a good connectivity between sensor nodes to better relay information toward the Sink node, (iii) and also a good energy management policy for a long life network. However, the deployment strategies (deterministic or random) have a great influence on above criteria. Ideally, a deterministic deployment is desirable, but in most cases the monitored region, for example battlefield, is difficult or dangerously accessible and thus, a random deployment remains the only possible alternative. This deployment method often leads collateral problems such as sparse or not at all covered areas. Several solutions has been proposed in the literature in order to solve the related problems to the network topology changes. And these solutions can be classified according three approaches: node adaptation, link adaptation and mobility (mobile sensor node or robot) [13]. Node adaptation techniques are often based on: (i) clustering which propose the network to have an hierarchical organization, (ii) set cover computation which organize the network into multiple subset where each one can cover the whole network for a period of time, (iii) and lastly node scheduling technics that relies on deploying redundant nodes and schedule their activity. Gupta and al. [14] use a node scheduling technique for topology healing and a probabilistic approach to determine the coverage redundancy degree. They schedule nodes activities on the one hand to save energy and also ensure a better coverage. Always in the same direction, Corke and al. propose in [15] two algorithms. The first algorithm uses neighbors informations to detect failed nodes and determine hole location. The second algorithm uses routing informations to detect a hole from a distance and try to maintain the routing path. Their solution require that nodes keep state informations into memory. Other solution [16][17] in the literature use another approach for topology healing, link adaptation technic and this consist of adapting communication parameters and exchanging neighbors informations. Others use mobility [18] to solve the holes problems related to coverage/energy. Works in [19][20][21], opt for an additional deployment of mobile nodes (generally robots with GPS) to maintain the coverage. These solutions offer effective holes healing but generate a high network load added to that gluttony in energy of the GPS module.

In this paper, we opted for a scheduling based solution rather than deploying additional mobile nodes. Because, energy should be well tuned in Wireless Sensor Networks. However, mobility based solutions, in addition to the expensive costs of equipments, use GPS, which is very energy intensive. And also, mobility is often not easy or not at all applicable to some regions because of their relief.

III. Model and Problem Description

A. Network model and problem description

We first present in this section some keys definitions and properties related to our proposed algorithm.

Definition 1: Sentinel Network Design

Here we assume a flat network with a huge number of sensor node uniformly deployed in an interest area (a network with a

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

high density of sensor nodes). And all nodes are initially in sleep mode for a while. When wakes up, node probe their vicinity looking for a sentinel (an active node that stands in guard for a dedicated area). If the probing operation is positive i.e. a sentinel node responds by sending a probe reply message, it turn back to sleep mode else it starts the guard round.

Definition 2 : Redundant Node Sleep Scheduling

We consider a sensor network with a huge number of nodes uniformly deployed in an interest area. The concept Redundant Nodes Sleep Scheduling (RNSS) consist of putting on off-duty all the redundant nodes and just let a minimum set of sentinel nodes that can ensure the require monitoring. In [22], authors explain the concept of completely redundant node. Therefore, according to figure 1, node n_2 (node n_2 's communication range is represented with dashed line) will be on off-duty mode because its area is covered by nodes s_1; s_5; s_6 and s_{10}. So n_2 can now direct itself to sleep for t_s seconds.

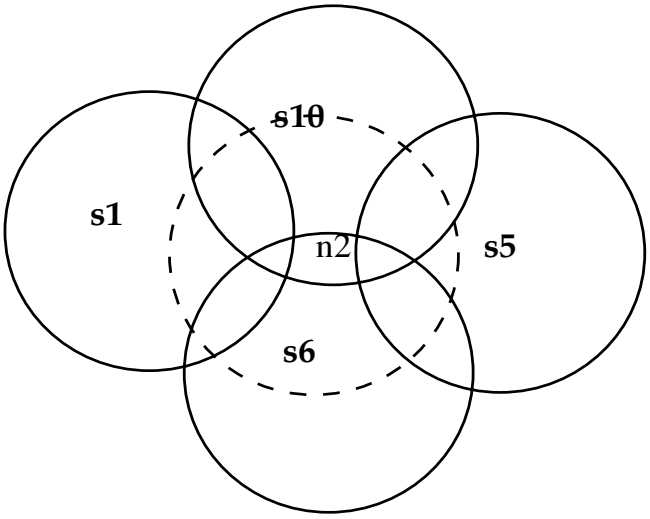


Figure 1. Complete coverage redundancy

Definition 3 : Strongly Connected Active Nodes

In our designed scheme, two active nodes must keep far from each other with a distance threshold $\delta = 2\mathrm{Rs}$. Rs represent a node sensing radius since it consist of a unit disk. And we consider the communication radius of each is R_c with $Rc \geq Rs$. This will permit to design a network in which all active nodes share better connectivity between them. We control nodes connectivity by adjusting the distance threshold δ, then this will permit us to explore better coverage degree with the scheme.

Our aims on this paper are to: (i) minimize the subset of sentinel nodes (on-duty nodes) use to monitor the interest area; (ii) minimize the energy usage at each sentinel sensor node; (iii) and finally design a fast topology recovery procedure. To do this, we will deem the deployment of a dense network (like in Definition 1) creating a high redundancy. Thus we propose to exploit that redundancy by activating the minimum subset of sensor nodes for the monitoring and put the rest in reserve and then give sufficient autonomy to reserved nodes to distributely manage their sleep time and adjust it with the probability of failure.

B. Sentinel Scheduling Problem

The random deployment often causes an unbalanced distribution of nodes through the monitored region. Then, if an active node fails by battery depletion or anything else, the area which was covered by that node will remains unmonitored (creation of coverage hole). And so, all the events that occur there, will pass unnoticed. Hence, to solve this problem, we consider Ω, the population of sensor nodes uniformly deployed in the interest area. And nodes have sufficient autonomy to organize themselves and select a minimum subset S where $S \subset \Omega$ of sentinel nodes. Hence, the subset $\bar{S}$ defined by $\bar{S} = \Omega - S$ falls into off-duty mode to conserve the energy. And finally nodes execute an algorithm that stands on a probabilistic scheme to control the off-duty nodes' wakeup.

IV. SENTINEL SCHEME

A. Node state transition

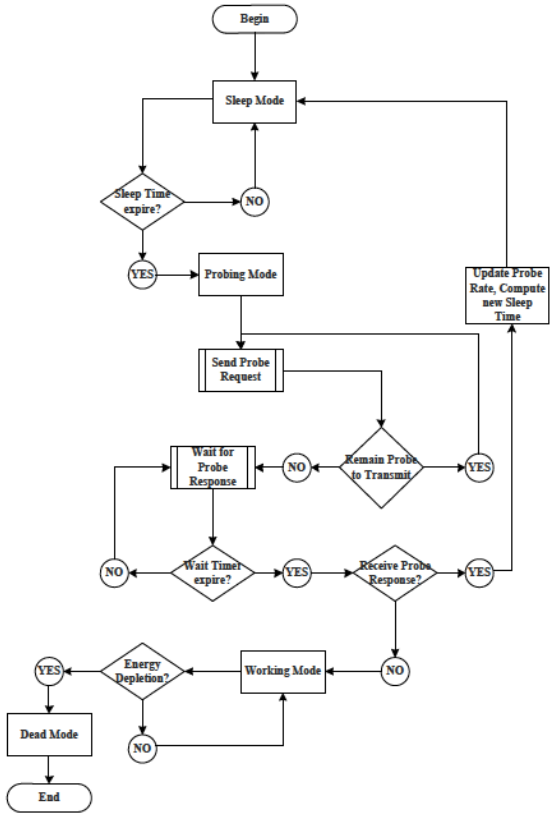


Figure 2. Sentinel node's state transition algorithm

We consider that a sentinel node can be in one of the four following states: sleeping, probing, working or dead.

Sleep mode: The sleep mode corresponding to node's initial state, where they turn off their radio module. We chose to turn off at sleep mode only the radio module, because it is difficult or impossible to put a node completely off-duty.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

Probing mode: The second mode in which a sentinel node can be is the probing mode where it has the ability to send/receive only control messages to/from its neighbors. From that state, a sentinel node can be active either go back to sleep mode.

Active mode: A node goes into active mode if and only if it detects no active neighbor. However, it starts to fulfill its role of sentinel node, that is say, to continuously monitor its dedicated area until the exhaustion of its battery or the reception of a probe reply from an older sentinel node (see activity withdrawal algorithm). Thus, the possible state transition from this mode is either going back to sleep or the death of the sensor.

Dead mode: Finally, the dead mode often characterizes sensor node's energy depletion (total battery exhaustion). This may also be due to a dysfunction of hardware component like sensing unit, communication module, etc.

B. Sleep Scheduling Algorithm

For a suitable energy usage, we designed a scheduling algorithm that permit to select just a minimum set S of onduty nodes to ensure the monitoring task. Then all the other nodes (redundant nodes) will be left on off-duty mode (Sleep mode) that is say, they turn off their radio module. The onduty subset selection (sentinel nodes selection) is done by the following rule "the first nodes that wakes up and find no other one in its vicinity, stands guard i.e. stands as sentinel node". Hence, all the nodes are initially deployed in sleep mode and each sensor node must be asleep for $ts\_{initial}$. After the $ts\_{initial}$ timer expire, node should probe their vicinity by sending probe request messages to look for an active neighbor. After several series of tests, we then sets the probe reply wait timer, t_w , to 1 second. After the t_w timer expire with no replies received, it immediately enter in active mode to monitor its vicinity. In case the probing node receives a reply from its neighbors, it check first if the responder is not far away from the distance threshold in respect to *SCAN* property. If *SCAN* is verified, node update its probe rate according to theorem 2 and then computes its new sleep time (theorem 1). Else, the node ignore the message and start its activity.

Theorem 1: *The Sleep Time Computation*

The wake-up timer of a given node is computed with a distributed scheme using the Weibull distribution probability and node's probe rate. The Weibull distribution is most suitable for our design because it permit to adjust node's sleep time when needed. Experience shows that electronic devices failure rate grows over time and therefore, the Weibull distribution will permit to compute decreasing sleep time over simulation or over network's operating time. The Weibull parameters i.e. scale parameter λ and the shape parameter β are chosen as follows: the Weibull scale parameter represents node's probe rate and is a function of time while the shape parameter is a value selected from {1.5, 2.0, 3.0}. We started the shape parameter's values at 1.5 because if β = 1.0, we have an exponential distribution.

Proof : We suppose that *R*, uniformly generated in range [0,1[, is the probability of awakening of a given node denoted by *X* and obtained by the Weibull cumulative distribution function. We aim to determine *t* such that :

$$R = 1 - F(t) = 1 - P[X \leq t] = 1 - \int_0^t f(u)du$$

Therefore, we have :

$$R = 1 - \left(1 - e^{-\left(\frac{t}{\alpha}\right)^{\beta}}\right)$$

$$\Rightarrow R = e^{-\left(\frac{t}{\alpha}\right)^{\beta}} \quad (1)$$

And then, we deduce from equation (1) a node's sleep time t_s by applying the logarithm :

$$lnR = \ln\left(e^{-\left(\frac{t}{\alpha}\right)^{\beta}}\right) = -\left(\frac{t}{\alpha}\right)^{\beta}$$

$$t_s = \alpha \ln\left(\frac{1}{R}\right)^{1/\beta} \quad (2)$$

Where $\alpha = 1/\lambda$ and β are respectively the Weibull scale and shape parameters. We define that λ represents a node's probe rate [12].■

Another significative contribution presented in this paper is that nodes are sufficiently autonomous for updating their probe rate used to calculate the sleep time. For this, we use the Weibull hazard function $h(t)$. And unlike PEAS and LDAS, here nodes have no need to keep neighbor information for the scheduling procedure. Some solutions in the literature use neighbors informations to take some decision. However, we propose to dynamically update nodes probe rate and this is done independently from neighbors informations (refer to Theorem 2). Thus our scheme scheme is designed to be completely distributed.

Theorem 2 : Dynamic probe rate adjustment

Before a sleep round, each node must compute its new probe rate based on the network's lifetime and its old probe rate. This is done at each node independently from its neighbor.

Proof : Using the Weibull hazard function we obtain from the survival function, we have :

$$h(t) = \lim_{\Delta t \to 0} \frac{1}{\Delta t} P(t < X \leq t + \Delta t | X > t)$$

$$h(t) = \frac{f(t)}{1 - F(t)}$$

Then, we have :

$$h(t) = \frac{\beta}{\alpha}\left(\frac{t}{\alpha}\right)^{\beta - 1}$$

And $h(t)$ represents the new probe rate ($h(t) = \lambda(t)$). ■

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

Input: s_i, s_j: two active nodes;
$d(s_i, s_j)$: distance between nodes s_i and s_j;
$a.s_i, a.s_j$: activity duration of node s_i respectively node s_j;
δ: distance threshold between two active nodes;

initialization: nodes receive probe replies;

if $d(s_i, s_j) \leq \delta$ **then**
break ;
else
if $a.s_i \leq a.s_j$ **then**
Node s_i computes a new sleep timer t_s;
$a.s_i \leftarrow 0$;
Node s_i turns back to sleep mode for t_s;
else
Node s_i ignore the received reply;
end
end

Algorithm 1 : Activity withdrawal algorithm

C. Activity withdrawal algorithm

Due to collisions, active nodes may be conflicting. And to solve this problem, we introduce an activity withdrawal procedure Algorithm-1. When they receive probes request messages, sentinel nodes may response by sending a probe reply message. Probe replies include the sender's coordinates *(x; y; z)* and it activity age $a.s_i$. In case where there are two or more conflicting active nodes, they all execute the withdrawal algorithm. Let us consider the scenario in Fig.3. Let us consider in this example a domain with the nodes n_0, n_4, n_5, n_{10} and n_{20} all initially in sleep mode. They have all sleep timer randomly generated according to the Weibull distribution. Thus, nodes can wake up at different dates. Suppose that node n_5 wakes up first and finds that there is no active neighbors, so it immediately starts its activity ($n_5 \rightarrow s_5$). After a while, the other nodes can wake up and scan the vicinity by sending probes request messages. The sentinel node s_5 by receiving probes request, will responds to its neighbors. His response may include its position and its activity duration (elapsed time since the beginning of its activity until the response). Nodes that hear node s_5 's response will check first if the *SCAN* property is verified. If yes, they update their probe rate and then compute a new sleep timer; otherwise they ignore the message. Since probes messages are broadcasted and the node's wake up are not synchronized, it is possible that collision occur and thereby prevent some messages reaching their destination. The node n_0 will wait until its t_w expires to start its activity. Thereby, we will have two sentinel nodes in the same area. Thus, to solve this problem, we introduce the activity withdrawal algorithm (algorithm 1) that permit to disable the youngest sentinel node.

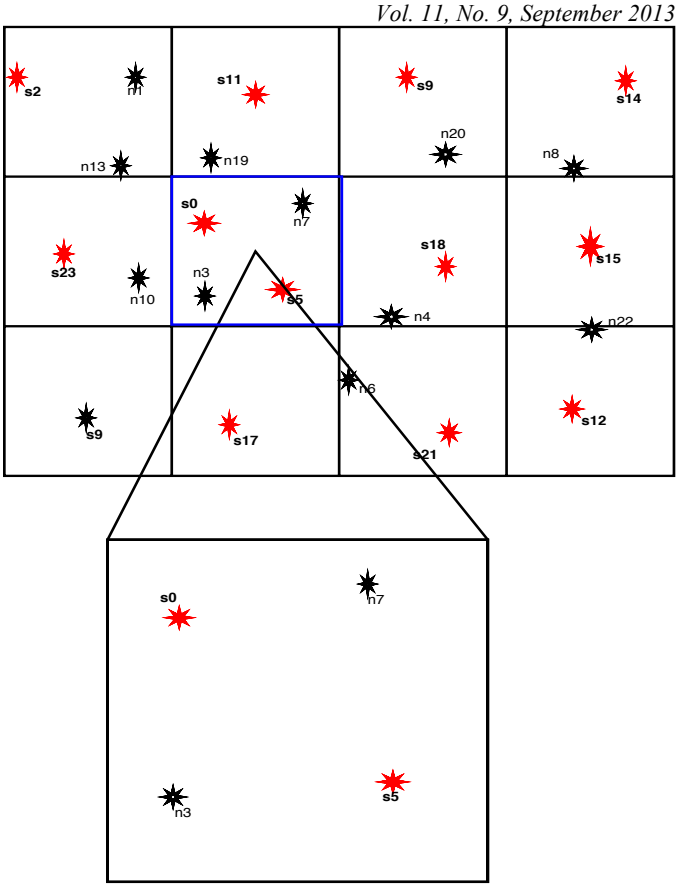


Figure 3. An example to illustrate redundant active node scenario

D. Topology healing procedure

As detailed in Fig. 2, after deployment, a subset of sentinel nodes monitor the region of interest until their energy exhaustion. When a sentinel node fails, one of its neighbors in the reserve subset will wakes up to fill the leaved hole. For more clarity, let us consider the scenario in fig. 4. Initially we have S = {S_0; S_1; S_8; S_{13}; S_{15}; S_{20}} and these nodes monitor the interest region until energy depletion. After a while, sentinel nodes S_0 and S_8 fail and the sentinel subset becomes S = {S_1; S_4; S_7; S_{12}; S_{13}; S_{15}; S_{20}}. Looking at this subset S, we find that there are four new sentinel nodes that is say S_4; S_7; and S_{12} . Without such redundancy, coverage holes can be created with the loss of nodes over time. As in the example of the Fig. 5, S_2 and S_4 die and leave uncovered their dedicated area. Since there is no node in reserve to compensate for the vacuum, the only alternative is the deployment of mobile nodes that requires a GPS guidance.

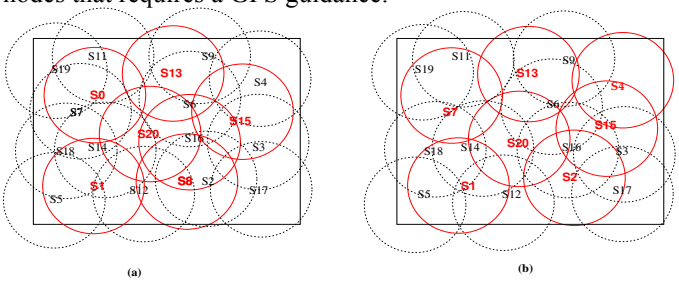


Figure 4. Coverage hole maintenance in redundant deployment scenario

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

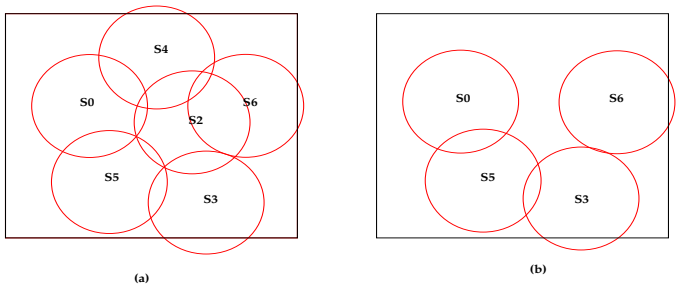


Figure 5. Coverage hole creation with non redundant deployment scenario

V. PERFORMANCE EVALUATION

In this section, we will evaluate our algorithm by measuring the control overhead charge and by comparing it with other algorithm in term of energy usage efficiency. To show that our algorithm is energy efficient, it will be compared to PEAS algorithm. Our scheme will be compared to that in [12] with performance ratio.

A. Simulation model and parameters

We have built a distributed node scheduling algorithm to perform network lifetime in wireless sensor networks. We simulate our scheme using Castalia [1], an OMNeT++ [2] framework designed for wireless sensor networks. For experimentations, we deployed uniformly the sensor nodes in a flat network. Sensor nodes are *2AA* battery equipped and are randomly deployed, initially in sleep mode, in a square field of *50 meters x 50 meters*. To be close to the reality, we assume that channel condition is not perfect and nodes sensing range is defined to *10 meters* ($\delta \leq 2R$ *ie* $\delta \leq 20$ *meters*). So that the probability that collision occur is not zero. Then to avoid much overhead processing, we choose small control messages (*25 octets* by default) to ensure nodes communications.

B. Energy efficiency evaluation

Fig. 6 shows an evaluation of the average energy consumption with different values of β parameter. And we can see that varying the β parameter has no major influence on the energy consumption so on the network lifetime.

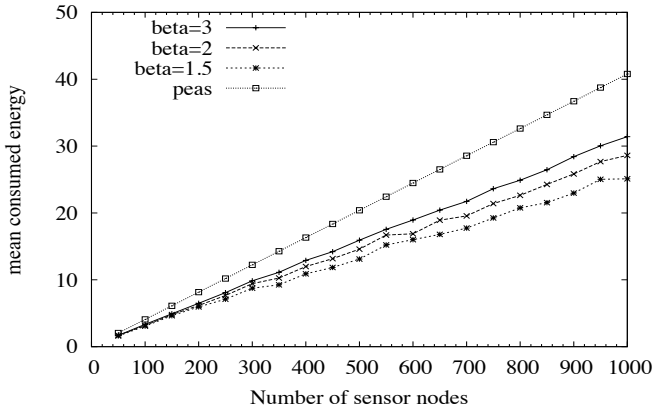


Figure 6. Average energy consumption by varying β parameter

[1] http://castalia.research.nicta.com.au

[2] www.omnetpp.org

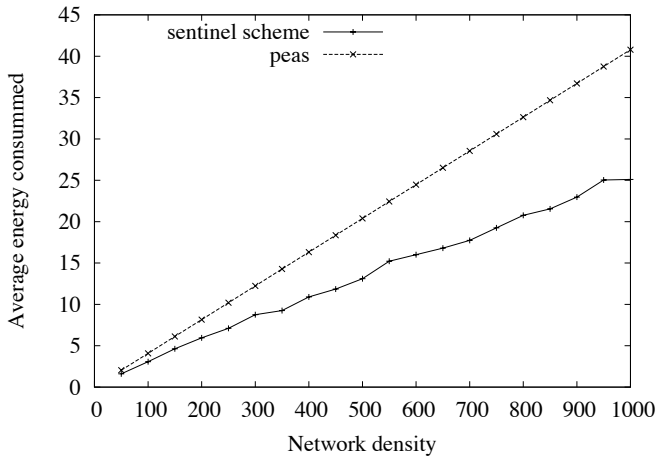


Figure 7. Average energy consumption between Sentinel scheme and PEAS

We choose to vary that parameter because it permit a generalization of some other probabilistic distributions such as Exponential ($\beta = 1$) or Rayleigh distribution ($\alpha = 1$, $\beta = 2$). We simulated the networks for 6000 seconds and we measured the average energy spent by the whole network and finally compared it with results from PEAS. From our simulation results, we make three observations that show that our scheme perform better performance and match to analytical predictions. First, we assess our scheme and comparing it with PEAS algorithm. And Fig. 7 shows that our proposed sentinel scheme achieve better performance than PEAS [12]. The expected average energy spent falls considerably when our algorithm is compared to PEAS and we note that our algorithm enables lower energy consumption with a ratio of ' 36% of the total energy consumption. Second, we see beyond energy efficiency, that our solution permit to take into account the recurring nodes failure by dynamically adjusting nodes sleep timer to tend toward zero over time. Because nodes robustness fall over time and the probability of components failure become more important. And finally at our third observation, we see that our sentinel scheme support network scalability. In spite of all the computations are distributed in our scheme, Fig. 7 shows that growing the network density have not much more impact in the expected energy spent. The curves in the Fig. 6 show that the average energy consumption increases slightly with the number of network nodes. This is explained by the fact that the number of reserved nodes (nodes that probe their vicinity looking for a sentinel node) increases with the network density. And they often need to wake up themselves and check the presence of a sentinel node in the neighborhood, and these consumes some amount of energy due to probe messages exchanged.

C. Rapid maintenance evaluation

Our model is based on the probabilistic Weibull distribution to control sleeps nodes reserves. We applied a dynamic update of the Weibull scale parameter that is to say, the probing rate of nodes. Fig. 8 shows the evolution of the probing rate that increases as a function.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

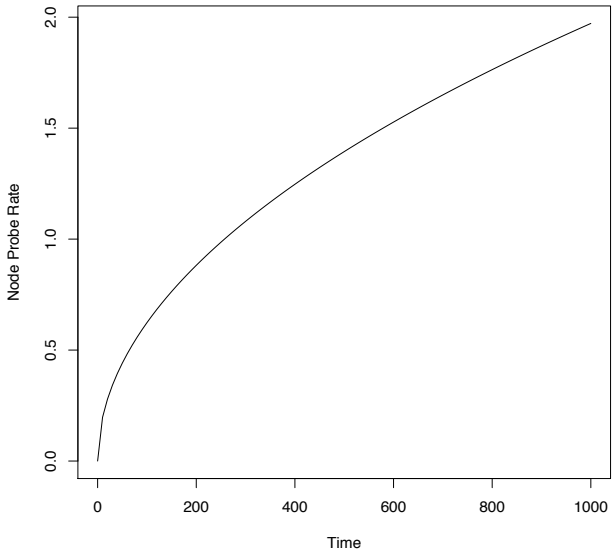


Figure 8. Node's probe rate adhustment over time

Once the probing rate obtained nodes use it to determine their sleep time. The standby time is inversely proportional to the probing rate (see figure 9). The nodes are decreasing their waking function of time and that in order to quickly replace a sentinel node that fails. Because, as we have raised above, the probability of failure of nodes increases over time.

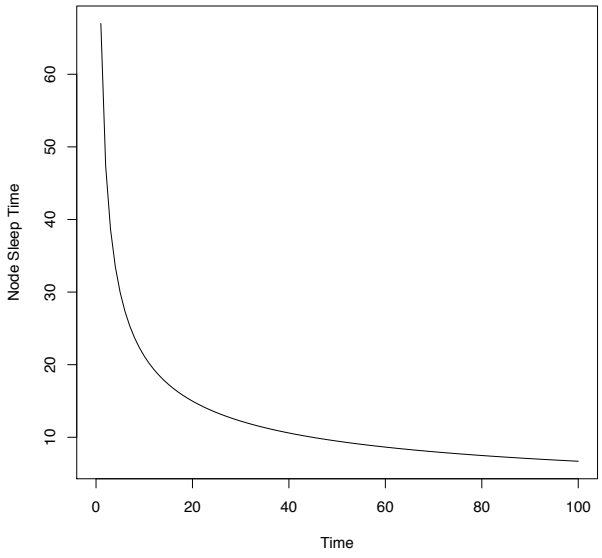


Figure 9. Node's sleep time updating over time

D. Overhead Control

In our scheme, each node autonomously manages its sleep/wake up timer. And this is possible due to control messages exchanged during the probing step. This communication between nodes may generate flow overhead (Fig. 10 and Fig. 11) and then affect the network performances. Figure 10 shows that our scheme faced to the collision problem. And before, we ran samples simulation to assess collision impact through nodes communications. We first configure our simulation by ordering each probing node to sent one probe request to scan its vicinity and we see that after a few time over 60% of nodes passed to active mode. It mean that nodes does not receive probe request and hence consider that there are no sentinels. And quiet the same scenario is obtained when the number of sent probe request at each node is fixed to two messages per probing round. Fig. 11 shows that there is a big gap between the number of sent probe request and the received probe. This shows that most of the messages sent by probing nodes does not reach their destination i.e. the sentinel nodes. We observed that the probability that collisions occur growth proportionally with overhead i.e. with network's size. Comparing Figures 6 and 7, we note a proportionality between the number of received probes messages and the number of received probe responses that is to say that every one sentinel node that has received a probe message, it effectively sent a reply that came to the destination.That's why we fixed after several experiences, the number of probe request at each node to 3 messages and introduce the withdrawal algorithm to face the redundant active nodes problem due to collisions.

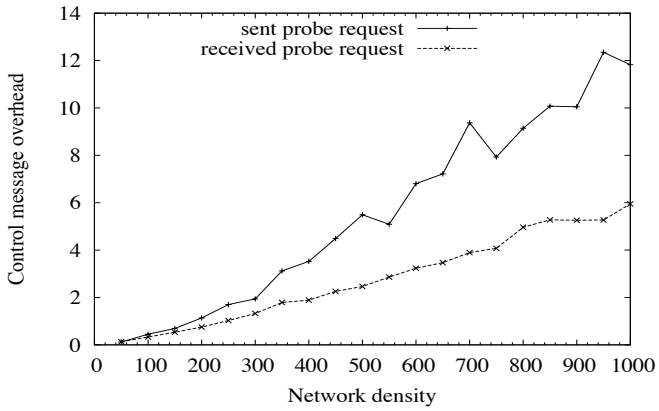


Figure 10. Control message overhead : sent probe requests vs. received probe requests

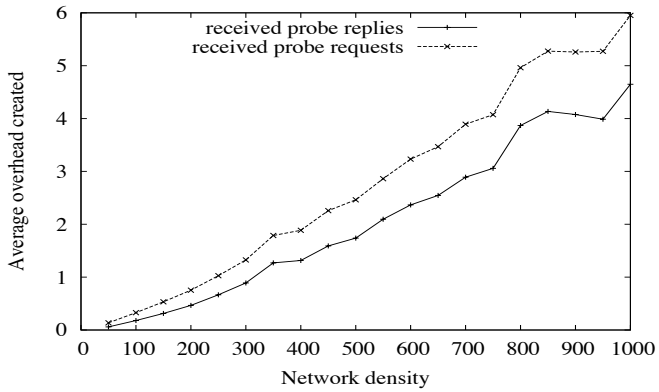


Figure 11. Control message overhead :received probe requests vs. received probe replies

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

VI. CONCLUSION AND FUTURE WORKS

In this paper we analyze the design, implementation and experimental evaluation of a new scheduling algorithm based on sentinel scheme. The sentinel scheme exploit the redundancy offered by the cheap tiny sensor devices to ensure network accuracy and then prolong its lifetime. We also propose an energy aware sleep scheduling and rapid topology maintenance algorithm based on a sentinel scheme to enhance wireless sensor networks' lifetime. Our proposed scheme based on scheduling redundant nodes sleep periods, have several strengths. It permit first to schedule redundant nodes according to the Weibull distribution and guarantee an energy efficiency. And secondly, the Wiebull distribution helps to achieve autonomous operating nodes by dynamically adjusting node sleep time to take into account frequent nodes failure. Because, unlike in PEAS, our algorithm permit to dynamically adjust the nodes probe rate which is used to compute the sleep timer and no more need to keep into memory neighbors informations. Simulation results shows the robustness of our scheme by achieving energy efficient, scalable and fault tolerant algorithm. Through experimental figures, our proposed sentinel scheme presents better performances compare to PEAS.

Our future works include analyzing our scheme under the coverage problem and evaluate the lifetime evolution. And this will add more functionalities to our scheme and will make it more suitable for long life wireless sensor networks.

VII. REFERENCES

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, et E. Cayirci, « Wireless sensor networks: a survey », Comput. Networks, vol. 38, no 4, p. 393‑422, 2002.

[2] T. Bokareva, W. Hu, S. Kanhere, B. Ristic, N. Gordon, T. Bessell, M. Rutten, et S. Jha, « Wireless sensor networks for battlefield surveillance », in Proceedings of the land warfare conference, 2006.

[3] H. Alemdar et C. Ersoy, « Wireless sensor networks for healthcare: A survey », Comput. Networks, vol. 54, no 15, p. 2688‑2710, 2010.

[4] H. Yan, Y. Xu, et M. Gidlund, « Experimental e-health applications in wireless sensor networks », in Communications and Mobile Computing, 2009. CMC'09. WRI International Conference on, 2009, vol. 1, p. 563‑567.

[5] J. K. Hart et K. Martinez, « Environmental Sensor Networks: A revolution in the earth system science? », Earth-Sci. Rev., vol. 78, no 3, p. 177‑191, 2006.

[6] S. Hussain, S. Schaffner, et D. Moseychuck, « Applications of wireless sensor networks and RFID in a smart home environment », in Communication Networks and Services Research Conference, 2009. CNSR'09. Seventh Annual, 2009, p. 153‑157.

[7] D. Virmani et S. Jain, « Decentralized Lifetime Maximizing Tree with Clustering for Data Delivery in Wireless Sensor Networks », Int. J. Comput. Sci., vol. 8.

[8] J. K. Murthy, S. Kumar, et A. Srinivas, « Energy efficient scheduling in cross layer optimized clustered wireless sensor networks », Int'l J. Comput. Sci. Commun., vol. 3, no 1, p. 149‑153, 2012.

[9] M. T. Thai, F. Wang, D. H. Du, et X. Jia, « Coverage problems in wireless sensor networks: designs and analysis », Int. J. Sens. Networks, vol. 3, no 3, p. 191‑200, 2008.

[10] D. Zorbas, D. Glynos, P. Kotzanikolaou, et C. Douligeris, « Solving coverage problems in wireless sensor networks using cover sets », Ad Hoc Networks, vol. 8, no 4, p. 400‑415, 2010.

[11] S. Sengupta, S. Das, M. Nasir, A. V. Vasilakos, et W. Pedrycz, « An Evolutionary Multiobjective Sleep-Scheduling Scheme for Differentiated Coverage in Wireless Sensor Networks », Syst. Man Cybern. Part C Appl. Rev. IEEE Trans., vol. 42, no 6, p. 1093‑1102, 2012.

[12] F. Ye, G. Zhong, J. Cheng, S. Lu, et L. Zhang, « PEAS: A robust energy conserving protocol for long-lived sensor networks », in Distributed Computing Systems, 2003. Proceedings. 23rd International Conference on, 2003, p. 28‑37.

[13] C. Sengul, A. C. Viana, et A. Ziviani, « A survey of adaptive services to cope with dynamics in wireless self-organizing networks », ACM Comput. Surv. CSUR, vol. 44, no 4, p. 23, 2012.

[14] H. P. Gupta, S. V. Rao, et T. Venkatesh, « Sleep scheduling for partial coverage in heterogeneous wireless sensor networks », in Communication Systems and Networks (COMSNETS), 2013 Fifth International Conference on, 2013, p. 1‑10.

[15] P. Corke, R. Peterson, et D. Rus, « Finding holes in sensor networks », in the IEEE Workshop on Omniscient Space: Robot Control, 2007.

[16] J. Ben-Othman, K. Bessaoud, A. Bui, et L. Pilard, « Self-stabilizing algorithm for efficient topology control in Wireless Sensor Networks », J. Comput. Sci., 2012.

[17] J. Li, Y. Wu, J. A. Stankovic, S. H. Song, Z. Zhong, T. He, B. W. Kim, et S.-S. Joo, « Predictive dependency constraint directed self-healing for wireless sensor networks », in Networked Sensing Systems (INSS), 2010 Seventh International Conference on, 2010, p. 22‑29.

[18] B. Wang, H. B. Lim, et D. Ma, « A survey of movement strategies for improving network coverage in wireless sensor networks », Comput. Commun., vol. 32, no 13, p. 1427‑1436, 2009.

[19] A. Ghosh, « Estimating coverage holes and enhancing coverage in mixed sensor networks », in Local Computer Networks, 2004. 29th Annual IEEE International Conference on, 2004, p. 68‑76.

[20] B.-R. Kao et K. R. Lai, « A Multi-hop Dynamic Connectivity and Maintenance Algorithm for Wireless Sensor Networks », in Advanced Information Networking and Applications (AINA), 2011 IEEE International Conference on, 2011, p. 404‑410.

[21] H.-Y. Shiue, G.-J. Yu, et J.-P. Sheu, « Energy hole healing protocol for surveillance sensor networks », in Workshop on WASN, 2005, vol. 2005.

[22] K. Wu, Y. Gao, F. Li, et Y. Xiao, « Lightweight deployment-aware scheduling for wireless sensor networks », Mob. Networks Appl., vol. 10, no 6, p. 837‑852, 2005.

AUTHORS PROFILE

Dame DIONGUE is currently a Ph.D. student in computer science at LANI/Department of Computer Sciences of Gaston Berger University. He received a B. Sc. Degree in Computer Sciences from University Cheikh Anta Diop of Dakar, Senegal in 2007 and a M. Sc. Degree from University Cheikh Anta Diop of Dakar, Senegal in 2011. His areas of interest are wireless sensor networks, wireless mesh networks and optimization.

Ousmane THIARE. Received a PhD in computer science (Distributed systems) at 2007 from the university of Cergy Pontoise, France. He is an Associate Professor in Gaston Berger University of Saint-Louis Senegal. He has been author and co-author of published papers in several journals and recognized international conferences and symposiums.

Design of Generic Framework for Botnet Detection in Network Forensics

Sukhdilpreet Kaur
Computer Science and Engineering
Punjabi University Regional Centre for Information Technology and Management
Mohali, India .

Amandeep Verma
Computer Science and Engineering
Punjabi University Regional Centre for Information Technology and Management
Mohali, India .

Abstract— **With the raise in practice of Internet, in social, personal, commercial and other aspects of life, the cybercrime is as well escalating at an alarming rate. Such usage of Internet in diversified areas also augmented the illegal activities, which in turn, bids many network attacks and threats. Network forensics is used to detect the network attacks. This can be viewed as the extension of network security. It is the technology, which detects and also suggests prevention of the various network attacks. Botnet is one of the most common attacks and is regarded as a network of hacked computers. It captures the network packet, store it and then analyze and correlate to find the source of attack. Various methods based on this approach for botnet detection are in literature, but a generalized method is lacking. So, there is a requirement to design a generic framework that can be used by any botnet detection. This framework is of use for researchers, in the development of their own method of botnet detection, by means of providing methodology and guidelines. In this paper, various prevalent methods of botnet detection are studied, commonalities among them are established and then a generalized model for the detection of botnet is proposed. The proposed framework is described as UML diagrams.**

Keywords- Network forensics, Botnets, Botnet detection methods, class diagrams, activity diagram.

I. INTRODUCTION

Cyber crime is a huge problem these days. In past few years many researchers have done research on network forensics to reduce the cyber crime. Network forensics is the forensic science that investigates the network traffic and analyzes it for the detection of network attacks. It also tries to find out the source of attack [1]. Botnet is one of the network attacks. It is a network of infected machines called Zombies that have their own life cycle. A controller called botmaster controls Botnets. There is a need to detect the attacks and to prevent them. Detection methods detect and prevent these attacks and try to find out the source of attacks. Many methods of botnet detection are available in literature that are broadly classified into two categories Honeynet based [2] and Passive network traffic monitoring based [3]. Passive network traffic monitoring methods include Botnet Detection Through Fine Flow Classification [4], Detecting Botnets Through Log Correlation [5], Detecting Botnets with Tight Command and Control [6], Botnet Detection by Monitoring Similar Communication Patterns [7], DNS based [8], Data mining based, anomaly based and signature based [9]. All the methods have their specific framework but the generic framework is missing.

In the present study, the focus is around the design of generalized model for botnet detection method because many botnet detection methods are available in literature but there is no such generalized approach. The generic framework of botnet detection is lacking in the literature, which motivates the present study to design a generalized model for botnet detection. This work is indented for those researchers who want to implement a new model for the botnet detection that considers the general architecture.

This paper is organized as follows: Section II presents the background knowledge that describes forensics, network forensics, botnets and botnet detection methods and it also includes the proposed taxonomy of botnet detection methods. The literature review is discussed in section III. The proposal of the generic framework of botnet detection methods is presented in section IV. Future work is stated in section V.

II. BACKGROUND

A. Forensics

Forensics is the investigation technique that is used to gather evidences of some criminal activities. Forensic sciences have many branches and network forensics is one of them.

B. Network Forensics

Network forensics is a branch of forensics science and is the extension of network security. Network security simply detects and prevents the attacks but the network forensics has the capability to do investigation [10]. Network forensics is the investigation technology, which captures the network packets, record them for investigation and then analyze and correlate the recorded network data to find out the source of attacks [1].

C. Network attacks

With the increase usage of Internet, there is also a rapid increase in cyber crime, which includes various network attacks. Network attacks exploit the vulnerabilities of the

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

system and gain unauthorized access to the system [11]. One of the network attacks is botnet.

D. Botnets

Botnet is one of the common network attacks these days. Botnet is defined as a network or group of compromised computers called zombies, which are controlled by a botmaster automatically [3]. The botmaster controls the whole botnet using Command and Control servers [9].

E. Botnet detection methods

Botnet detection methods detect the botnet attacks. The botnet detection methods are broadly classified into two categories Honeynet based botnet detection and passive network traffic monitoring [12]. Honeynet is made of collection of more than one honeypot and a honeywall. A honeypot is a system designed to attract the attackers so as to observe their activities and find out solutions and honeywall is software used to do it [2]. Passive network traffic monitoring methods are further classified into IDS based detection, DNS based and data mining based detection techniques.

Figure 1 presents the proposed taxonomy based on the review of literature.

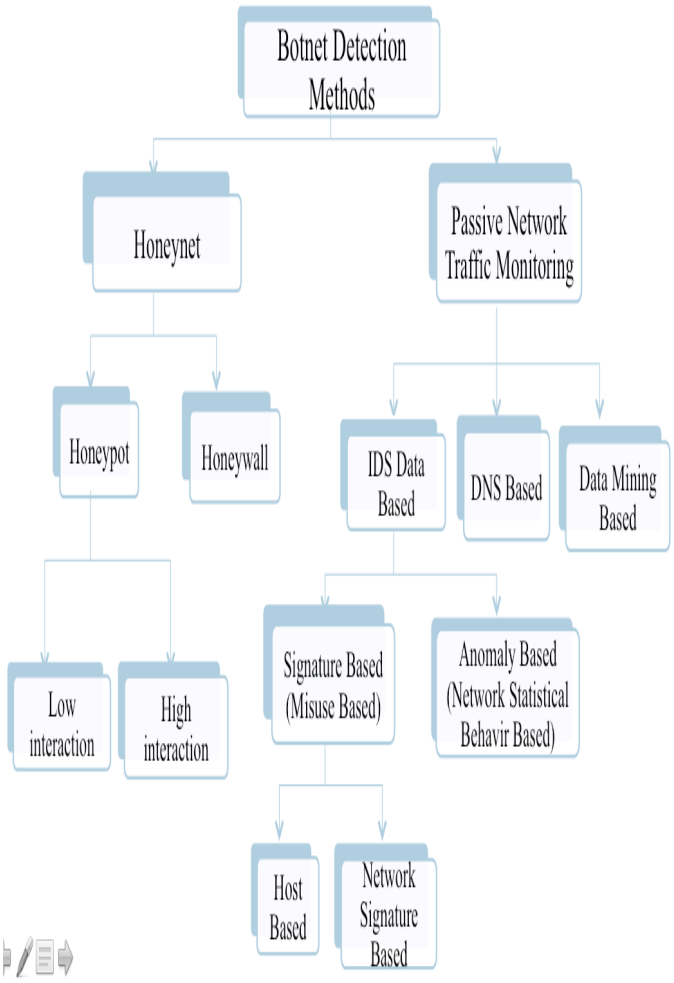


FIGURE 1. Proposed taxonomy of botnet detection methods

III. LITERATURE REVIEW

Cyber crime is a huge problem these days. In past few years many researchers have done research on Network forensics to reduce the cyber crime.

A. Network Forensics in literature

Ahmad Almulhem and Issa Traore [1] explored the topic of network forensics and proposed architecture of network forensics system. The proposed architecture manages to collect attack data at network and hosts. It is a capable of bypassing encryption if used by a hacker.

The challenges in deploying a network forensics infrastructure are highlighted by Ahmad Almulhem [10] in "Network Forensics: Notions and Challenges". The various aspects of network forensics and related technologies were presented with limitations of those technologies.

B. Botnet and Botnet Detection in literature

J.S.Bhatia, et al [2] presented the introduction to various Internet attacks. They discuss the botnet attacks and propose an approach to detect the botnet attacks that use the IRC and HTTP protocols. The proposed approach is based on Virtual Honeynet based system. They evaluated the approach using real world network traces.

Maryam Feily, et al [3] presented a survey on botnet and botnet detection. The presented survey clarifies what is botnet and also discusses the various botnet detection techniques. Their survey divides the botnet detection techniques into four categories: DNS-based, signature based, anomaly based and mining-based. It also compares the various botnet detection techniques.

Xiaonan Zang, et al. [4] conducted an experiment to observe the discriminating capabilities of the Hierarchical and K mean clustering algorithms and exploring a RTT adjustment procedure to mix the botnet trace with the background Internet traffic. Their experiment has shown the proposed capabilities.

Yousof Al-Hammadi and Uwe Aickelin [5] proposed a new technique to detect the presence of botnets. They used an interception technique to monitor Windows Application Programming Interface *(API)* functions calls made by communication applications and store these calls with their arguments in log files. They proposed an algorithm to detect botnets based on monitoring abnormal activity by correlating the changes in log file sizes from different hosts [5].

Systems detect botnets by examining traffic content for IRC commands or by setting up honeynets. W. Timothy Strayer, et al. [6] proposed an approach for detecting botnets by examining the flow characteristics such as duration, bandwidth, and packet timing that looks for evidence of botnet Command and Control activity. They constructed an architecture that first eliminates traffic that is unlikely to be a part of a network of bots; the remaining traffic is classified into a group that is likely to be part of a botnet, and then correlates the likely traffic to find common communications patterns that would suggest the activity of a botnet. The main focus of this method is on

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

reduction of data set by feeding the traffic packet traces into a series of quick reduction filters.

Hossein Rouhani Zeidanloo and Azizah Bt Abdul Manaf [7] provide taxonomy of botnets C&C channels and they also evaluate the well-known protocols that are being used. They also proposed a general detection framework that focuses on botnets based on P2P and IRC protocols. Their proposed botnet detection framework does not need any prior knowledge like signatures of the botnets.

Sandeep Yadav and A.L. Narasimha Reddy [8] explored the techniques that may utilize the failed domain queries. They present the DNS based botnet detection method.

Yousof Ali Abdulla Al-Hammadi [9] presented an approach that is host-based behavior for the detection of botnets. He monitor the function calls within a time window using various correlation algorithms. He uses an intelligent algorithm that is inspired from the immune systems.

The concepts of network attacks and network security along with cryptography are discussed in [11] by William Stallings.

Alexander V. Barsamian [12] proposed a framework to characterize the network behavior. He starts the research by collecting the network traffic from packet series and hypothesizes that they will characterize the behavior of traffic from threat data. He develops a method to measure the conformity and also detect behavioral changes and also evaluate it. He uses the Kullback-Leibler divergence method for this. He also describe various methods based on K-means approximation for detecting synchronous behavior .He analyze an application of their proposed methods and detect the hosts on the network for the presence of botnet infection.

Robert F. Erbacher, et al [13] introduced a multi-layered architecture to detect the various botnets. They use multiple techniques to detect the old as well as new botnet attacks that cannot be detected by a single technique. For the detection of well-known old botnet attacks, they use signature type techniques and for new botnets, data mining are used.

IV. PROPOSED FRAMEWORK FOR BOTNET DETECTION

Generic framework of the model for botnet detection is proposed in this section. The proposed framework is composed of some components as described below. The design of the proposed model is given in this paper. To design the proposed model some UML diagrams like class diagrams and activity diagram are used.

A. Common components of the generalized botnet detection methods

There are some common components that were used by the detection methods that are prevalent in the literature [2,6,7, 13]. This research extracts all the common components followed by the prevalent methods and use the extracted components for designing a generic framework of the model for botnet detection. The extracted common components that are used to design the generic framework of the model for botnet detection method are:

1. Filters
2. Classifiers
3. Correlator
4. Clusters
5. Analyzer

B. Design

The Model is designed using the UML diagrams. The class diagrams and activity diagram of UML tools are used in the present study. UML diagrams best represents the model and make it easier to understand the concept. UML helps to visualize the designs so that it can be checked against the requirements. There are many UML diagrams. In the present study various classes are used to build the Ontology of Botnet detection method so class diagrams best represent various classes and their subclasses. The flow between the processes of the classes cannot be shown using class diagrams only. To show the flow of data and interaction between the classes, the Activity diagram is used during design phase.

Eight classes are created to design a generic model of botnet detection. The classes are DataSource class that depicts the various sources of data to be analyzed, TrafficScanner class that represents the data capturing tool, PacketFilter class representing the filter components used in botnet detection methods of the literature, FlowClassificationEngine that depicts the classifier component of the prevalent botnet detection methods, PairwiseCorrelator class representing the correlator component of the detection methods studied, Clustering class represents the clusters component extracted from literature, TopologicalAnalyzer class that shows the Analyzer component of existing botnet detection methods and Result class that will show the details of the report generated at the end.

The description of all the classes used in the proposed design is explained here.

1) Class Diagrams

Class diagrams show the static structure of the system to be designed. They represent the entities that share the common characteristics.

Figure 2 shows the class diagram of the class DataSource. It shows the subclasses of the DataSource class. The subclasses of DataSource class are NetworkTrafficInformation, SystemProcessInformation and FileSystemInformation. NetworkTrafficInformation class has a subclass DNS.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

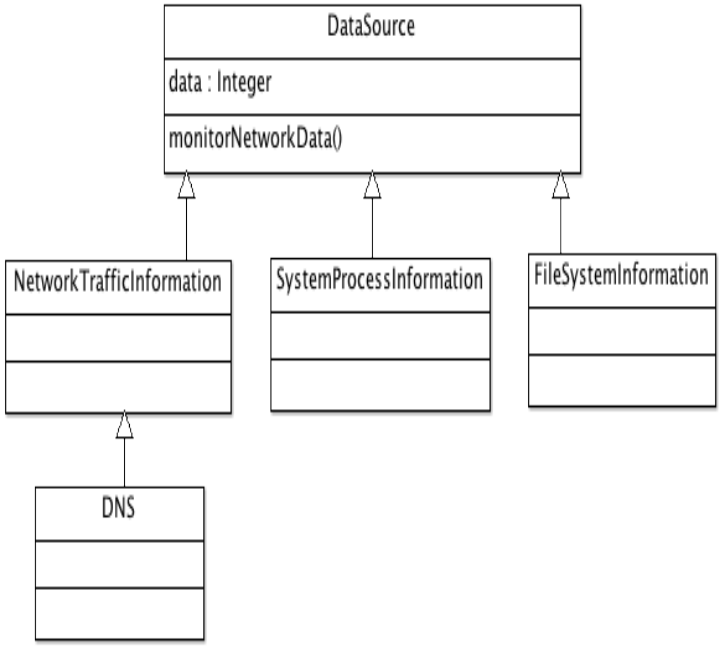


Figure 2. Class diagram of DataSource class

Figure 3 shows the TrafficScanner class. TrafficScanner class has an operation that captures and monitors the data. This class is composed of two classes Agents and Sensors. Agents class gather specific network information and create log files. Sensors class monitors the data in packets and also examine the data. Agents class have two subclasses ActiveAgents and PassiveAgents [14]. ActiveAgents further have a subclass Sniffers that is a data capturing tool. Sensors send the suspicious data to the MarkingModule class.

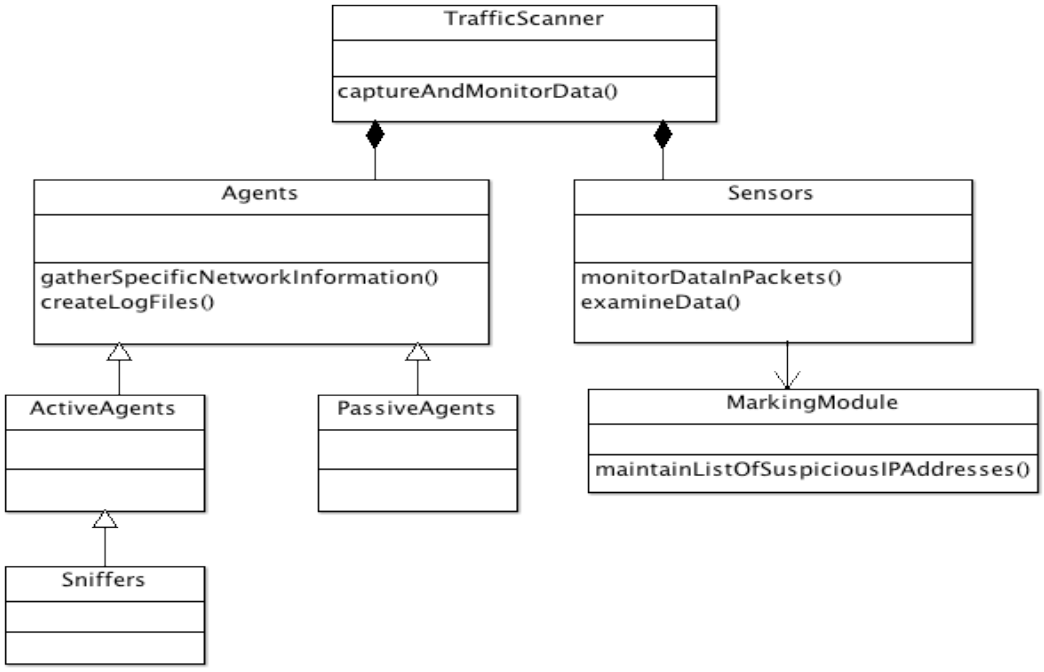


Figure 3. Class diagram of TrafficScanner class

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

Figure 4 shows the PacketFilter class. This class has attribute called flowAttribute. PacketFilter class has operations detectTrafficContent that detects the contents of the traffic; convertPacketTracesIntoFlowSummaries that convert the packets into flow summaries and eliminate C2 Flows. PacketFilter class is composed of classes QuickDataReduction and IncompleteCommunicationFilter. QuickDataReduction class selects the TCP based flow and the other class that is IncompleteCommunicationFilter class filters out the handshaking process that is SYN-RST exchanges.

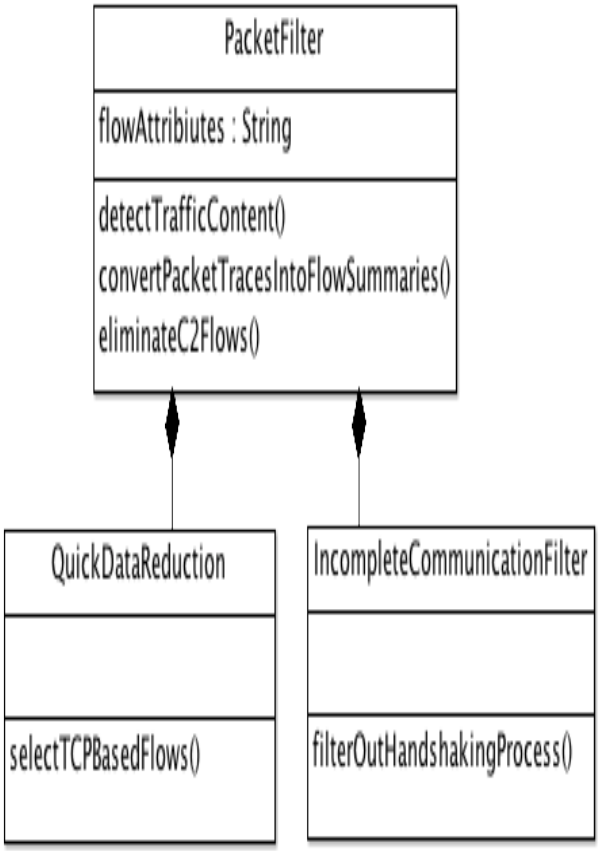


Figure 4. Class diagram of PacketFilter class

Figure 5 describes the FlowClassificationEngine class. It has an attribute payload and two operations classifyTrafficIntoGroups that classifies the incoming traffic into groups and Separate IRCandHTTPtraffic that separates the IRC traffic from the HTTP traffic. FlowClassificationEngine class is composed of FlowBasedDataReduction class and MachineLearningTechniques class. FlowBasedDataReduction class extracts the payload from the flow summaries. MachineLaerningTechniques class does content matching and it has two subclasses SignatureBasedClassifier and DecisionTreeBasedClassifier. SignatureBasedClassifier inspects the payload and DecisionTreeBasedClassifier detect the network anomalies. DecisionTreeBasedClassifier class is composed of Algorithms class and DataToAdjustAlgorithm class. Algorithm class depicts the algorithms that are used to implement the classifiers and DataToAdjustAlgorithm class represents the data required to adjust the algorithms that are used.

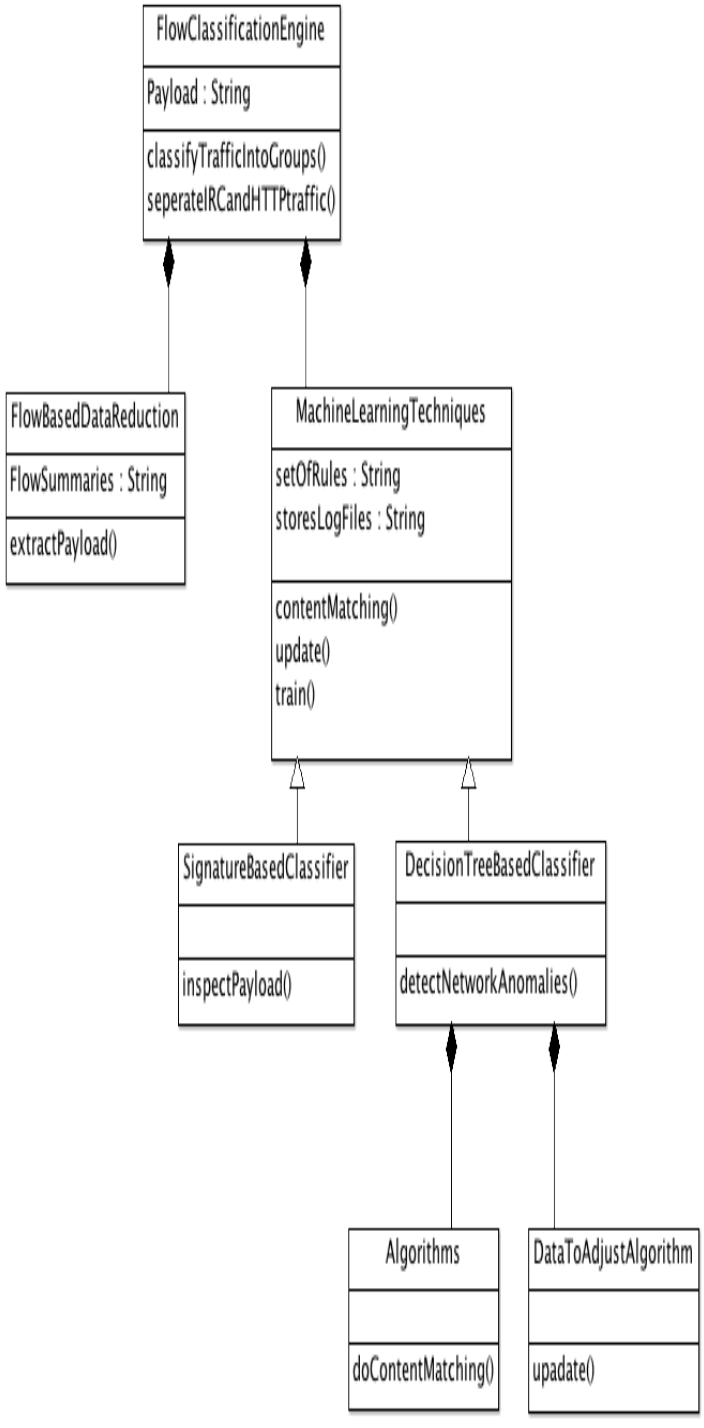


Figure 5. Class diagram of FlowClassificationEngine

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

Figure 6 shows the PairwiseCorrelator class. It has two attributes flowCharacteristics and payloadCommandSignatures. PairwiseCorrelator class does the pairwise examination of data so as to find out whether one flow is correlated to another flow and then finds the correlation value. It is composed of CorrelationAlgorithm class. CorrelationAlogrithm class implements the correlator.

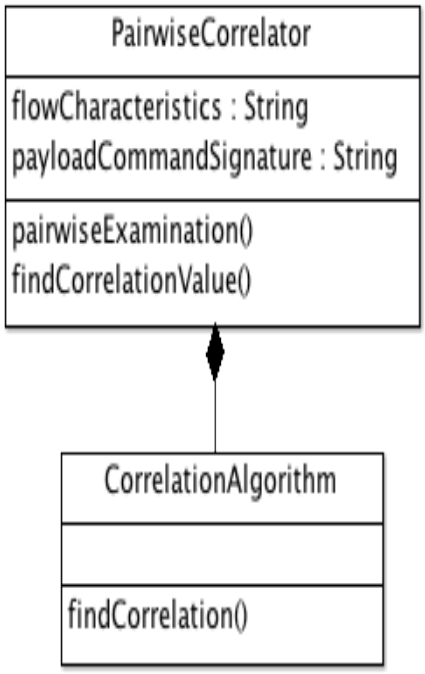


Figure 6. Class diagram of PairwiseCorrelator class

Figure 7 shows the Clustering class. Its attributes are timingCharacteristics and packetSizeCharacteristics. Clustering class group the flows that have similar flow characteristics.

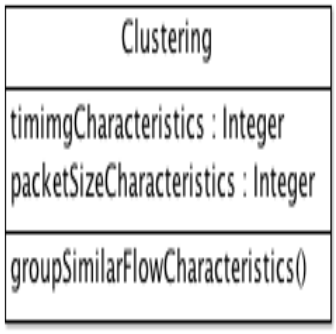


Figure 7. Class diagram of clustering class

Figure 8 shows the TopologicalAnalyzer class, which identifies the controller of the botnets. The controller of botnets is the botmaster. This class finds out the details of the botmaster and sends the details to the Result class.

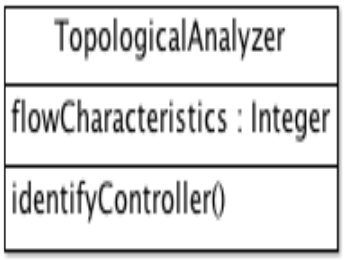


Figure 8. class diagram of TopologicalAnalyzer class

Figure 9 shows Result class. This class represents the result obtained after analyzing the traffic. It shows the details of the controller of the botnets. The details include the IP address of the bot controller along with the name of the bot.

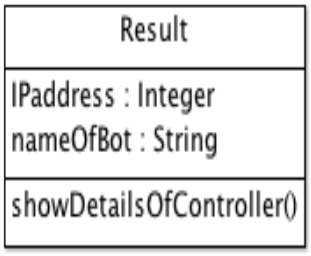


Figure 9. Class diagram of Result class

2) Activity Diagram

Activity diagram is also used in this research to design the general model for botnet detection methods. Activity diagrams are best to represent the flow of control between activities and show the system behavior. It is the graphical representation to show that the data moving in the system.

The proposed framework designed for the botnet detection method is demonstrated if Figure 10.

Figure 10 depicts the flow of control between processes of each class in the proposed framework of botnet detection method. Flow starts from the class DataSource that can be network traffic information, system process information and file system information. This class sends the data to the TrafficScanner class, which is a composition of Agents and Sensors. TrafficScanner gathers the specific Network information, create Log files, monitor the data and maintain a list of suspicious IP addresses. It forwards the Packet traces to next class. The next class is Packet Filter, which converts the packet traces to flow summaries, detect traffic content, select the TCP based flow and filters out the handshaking process. Then the remaining flows are sent to the FlowClassificationEngine. FlowClassificationEngine class extract and inspect the payload, does content matching and classify the flow into chat-like and non-chat like flows. The chat-like flows are forwarded to the PairwiseCorrelator that does the pairwise correlation and find the correlated values. The correlated flows are then sent to the Clustering class so that it can group the remaining network traffic with similar flow characteristics and store them in the database. The clusters from the database, in collection, are sent to the

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

TopologicalAnalyzer class that finds out the controller. Controller is the source of attack, which is botmaster. TopologicalAnalyzer class sends the details of the controller to the Result class. Result class presents the result by showing the details of the controller. The details include IP address and name of the botnet.

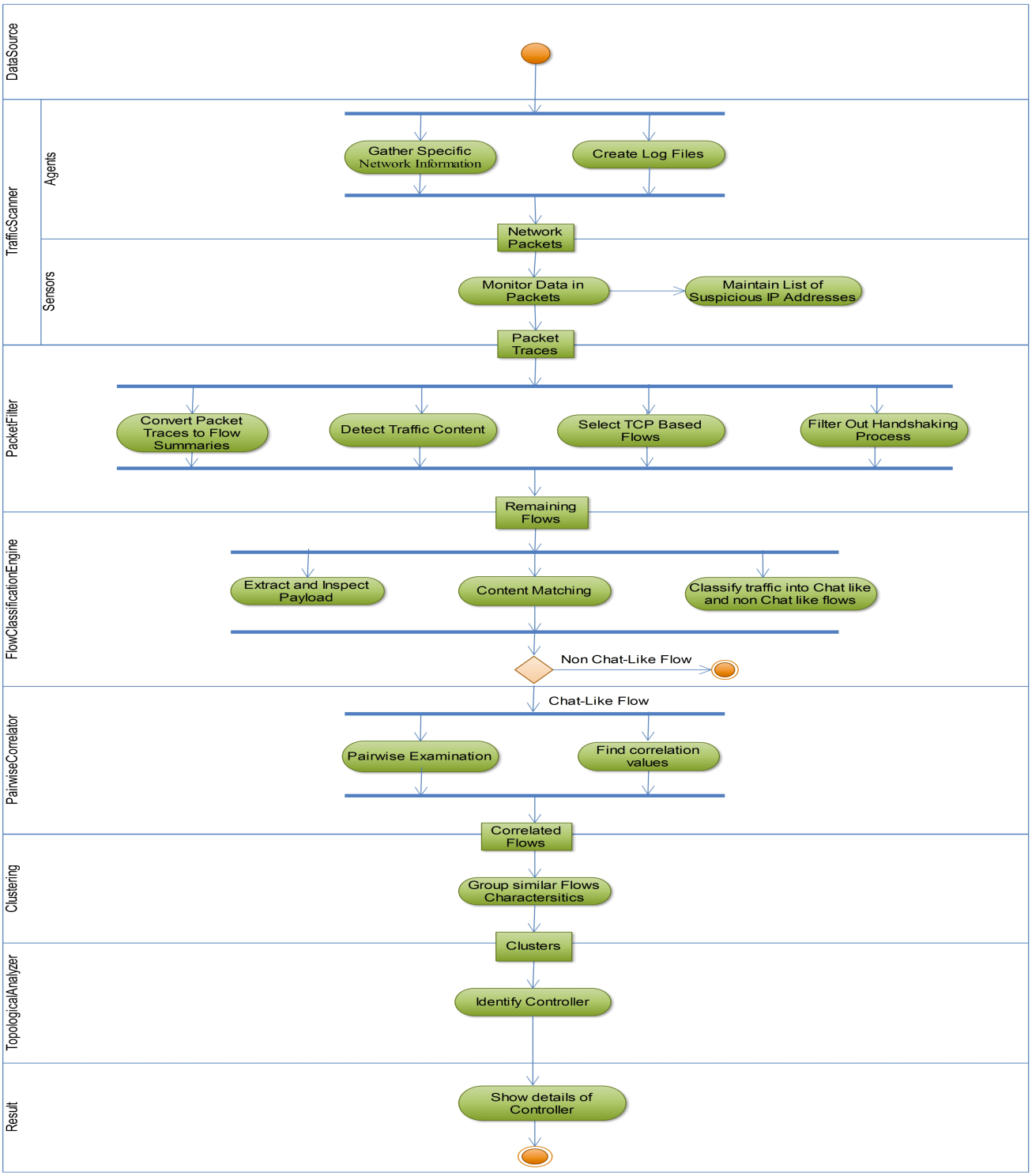


Figure 10. Design of proposed generic framework of botnet detection method

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

V. FUTURE WORK

The present study can be extended in future. The further research directions are the generalization with specialization that is to be added to address the specific concerns. A comprehensive version, that can be used to detect attacks, other or in addition to botnet detection, can be devised.

ACKNOWLEDGMENT

We are highly indebted to God for his blessings and love throughout my life and for not letting me down in difficult times. We are grateful to our family for their support.

REFERENCES

[1] Ahmad Almulhem and Issa Traore, "Experience with Engineering a Network Forensics System", in Proc. of international conference on Information Networking (ICOIN2005), Jeju, Korea, Vol. 3391, pp. 62-71, January 2005.

[2] J.S.Bhatia, R.K.Sehgal and Sanjeev Kumar, "Botnet Command Detection using Virtual Honeynet", in *International Journal of Network Security & Its Applications,* vol.3, no.5, pp. 117-189, Sep 2011.

[3] Maryam Feily, Alireza Shahrestani and Sureswaran Ramadass, "A Survey of Botnet and Botnet Detection", in *Proc. of 3rd International Conference of IEEE on Emerging Security Information, Systems and Technologies, SECURWARE '09,* Athens, Glyfada, pp. 268 - 273, June 2009.

[4] Xiaonan Zang , Athichart Tangpong, George Kesidis and David J. Miller, "Botnet Detection Through Fine Flow Classification", unpublished, Departments of CS&E and EE, The Pennsylvania State University, University Park, PA, *Report No. CSE11-001,* Jan. 31, 2011.

[5] Yousof Al-Hammadi and Uwe Aickelin, "Detecting Botnets Through Log Correlation", in *Proc. of the Workshop on Monitoring, Attack Detection and Mitigation (MonAM2006), Tubingen, Germany,* pp. 97-100, 2006.

[6] W. Timothy Strayer, Robert Walsh, Carl Livadas and David Lapsley, "Detecting Botnets with Tight Command and Control", in *Proc. of the 31st IEEE Conference on Local Computer Networks (LCN),* pp. 195-202, 2006.

[7] Hossein Rouhani Zeidanloo and Azizah Bt Abdul Manaf, "Botnet Detection by Monitoring Similar Communication Patterns", in *International Journal of Computer Science and Information Security,* vol. 7, no. 3, pp. 36-45, 2010.

[8] Sandeep Yadav and A.L. Narasimha Reddy, "Winning with DNS Failures: Strategies for Faster Botnet Detection", *in Proc. of 7th International ICST Conference, SecureComm,* London, UK, pp. 446-459, 2011.

[9] Yousof Ali Abdulla Al-Hammadi, "Behavioural Correlation for Malicious Bot Detection", Ph.D. dissertation, The University of Nottingham, April 2010.

[10] Ahmad Almulhem, "Network Forensics: Notions and Challenge", *IEEE International Symposium on Signal Processing and Information Technology,* pp. 463 - 466, 2009.

[11] William Stallings, " Cryptography and Network Security: Principles and Practices", 3rd ed. Pearson Education, 2002 ISBN:0130914290

[12] Alexander V. Barsamian, "Network Characterization For Botnet Detection Using Statistical-Behavioral Methods", M.S. thesis, Thayer School of Engineering Dartmouth College, Hanover, New Hampshire, June 2009.

[13] Robert F. Erbacher, Adele Cutler, Pranab Banerjee and Jim Marshall, "A Multi-Layered Approach to Botnet Detection", *in Proc. of the International Conference on* Security *and Management,* Las Vegas, Nevada, USA, pp. 301- 308, July 2008.

[14] Alisha Cecil, "A Summary of Network Traffic Monitoring and Analysis" [online]. Available: http://www.cse.wustl.edu/~jain/cse567-06/ftp/net_monitoring.pdf accessed on August 30, 2013.

Visualization for Information Retrieval in Regional Distributed Environment

Mamoon H. Mamoon, Hazem M. El-Bakry, Amany A. Slamaa
Faculty of computer science & information system
Mansoura University, EGYPT

Abstract— **Information retrieval (IR) is the task of representing, storing, organizing, and offering access to information items. The problem for search engines is not only to find topic relevant results, but results consistent with the user's information need. How to retrieve desired information from the Internet with high efficiency and good effectiveness is become the main concern of internet user-based. The interface of the systems does not help them to perceive the precision of these results. Speed, resources consuming, searching and retrieving process also aren't optimal. The search engine's aim is developing and improving the performance of information retrieval system and gifting the user whatever his culture' level. The proposed system is using information visualization for interface problems, and for improving other side of web IR system's problems, it uses the regional crawler on distributed search environment with conceptual query processing and enhanced vector space information retrieval model (VSM). It is an effective attempt to match renewal user's needs and get a better performance than ordinary system**.

Keywords- Regional distributed crawler, VSM, conceptual weighting, visualization, WordNet, information visualization, web information retrieval.

I. INTRODUCTION

This paper tries to aggregate an optimal or at least semi-optimal information retrieval system by present visualized results supported by more efficient search engine than the standard. A search engine operates in the following order: Web crawling, Indexing, and Searching. The development include them as distributed regional web crawler, conceptual searching. The refinement on proposed system not stopped only at searching and results but also accommodate to involve personalization benefits.

The goal of an information retrieval system is to maximize the number of relevant documents returned for each query. Keyword information retrieval systems often return a proportion of irrelevant documents because matching keywords is imprecise: words can have different meanings when used in different contexts, and a single idea can often be expressed by several different words or synonyms. Information retrieval systems can be made more precise by matching concepts, keywords for which the intended meaning has been identified, either with information from a lexicographic database in the case of documents, or by asking the user to choose one meaning from several possible meanings in the case of queries.

The matching algorithms used by keyword IR systems are imprecise and retrieve irrelevant as well as relevant results. Two causes of this imprecision are terminology and semantics, both aspects of natural language. Terminology affects retrieval because different people use different words for the same concept. Terminology is often cultural; a pavement in the UK is a sidewalk in the US, for example.

Semantics affects retrieval because the text of a document may not contain the exact keywords in the query but may nevertheless be about the topic of interest. This problem is exacerbated by polysemy. Polysemous words have different meanings in different contexts. For example, Java can refer to the Island in Indonesia, a type of coffee, coffee itself, or the object-oriented programming language. Matching a word does not identify the context in which it is used. The polysemous meanings, or senses, of words can lead to keyword queries that are ambiguous. Identifying the intended meaning of keywords can improve the precision of IR systems.

The concept IR model proceeds in three stages: the concepts in each document in the collection are identified, the concepts in a query are identified, and the query concepts are matched with the document concepts. Using the concept declared in details in sections 1, 2. And see how the concept also enhances one of information retrieval model – vector

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

space Model - not only intended meaning problem in section 2.

As declared before and according to web information retrieval problems [18], there is attempt to solve some of them. In proposed system, there are some visualization forms, user select which one he prefer. The suitable form will display as default according to information that conclude from his profile and his set cabbalists – software and hardware -. This option is different on other retrieval – visualization system that it display only one form don't care that it easy and suitable all culture (inchoate or experienced) users. The system use personalization not only for customize results but also in improving searching process and increase time response. Figure 1 shows main components of the proposed system (VIZIRRD).

The Visualizing Information Retrieval system (VIZIR) or visual information retrieval for the web has two main engines; search engine and visualization engine. Each one of them has own input and output and component that declared in next figure 2. This system also has a personalized feature. Combining these three will increase: performance, efficiency and each user get own system which declare how that achieve in the following sections.

II. Prepare Your Paper Before Styling

Whenever a user wants to retrieve a set of documents, he starts to construct a concept about the topic of interest; such a conceptualization is called the "information need". Given an "information need", the user must formulate a query that is adequate for the information retrieval system. Usually, the query is a collection of index terms, which might be erroneous and improper initially. In this case, a reformulation of the query should be done to obtain the desired result. The reformulation process is called query expansion [4]. So, Query expansion (QE) is the process of reformulating a seed query to improve retrieval performance in information retrieval operations by expanding search query to match additional documents [9 and 41]. In the context of web search engines, query expansion involves evaluating a user's input (what words were typed into the search query area and sometimes other types of data) and expanding the search query to match additional documents. Query expansion involves techniques such as:

1. Finding synonyms of words, and searching for the synonyms as well.
2. Finding all the various morphological forms of words by stemming each word in the search query.
3. Fixing spelling errors and automatically searching for the corrected form or suggesting it in the results.
4. Re-weighting the terms in the original query [9].

2.1- Query expansion and WordNet:

Keyword IR systems retrieve documents by matching the keywords in the query with the keywords in the documents. A simple data structure that maps keywords to documents is an inverted index. Each keyword (K_i) is listed in an index and points to a list of the documents (Di) that contain the keyword:

- $K_1 \Rightarrow D_1, D_2, D_3, D_4$
- $K_2 \Rightarrow D_1, D_2$
- $K_3 \Rightarrow D_1, D_2, D_3$
- $K_4 \Rightarrow D_1$

Query expansion is one automated technique that has been used to address the imprecision of text retrieval techniques (Spink 1994). Query expansion adds keywords to a query that are related to the keywords supplied by the user, such as the synonyms of the keyword. For example, if the original query contains a single keyword, K, the synonyms of the keyword are added as disjunctions. The query, Q = K is expanded to incorporate the synonyms S_i of K:

$$Q_E = K \vee S_1 \vee S_2$$

Adding the synonyms helps to overcome the problem of different words being used for the same concept also when user's query is only one, two, or maximum three words. Automatic methods of choosing synonyms are required because users find it difficult to come up with alternative search terms.

Query expansion techniques can be enhanced with concepts to make the expanded queries more specific. Once the keywords in a query have been disambiguated into concepts, the keywords relating to the generalizations, specializations, and the parts of the concepts can be added to the query [7]. The importance of concept will be duplicated if we use it

2.2 Query expansion role in precision and recall

Search engines invoke query expansion to increase the quality of user search results. It is assumed that users do not always formulate search queries using the best terms. Best in this case

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

may be because the database does not contain the user entered terms.

By stemming a user-entered term, more documents are matched, as the alternate word forms for a user entered term are matched as well, increasing the total recall. This comes at the expense of reducing the precision. By expanding a search query to search for the synonyms of a user entered term, the recall is also increased at the expense of precision. This is due to the nature of the equation of how precision is calculated, in that a larger recall implicitly causes a decrease in precision, given that factors of recall are part of the denominator. It is also inferred that a larger recall negatively impacts overall search result quality, given that many users do not want more results to comb through, regardless of the precision.

The goal of query expansion in this regard is by increasing recall, precision can potentially increase (rather than decrease as mathematically equated), by including in the result set pages which are more relevant (of higher quality), or at least equally relevant. Pages which would not be included in the result set, which have the potential to be more relevant to the user's desired query, are included, and without query expansion would not have, regardless of relevance. By ranking the occurrences of both the user entered words and synonyms and alternate morphological forms, documents with a higher density (high frequency and close proximity) tend to migrate higher up in the search results, leading to a higher quality of the search results near the top of the results, despite the larger recall [9].

III. Proposed Information Retrieval Model to use

A fundamental weakness of current information retrieval method is that the vocabulary that searchers use in formulating their queries is often not the same as the one by which the information has been indexed. In an attempt to resolve this drawback has been to combine Vector Space Model (VSM) and WordNet [4, 5 and 10] ontology after replacing the Term frequency- Inverse Document Frequency TF-IDF term weighting with Concept-based Term Weighting (CBW) to Compatible with WordNet. WordNet is utilized to get conceptual information of each word in the given query context.

3.1- a new Query term weighting for Vector Space Model

3.1.1 Problem definition and suggestion

The main disadvantage of the vector space model is that it does not in any way define what the values of the vector components should be. The problem of assigning appropriate values to the vector components is known as term weighting. Early experiments by Salton (1971) and Salton and Yang (1973) showed that term weighting is not a trivial problem at all. They suggested so-called tf :idf weights, a combination of term frequency tf , which is the number of occurrences of a term in a document, and idf , the inverse document frequency, which is a value inversely related to the document frequency df , which is the number of documents that contain the term. Many modern weighting algorithms are versions of the family of tf :idf weighting algorithms. Salton's original tf :idf weights perform relatively poorly, in some cases worse than simple idf weighting [5].

As calculating query term importance was a fundamental issue of the retrieval process. The traditional term weighting *scheme* TF-IDF approach has following drawbacks:

1- Rare terms are no less important than frequent terms – IDF assumption
2- Multiple appearances of a term in a document are no less important than single appearance – TF assumption

Because of IDF assumption, the TF-IDF term weighting scheme assigns higher weights to the rare terms frequently. Thus, it will influence the performance of classification. *Concept-based Term Weighting* (CBW) calculates term importance by utilizing conceptual information found in the WordNet ontology. CBW was fundamentally different than IDF in that it was independent of document collection. The significance of CBW over IDF is that:

1- CBW introduced an additional source of term weighting using the WordNet ontology.
2- CBW was independent of document collection statistics, which is a feature that affects performance [5].

3.2.2 Vector Space Model (VSM) using WorldNet

Term significance can be effectively captured using CBW and then be used as a substitute or possible co-contributor to IDF. CBW presents a new way of interpreting ontologies for retrieval, and introduces an additional source of term importance information that can be used for term weighting. In

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

proposed method, Concept-based Term Weighting (CBW) technique is used to calculate term importance by finding the conceptual information of each term using WordNet ontology. The significance of this technique is that:

1. it is independent of document collection statistics,
2. it presents a new way of interpreting ontologies for retrieval,
3. It introduces an additional source of term importance information that can be used for term weighting.

In this research project WordNet is the chosen ontology used by CBW. To determine generality or specificity for a term, conceptual weighting employs four types of conceptual information in WordNet:

1. Number of Senses.
2. Number of Synonyms.
3. Level Number (Hypernyms).
4. Number of Children (Hyponyms/Troponyms).

Overview of Concept based term weighting to calculate CBW value of a query term is shown in figure 3. As shown, there are three main steps involved to find the weight of a query. Extraction step extracts conceptual information of each word based on each POS (Noun, Verb, Adjectives) from WordNet. Weighting step find the weight of each extracted integer values for each POS based on weighting functions. After weighting fusion is applied to get a single CBW value for a query term. Any terms used in the query that are non-WordNet terms were given a default high CBW value. This is based on the assumption that the term does not appear in WordNet, is most likely a specific term, and thus it is highly weighted.

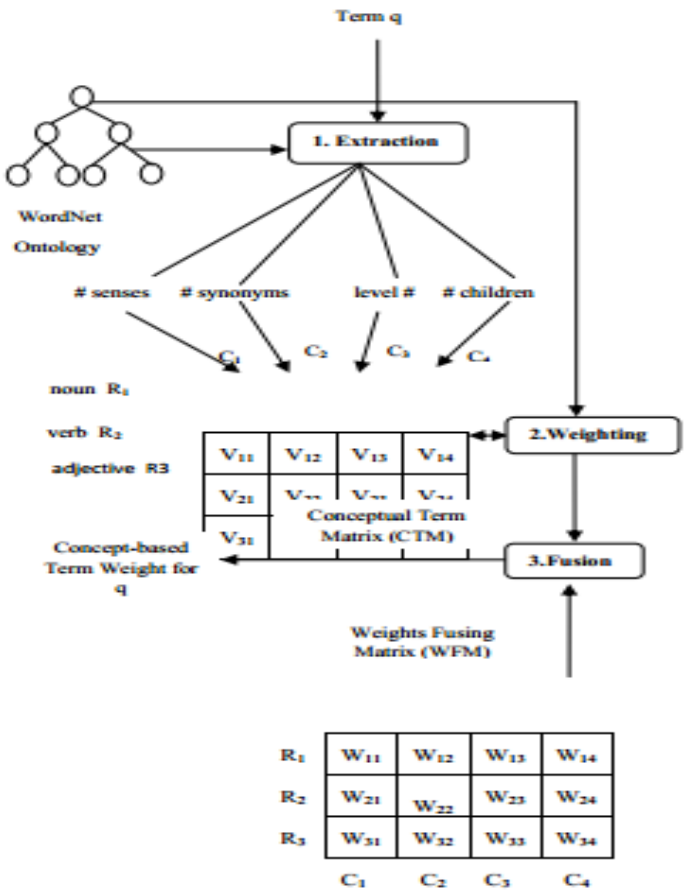


Figure 3: overview of concept-based term weighting (CBW)

The block diagram shown in figure below consists of three main steps:

1. Extraction
2. Weighting
3. Fusion

Extraction:

This step works on a query given by user and extracts the conceptual value for each input query term from WordNet which includes number of senses, number of synonyms, level number (Hypernyms) and number of Children (Hyponyms/Troponyms). Extraction is done by using extraction algorithm [2] as shown below. Initially all values in conceptual term matrix (CTM) are set to -1. Then senses for each POS are counted from WordNet and listed in the first

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

column of CTM. Similarly synonyms for each POS are found by selecting maximum synonyms for senses given by WordNet for a query term. Levels for each POS are found by selecting minimum hypernyms for senses given by WordNet for a input query term and listed in third column of CTM. And finally children for each POS are found by selecting maximum hyponyms/troponyms for senses given by WordNet for a query term. These extracted integer values are stored in Conceptual Term Matrix (CTM) [5].

1. Initialize CTM to (-1).
2. For each row R_m in CTM:

2.1 Get set of synsets S in R_m section (POS) of

WordNet in which q belongs to: S = WordNet (q, POS).

2.2 Extract conceptual information from S:

a. V_{m1} = COUNT(S)

b. V_{m2}= MAX($S_{synonyms}$)

c. V_{m3}= MIN(S_{level})

d. V_{m4}= MAX($S_{children}$)

Note: POS is Part of Speech.

Extraction Algorithm

Stemming the terms before building the inverted index has the advantage that it reduces the size of the index, and allows for retrieval of web pages with various inflected forms of a word (for example, when searching for web pages with the word computation, the results will include web pages with computations and computing). Stemming is easier to do than computing base forms, because stemmers remove suffixes, without needing a full dictionary of words in a language [5].

Weighting

Weighting is the next step after extraction. Weighting functions convert extracted integer values into weighted values in the range [0, 1]. These weighted values are stored in weighted conceptual term matrix. Based on min, max and avg values for each POS (noun, verb and adjectives) weighting functions are designed as shown in equation (1) and (2). The level number and the number of children are both set to zero for adjectives because adjectives are not organized in a conceptual hierarchy since they are only descriptors of nouns. Therefore, it is not possible to extract the level number and the number of children from WordNet for adjectives. Therefore weighting functions are not created for level number and number of children of adjectives.

a) General Weighting Function for

i. Nouns, Verbs Senses, Synonyms and Children

ii. Adjectives Senses and Synonyms

$$f(x) = \begin{cases} 0 & , x \geq Max \\ 0.5 & , x = Avg \\ 1 & , x = Min \\ f(x-\Delta x) - \frac{0.5 \times \Delta x}{Avg - Min}, & Min < x < Avg \\ f(x-\Delta x) - \frac{0.5 \times \Delta x}{Avg - Min}, & Max > x > Avg \end{cases}$$

Eq. (1)

b) General Weighting Function for Nouns, Verbs Levels

$$f(x) = \begin{cases} 0 & , x = Min \\ 0.5 & , x = Avg \\ 1 & , x \geq Max \\ f(x-\Delta x) + \frac{0.5 \times \Delta x}{Avg - Min}, & Min < x < Avg \\ f(x-\Delta x) + \frac{0.5 \times \Delta x}{Avg - Min}, & Max > x > Avg \end{cases}$$

Eq. (2)

In above functions is taken as an error factor. These all functions are based on Min, Max and Avg values of each POS. For noun, verb and adjective's senses, weight 0 is assigned for an integer value greater than or equal to Max, weight 0.5 is

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

assigned for an integer value equal to Avg and weight 1 is assigned for an integer value equal to Min. For an integer value in the range [Min, Avg] is given a weight in the range [0.5, 1] and an integer value in the range [Max, Avg] is given a weight in the range [0, 0.5]. Same rules are applied for noun, verb and adjective's synonyms and children. For noun, verb and adjective's level, weight 0 is assigned for an integer value equal to Min, weight 0.5 is assigned for an integer value equal to Avg and weight 1 is assigned for an integer value greater than or equal to Max. For an integer value in the range [Min, Avg] is given a weight in the range [0, 0.5] and an integer value in the range [Avg, Max] is given a weight in therange [0.5, 1].

Fusion

Fusion is the last step to get single CBW value of a query that determines the importance of a term. Fusion is performed on weighted conceptual term matrix which is the result obtained by weighting. Fusion considers a new matrix named as Weights Fusing Matrix (WFM) of size 3*4 with all values set to 0.5 to give an average effect. WFM is shown below.

R_1	0.5	0.5	0.5	0.5
R_2	0.5	0.5	0.5	0.5
R_3	0.5	0.5	0.5	0.5
	C_1	C_2	C_3	C_4

Fusing steps:

1. Fuse each column of the weighting CTM with the columns of WFM using column weighted average function.

$$C_n = \frac{\sum_m Vm \times W_{mn}}{\sum W_m}, n = 1, 2, 3, 4$$

Eq. (3)

2. Fuse the row R generated in step (1), as shown in previous using row weighted average to give the CBW term importance.

$$CBW_q = \frac{\sum_n C_n \times W_n}{\sum_n W_n}$$

Eq. (4)

Where W is a set of weights with each element being a value in the range [0, 1], and set to 0.5 by default.

Note: weighting (CBW) = Weighted Conceptual Term Matri Weighted (CTM) X Weight Fusing Matrix (WFM)

IV. IMPLEMENTATION OF THE CONCEPT IN THREE PHASES

The first stage in the concept IR model is to identify the concepts in the documents in the collection. An index must be built that maps concepts to documents to enable fast retrieval. This process need only be done once for each document added to the collection. The index can be updated incrementally as each new document is added to the collection.

The concepts in a document are identified by first extracting the keywords and removing the duplicates and stop words. Each keyword is then added to a list of concepts. A keyword with more than one sense must be disambiguated before being added to the list of concepts. Five tests are performed to identify which sense of a keyword is present in a document. A point is awarded if any of the following conditions are met:

1. one or more of the synonyms of the sense occur in the document;
2. the sense is a part of a concept that occurs in the document;
3. a concept that occurs in the document is part of the sense;
4. the sense is a generalization of a concept that occurs in the document;
5. The sense is a specialization of a concept that occurs in the document.

The application of each test produces a matching score that indicates the algorithm's level of confidence that the concept is present in the document. Tied scores can be presented to a domain expert for final classification. Each concept (C_i) is stored in an index and points to a list of the documents that contain the concept. For example:

- $C_1 \Rightarrow D_1, D_2, D_3, D_4$

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

- $C_2 \Rightarrow D_1, D_2$
- $C_3 \Rightarrow D_1, D_2, D_3$
- $C_4 \Rightarrow D_1$

The list of documents for each concept in the index is augmented with the score M_i of matching concept C with document D_i:

$$C \Rightarrow \{D_1, M_1\}, \{D_2, M_2\}, \{D_3, M_3\}$$

The same Boolean operations can be applied to an index of concepts as for an index of keywords.

For example, the simple query Q = java, must be disambiguated by asking the user which of the three senses of Java is intended. The keywords in a query can be disambiguated by presenting a list of the senses of each keyword and enabling the user to select the intended senses.

Disambiguating a keyword by selecting one sense over the other senses indicates that documents containing the other senses should not be retrieved. If query Q is disambiguated into sense 3, the object-oriented programming language, then documents about the island or the beverage should not be retrieved. This requirement can be met by ensuring that documents containing the synonyms, specializations, generalizations, etc. of the other senses are not retrieved. This translates into a query such as:

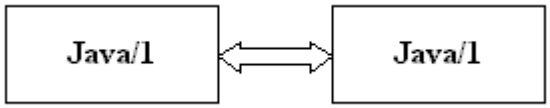
$$Q_E = \text{java} \wedge \neg(\text{jakarta} \vee \text{indonesia} \vee \text{bali}) \wedge \neg(\text{espresso} \vee \text{caffeine} \vee \text{tea})$$

This query requests documents that contain the keyword Java but not the keywords that relate to the two senses of Java that are not required: the island and the beverage. The keywords that represent the island are one of its parts, Jakarta, the whole of which Java is a part, Indonesia, and another part of Indonesia, Bali. These are a selection of the keywords that might be present in a document about Java the island. The keywords that represent the beverage are a type of coffee, espresso, a substance that is part of coffee, caffeine, and an alternative to coffee, tea. These are a selection of the keywords that might be present in a document about Java the beverage.

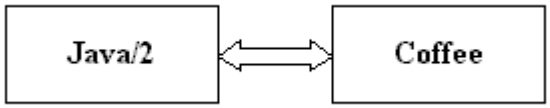
The final stage of the concept IR model is to match the concepts in a query with the concepts in the documents. Documents are matched with queries using the concept ⇒ document index. The degree to which the concepts in a query match the concepts in the documents is represented by a numerical matching score that is used to rank the results.

The concept IR model is more flexible than the strict matching performed by the Boolean keyword model. The Boolean model partitions documents into two sets: those documents that contain the query keywords and those do not. This strict partitioning does not fit well with natural language. The concept IR model enables documents to be retrieved that match queries in varying degrees.

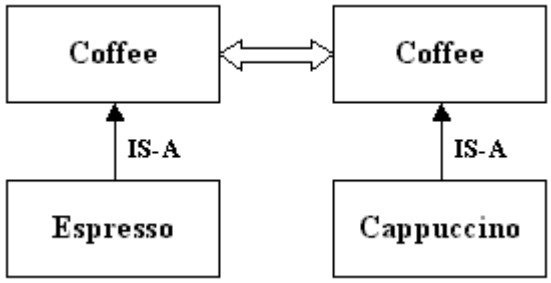
Five matching rules—based on the relations described in section 3—are used to generate a matching score. The base rule is that the same sense of a keyword always matches itself.



Synonyms of the same sense of a keyword always match.

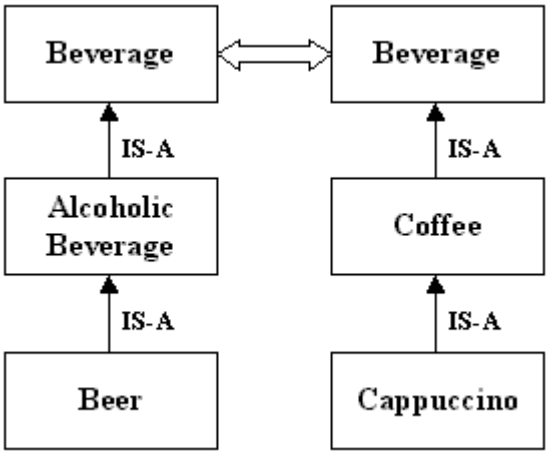


Concepts can be matched by the hyponym (generalization) relation. For example, espresso and cappuccino are both types of coffee, i.e. coffee is a generalization of both espresso and cappuccino. Concepts can be matched by applying a relation more than once.

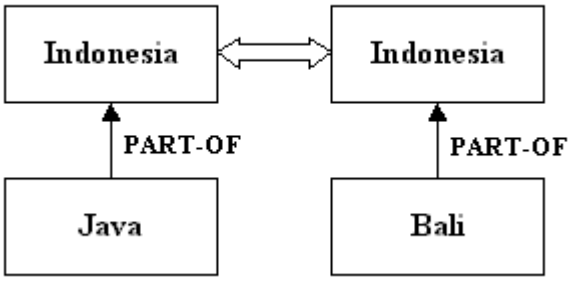


http://sites.google.com/site/ijcsis/
ISSN 1947-5500

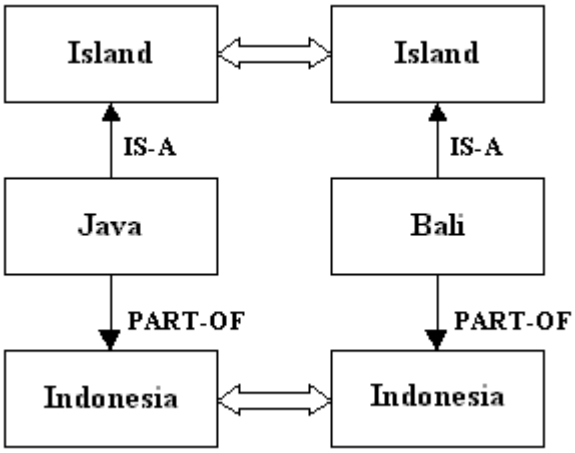
Similarly, beer is a type of alcoholic beverage and that cappuccino is a type of coffee. Alcoholic beverage and coffee are both types of beverage: beverage is a generalization of a generalization of beer and coffee.



Concepts can be matched by the meronym (part-of) relation. For example, Java and Bali are parts of Indonesia. Concepts can also be matched by more than one relation at once.



Java and Bali match Island with the hyponym relation, and match Indonesia with the meronym relation.

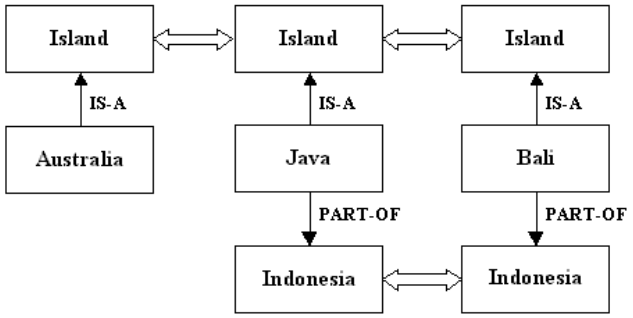


Two concepts, A and B, are matched in the following order:

1. check if A = B;
2. check if A is a synonym of B;
3. check if A and B are part of the same concept;
4. check if A and B have a common generalization;
5. check if A is a generalization of B.

The relation that matches two concepts determines the matching score. For example, the meronym relation is stronger than the hyponym relation. Concepts that are part of a whole are more closely related that generalizations of those parts. Java is more closely related to Bali than to Australia, for example, because Java and Bali are part of Indonesia, even though they are all islands.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500



Matching scores are weighted by the relation used to match the concepts. The relations have different weights and are weighted in the following order, from highest to lowest:

1. exact match;
2. synonyms;
3. parts;
4. specializations;
5. Generalizations.

Matching scores are also weighted by the number of relations used to match the concepts; the larger the number of relations used, the lower the score. For example, espresso and cappuccino would have a higher matching score than beer and cappuccino even though the three concepts have a common generalization, beverage. Espresso and cappuccino are matched with one application of the hyponym relation; beer and cappuccino are matched with two applications [7].

To overcome query formulation and the inherent word ambiguity in natural language problems, researchers have focused on automatic query expansion to help the user formulate what information is really needed, declared before. Another widely used method of query expansion is the use of relevance feedback from the user which gives the relevance of documents to clarify the ambiguity. In fact, these two techniques complement each other. However, the mechanisms of relevance feedback based on words or documents in the past research both have their own deficiencies [8]. This involves the user performing a preliminary search, then examining the documents returned and deciding which are relevant. Finally, terms from these documents are added to the query and the search is repeated. This obviously requires human intervention and, as a result, is inappropriate in many situations. However, there is a similar approach, sometimes called pseudo-relevance feedback, in which the top few documents from an initial query are assumed relevant and are used for automatic feedback [4].

Spink et al. (2000) present results from the use of relevance feedback in the Excite search engine. Only about 4% of user query sessions used the relevance feedback option, and these were usually exploiting the ``More like this" link next to each result. About 70% of users only looked at the first page of results and did not pursue things any further. For people who used relevance feedback, results were improved about two thirds of the time [11].

5. User interface and information visualization

User interfaces are a communication between human information seekers and information retrieval systems. Information seeking is an imprecise process. When users approach an information access system they often have only a fuzzy understanding of how they can achieve their goals. Thus the user interface should aid in the understanding and expression of information needs. It should also help users formulate their queries, select among available information sources, understand search results, and keep track of the progress of their search [1].

The roles of user interface are:

1- Aiding in the understanding and expression of information needs.
2- helping users formulate their queries, select among available information sources, understand search results, and keep track of the progress of their search.(formulate/ select/ understand/ keep track)

What makes an effective human- computer interface?

"Well designed, effective computer systems generate positive feelings of success, competence, mastery,

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

and clarity in the user community. When an interactive system is well-designed, the interface almost disappears, enabling users to concentrate on their work, exploration, or pleasure [1] ."…….Ben shneiderman

Below we discuss those principles that are of special interest to information access systems.

1- Design principles
   a. *Offer informative feedback*: users with feedback about the relationship between their query specification and documents retrieved, about relationships among retrieved documents, and about relationships between retrieved documents and metadata describing collections. If the user has control of how and when feedback is provided, then the system provides an *internal locus of control.*
   b. *Reduce working memory load*: information access is an iterative process, the goals of which shift and change as information is encountered. one key way information access interfaces can help with memory load is to provide mechanisms for keeping track of choices made during the search process, allowing users to return to temporarily abandoned strategies, jump from one strategy to the next, and retain information and context across search sessions. Another memory-aiding device is to provide browsable information that is relevant to the current stage of the information access process. This includes suggestions of related terms or metadata, and search starting points including lists of sources and topic lists.
   c. *Provide alternative interfaces for novice and expert users*: an important tradeoff in all user interface design is that of simplicity versus power. Simple interfaces are easier to learn, at the expense of less flexibility and sometimes less efficient use. Powerful interfaces allow a knowledgeable user to do more and have more control over the operation of the interface, but can be time-consuming to learn and impose a memory burden on people who use the system only intermittently. A common solution is to use a "scaffolding" technique. The novice user is presented with a simple interface that can be learned quickly and that provides the basic functionality of the application, but is restricted in power and flexibility. Alternative interfaces are offered for more experienced users, giving them more control, more options, and more features, or potentially even entirely different interaction models. Good user interface design provides intuitive bridges between the simple and the advanced interfaces [1].

From the viewpoint of user interface design, people have widely differing abilities, preferences, and predilections. Important differences for information access interfaces include relative spatial ability and memory, reasoning abilities, verbal aptitude, and (potentially) personality differences. Age and cultural differences can contribute to acceptance or rejection of interface techniques. An interface innovation can be useful and pleasing for some users, and foreign and cumbersome for others. Thus software design should allow for flexibility in interaction style, and new features should not be expected to be equally helpful for all users [1].

An important aspect of human-computer interaction is the methodology for evaluation of user interface techniques. Users require only a few relevant documents and do not care about high recall to evaluate highly interactive information access systems, useful metrics beyond precision and recall include: time required to learn the system, time required to achieve goals on benchmark tasks, error rates, and retention of the use of the interface over time [1].

Visualization

The human perceptual system is highly attuned to images, and visual representations can communicate some kinds of information more rapidly and effectively than text. For example, the familiar bar chart or line graph can be much more evocative of the underlying data than the corresponding table of numbers. The goal of information visualization is to translate abstract information into a visual form that provides new insight about that information. Visualization has been shown to be successful at providing insight about data for a wide range of tasks.

The field of information visualization is a vibrant one, with hundreds of innovative ideas burgeoning on the Web. However, applying visualization to textual information is quite challenging, especially when the goal is to improve search over text collections. As discussed, search is a means towards some other end, rather than a goal in itself. When reading text, one is focused on that task; it is not possible to read and visually perceive something else at the same time. Furthermore, the nature of text makes it difficult to convert it to a visual analogue.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

Proposed visualization engine

The idea was to select existing visualizations for text documents and to combine them in a novel way. Our selection of existing visualizations was based on the assumption to find expressive visualizations keeping in mind the target users, their tasks, their technical environment (typical desktop PC and not a high end workstation for extraordinary graphic representations) and the type of data to be visualized (text documents). The idea was to visualize additional information about the retrieval documents to the user in a way that is intuitive, fast to interpret and can scale to large document sets.

Another important difference of our VIZIR system with existing retrieval systems for the Web is the comprehensive visual support of different steps of the information seeking process. The visual views used in VIZIR supports the interaction of the user with the system during the formulation of the query (e.g. visualization of related terms of the query terms with a graph), during the review of the search results (e.g. visualization of different document attributes like date, size, relevance of the document set with a scatter plot or visualization of the distribution of the relevance of the query terms inside a document with a TileBar), and during the refinement of the query (e.g. visualization of new query terms based on a relevance feedback inside the graph representing the query terms).

Visualization engine component

Systems combing the functionality of retrieval systems with the possibilities of information visualization systems are called visual information seeking systems. The next design decision after retrieving was to transform and save all search results and their characteristics in a local repository (RDBMS) with a specific data schema. The data schema for each document is described in data tables and represents additional information about the retrieved documents. There are two categories of additional information that could be visualized: visualization of document attributes, and visualization of inter-document similarities. It use predefined document attributes (e.g. title, relevance, date, size, document type, server type), and visualizations that show how the retrieved documents relate to each of the terms used in the query (query terms` distribution).

The next step in the development process was to find visual mappings of the data tables to good visual structures. All available attributes of each document are shown in different columns of the table. Each row shows one document. The user has the possibility to sort each document attribute in an increasing or decreasing order or to customize the table to his personal preferences (e.g. to show only the attributes he is interested in or to rearrange the order of the columns).the important design decision was to use a multiple view approach offering the user the possibility to choose the most appropriate visualization view for his current demand or individual preferences.

In all different views we have made extensive use of different interaction techniques (e.g. direct manipulation, details- on - demand, zooming, dynamic queries, sorting) to give the user control over the mapping of data to visual structures [12].

Briefly each technique breaks down into four data stages, three types of data transformation and four types of within stage operators. The four data stages are: value, analytical abstraction, visualization abstraction, and view, as seen in table 1. Transforming data from one stage to another requires one of the three types of data transformation operators: data transformation, visualization transformation, and visual mapping transformation, as seen in table 2 [13].

Table 1: data stages in the data state model

Stages	Description
Value	The raw data
Analytical abstraction	Data about data, or information, meta data
Visualization abstraction	Information that is visualizable on the screen using a visualization technique
view	The end-product of the visualization mapping, where the user sees and interprets the picture presented to her

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

Table 2: transformation operators

Processing step	Description
Data Transformation	Generates some forms of analytical abstraction from the value (usually by extraction)
Visualization Transformation	Takes an analytical abstraction and further reduces it into some form of visualization abstraction, which is visualizable content.
Visual Mapping Transformation	Takes information that is in a visualizable format and presents a graphical view.

Distributed Crawler, distributed search engine, personalization

Distributed search is a search engine model in which the tasks of Web crawling, indexing and query processing are distributed among multiple computers and networks. Originally, most search engines were supported by a single supercomputer. In recent years, however, most have moved to a distributed model. Google search, for example, relies upon thousands of computers are crawling the Web from multiple locations all over the world. Our proposed distributed crawler is in detail in the next section.

In Google's distributed search system, each computer involved in indexing crawls and reviews a portion of the Web, taking a URL and following every link available from it (minus those marked for exclusion). The computer gathers the crawled results from the URLs and sends that information back to a centralized server in compressed format. The centralized server then coordinates that information in a database, along with information from other computers involved in indexing.

When a user types a query into the search field, Google's domain name server (DNS) software relays the query to the most logical cluster of computers, based on factors such as its proximity to the user or how busy it is. At the recipient cluster, the Web server software distributes the query to hundreds or thousands of computers to search simultaneously. Hundreds of computers scan the database index to find all relevant records. The index server compiles the results, the document server pulls together the titles and summaries and the page builder creates the search result pages.

Some projects, such as Wikia Search (formerly Grub) are moving towards an even more decentralized search model. Similarly to distributed computing projects such as SETI@home , many distributed search projects are supported by a network of voluntary users whose computers run client software in the background [17].

VI. Distributed crawler on client machine

The challenging task of indexing the web (usually referred as web-crawling) has been heavily addressed in research literature. However, due to the current size, increasing rate, and high change frequency of the web, no web crawling schema is able to pace with the web. While current web crawlers managed to index more than 3 billion documents, it is estimated that the maximum web coverage of each search engine is around 16% of the estimated web size [14]. Distributed crawling was proposed to improve this situation [19]. This has following benefits: (1) increased resource utilization, (2) effective distribution of crawling tasks with no bottle necks, (3) Configurability of the crawling tasks [14].

The paper describes the design and implementation of a crawler on client machine and delivery of the information from a web browser to search engine’s central database, and preprocessing, storage and retrieval of the information at the central location. The crawler scales to (at least) several hundred pages per second, is resilient against system crashes and other events, and can be adapted to various crawling applications. We present a new model and architecture of the Web Crawler using multiple HTTP connections to WWW. The multiple HTTP connection is implemented using multiple threads and asynchronous downloader module so that the overall downloading process is optimized. Unloads search engine’s crawling task to the millions of client machines that continuously scour the web, allows using processing power of these remote machines to extract the information from a web site that is being currently visited by a web browser. Since the extraction of information from visited pages is occurring in the

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

web browser, there is no need to store these pages on the central location computers of the search engine. Thus, the proposed approach may significantly alleviate three difficult problems of retrieval of information from the web – insufficient efficiency to harvest information from the web by crawlers, enormous requirements for storage of harvested pages, and requirements for processing power to extract the information from the pages.

The new model for the process of information retrieval from the web has the process consisting of the three major conceptual stages: information harvesting by a web browser at a client's location, delivery of the information from a web browser to search engine's central database, and preprocessing, storage and retrieval of the information at the central location.

The user specifies the start URL from the GUI provided. It starts with a URL to visit. As the crawler visits the URL, it identifies all the hyperlinks in the web page and adds them to the list of URLs to visit, called the crawl frontier. URLs from the frontier are recursively visited and it stops when it reaches more than five levels from every home pages of the websites visited and it is concluded that it is not necessary to go deeper than five levels from the home page to capture most of the pages actually visited by the people while trying to retrieve information from the internet. The web crawler system is designed to be deployed on a client computer, rather than on mainframe servers which require a complex management of resources, still providing the same information data to a search engine as other crawlers do, discuss the performance bottlenecks, and describe efficient techniques for achieving high performance [14-16].

V. The proposed distributed crawler

Crawlers consume resources: network bandwidth to download pages, memory to maintain private data structures in support of their algorithms, CPU to evaluate and select URLs, and disk storage to store the text and links of fetched pages as well as other persistent data.

A crawler for a large search engine has two major components, see figure 4. First, it has to have a good crawling strategy i.e. a strategy to decide which pages to download next, it called crawling application. Second, crawling system, it needs to have a highly optimized system architecture that can download a large number of pages per second while being robust against crashes, manageable, and considerate of resources and web servers.

A model of a crawler on the client side with a simple PC, which provides data to any search engines as other crawler provide. To retrieve all webpage contents, the HREF links from every page will result in retrieval of the entire web's content

1. Start from a set of URLs
2. Scan these URLs for links
3. Retrieve found links
4. Index content of pages
5. Iterate

The crawler designed has the capability of recursively visiting the pages. The web pages retrieved is checked for duplication i.e. a check is made to see if the web page is already indexed if so the duplicate copy is eliminated. This is done by creating a data digest of a page (a short, unique signature), then compared to the original signature for each successive visit as given in figure 5. From the root URL not more than five links are visited and multiple seed URLs are allowed. The indexer has been designed to support HTML and plain text formats only. It takes not more than three seconds to index a page. Unusable filename characters such as "?" and "&" are mapped to readable ASCII strings. The WWW being huge, the crawler retrieves only a small percentage of the web.

We have considered two major components of a crawler - collecting agent, and searching agent. The collecting agent downloads web pages from the WWW and indexes the HTML documents and storing the information to a database, which can be used for later search. Collecting agent includes a simple HTML parser, which can read any HTML file and fetch useful information, such as title, pure text contents without HTML tag, and sub-link. The searching agent-searching agent is responsible for accepting the search request from user, searching the database and presenting the search results to user. When the user initiates a new search, database will be searched for any matching results, and the result is displayed to the user, it never searches over WWW but it searches the database only. A high level architecture of a web crawler has been analyzed as in figure 6 for building web crawler system on the client machine. Here, the multi-threaded downloader downloads the web pages from the WWW, and using some parsers the web pages are decomposed into URLs, contents, title etc. The URLs are queued and sent to the downloader using some scheduling algorithm. The downloaded data are stored in a database.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

Soft ware architecture

The architecture and model of our web crawling system is broadly decomposed into four stages. The figure 7 depicts the flow of data from the World Wide Web to the crawler system. The user gives a URL or set of URL to the scheduler, which requests the downloader to download the page of the particular URL. The downloader, having downloaded the page, sends the page contents to the HTML parser, which filters the contents and feeds the output to the scheduler. The scheduler stores the metadata in the database. The database maintains the list of URLs from the particular page in the queue. When the user request for search, by providing a keyword, it's fed to the searching agent, which uses the information in the storage to give the final output.

1. HTML parser

We have designed a HTML parser that will scan the web pages and fetch interesting items such as title, content and link. Other functionalities such as discarding unnecessary items and restoring relative hyperlink (part name link) to absolute hyperlink (full path link) are also to be taken care of by the HTML parser. During parsing, URLs are detected and added to a list passed to the downloader program. At this point exact duplicates are detected based on page contents and links from pages found to be duplicates are ignored to preserve bandwidth. The parser does not remove all HTML tags. It cleans superfluous tags and leaves only document structure. Information about colors, backgrounds, fonts is discarded. The resulting file sizes are typically 30% of the original size and retain most of the information needed for indexing.

2. Creating an efficient multiple HTTP connection

Multiple concurrent HTTP connection is considered to improve crawler performance. Each HTTP connection is independent of the other so that the connection can be used to download a page. A downloader is a high performance asynchronous HTTP client capable of downloading hundreds of web pages in parallel. We use multi-thread and asynchronous downloader. We use the asynchronous downloader when there is no congestion in the traffic and are used mainly in the Internet-enabled application and activeX controls to provide a responsive user-interface during file transfers. We have created multiple asynchronous downloaders, wherein each downloader works in parallel and downloads a page. The scheduler has been programmed to use multiple threads when the number of downloader object exceeds a count of 20.

3. Scheduling algorithm

As we are using multiple downloaders, we propose a scheduling algorithm to use them in an efficient way. The design of the downloader scheduler algorithm is crucial as too many objects will exhaust many resources and make the system slow, too small number of downloader will degrade the system performance. The scheduler algorithm is as follows:

1. System allocates a pre-defined number of downloader objects (20 in our experiment).
2. User input a new URL to start crawler.
3. If any downloader is busy and there are new URLs to be processed, then a check is made to see if any downloader object is free. If true assign new URL to it and set its status as busy; else go to 6.
4. After the downloader object downloads the contents of web pages set its status as free.
5. If any downloader object runs longer than an upper time limit, abort it. Set its status as free.
6. If there are more than predefined number of downloader (20 in our experiment) or if all the downloader objects are busy then allocate new threads and distribute the downloader to them.
7. Continue allocating the new threads and free threads to the downloader until the number of downloader becomes less than the threshold value, provided the number of threads being used be kept under a limit.
8. Go to 3.

This mode is using a least amount of resources on the client machine. When a browser-crawler is site oriented then it crawls in a background the site that the user pointed a browser to. The crawling is performed while the user is viewing already downloaded page, it stops when the user points to a different page, and resumes when the page requested by a user is downloaded and can be viewed by the user. Hence, the crawling is transparent to a user and the only difference besides sending of the downloaded pages to the central location is that the client's CPU cycles are utilized that would be wasted with conventional browser. When the site is completely crawled then a browser-crawler continues breadth-first search (BFS) crawling of sites connected to the one that was completely harvested until a user points the browser to a new web site.

The immediate benefit to the user would be a cache of the crawled pages. Since it is likely that a user will want to view more than one page of a visited web site then his/her surfing experience will be enhanced by loading pages faster from the cache.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

4. Storing the web page information in a database

After the downloader retrieves the web page information from the internet, the information is stored in a database. The information harvesting that is performed by a web browser of a user surfing the web can also involve information extraction. This means that text of the downloaded page is processed to extract words that will be part of the index of the search index at the central location. The only thing that is different from conventional browser is that the downloaded pages are sent to the central location.

The delivery of the information from a web browser to search engine's central database can either be mandatory one-way communications or it could be regulated delivery using two-way communications. When using mandatory delivery client-crawler sends information extracted from each page to the central location unless the page is revisited during the same session and is the same as the one in the cache. When using regulated delivery, client-crawler sends a single request per site to the central location with the URL of the site that the user points to. The central location checks in the database to determine whether the site needs to be crawled (if the site has never been indexed before or if the information has not been updated recently), and sends response to the browser/client-crawler. If the site needs to be crawled then the browser-crawler continues with the BFS crawl. If the site has already been indexed recently, then browser-crawler starts crawling sites that are connected to the current one (if the central location indicates that these sites need to be crawled) until the user points the browser to a different site. This scheme avoids sending duplicate information and generating associated unnecessary traffic that would be resultant of situation when many users visit the same popular sites.

Another opportunity for optimization may be capability of the central location to direct client-crawlers to crawl web sites of interest while user browses a web site that is already completely harvested by other browser-crawlers.

Evaluation

The current model for information retrieval from the web is found to be flawed and inefficient:

- A significant portion of the web cannot be accessed by crawlers and, thus, is not available to retrieve information from.
- Crawlers have relatively low harvesting rate that translates into low refresh rate for indexed pages that leads to 'stale' pages that are out of date.
- Crawlers generate huge amount of traffic that impedes useful communications
- Crawlers require significant resources such as enormous computer farms to harvest and store pages.

The new model for information retrieval from the web has been proposed where web browsers are used as main tool to harvest web pages and extract information from the pages. The extracted information is sent by the browser-crawlers to a central location where it is indexed, stored, and is accessible for retrieval by end users. Traditional crawlers are used as an auxiliary tool to harvest pages from portions of the web that are not currently being harvested by browser-crawlers. Given a large user base of browser-crawlers the new model can provide the following benefits:

- Our crawling system which can be deployed on the client machine to browse the web concurrently and autonomously, it combines the simplicity of asynchronous downloader and the advantage of using multiple threads.
- It reduces the consumption of resources as it is not implemented on the mainframe servers as other crawlers also reducing server management. The proposed architecture uses the available resources efficiently to make up the task done by high cost mainframe servers.
- The coverage of the web can be significantly improved where browser-crawlers can harvest pages that could not be retrieved by traditional crawlers such as static HTML pages that are not reachable to traditional crawlers, dynamic pages that require user interaction, and pages that are prohibited to crawlers
- The harvesting rate can be significantly improved given a large user base of browser-crawlers that may lead to decreased level of 'staleness' of the indexed pages.
- The new model provides near real time dynamic view of the usage of the web providing wealth of information about web usage patterns and statistics.
- The new model may significantly speed up discovery of new web sites since during deployment of a new web site its web pages are accessed with browser-crawlers to test the site.
- The new model may allow reduction in resources required for the process of information retrieval since the tasks of harvesting pages from the web and extracting information from the pages is off loaded to computers hosting browser-crawlers

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

- The use of browser as a crawler may noticeably improve its user's web-surfing experience by providing a large cache of harvested pages.

Personalization

In the modern Web, as the amount of information available causes information overloading, the demand for personalized approaches for information access increases. Personalized systems address the overload problem by building, managing, and representing information customized for individual users. This customization may take the form of filtering out irrelevant information and/or identifying additional information of likely interest for the user.

In the paper, the user can get the personalization benefits in customize his search or his profile, or in indirect way by displaying the suitable visualization technique according his resource's ability. In the following sections, the paper describes how user profile improves retrieval process and helps other users. Section 1 explains what user profile, discusses user profiles specifically designed for providing personalized information access. Section 2 handle regional crawler that not conflict with distributed crawler, and don't mean boundaries with regional word.

There are a wide variety of applications to which personalization can be applied and a wide variety of different devices available on which to deliver the personalized information. Early personalization research focused on personalized filtering and/or rating systems for e-mail, electronic newspapers, Use net newsgroups, and Web documents. More recently, personalization efforts have focused on improving navigation effectiveness by providing browsing assistants, and adaptive Web sites. Because search is one of the most common activities performed today, many projects are now focusing on personalized Web search.

Most personalization systems are based on some type of user profile, a data instance of a user model that is applied to adaptive interactive systems. User profiles may include demographic information, e.g., name, age, country, education level, etc, and may also represent the interests or preferences of either a group of users or a single person. Personalization of Web portals, for example, may focus on individual users, for example, displaying news about specifically chosen topics or the market summary of specifically selected stocks, or a groups of users for whom distinctive characteristics where identified, for example, displaying targeted advertising on e-commerce sites.

In order to construct an individual user's profile, information may be collected explicitly, through direct user intervention, or implicitly, through agents that monitor user activity. Although profiles are typically built only from topics of interest to the user, some projects have explored including information about non-relevant topics in the profile. In these approaches, the system is able to use both kinds of topics to identify relevant documents and discard non-relevant documents at the same time.

Profiles that can be modified or augmented are considered dynamic, in contrast to static profiles that maintain the same information over time. Dynamic profiles that take time into consideration may differentiate between short-term and long-term interests. Short-term profiles represent the user's current interests whereas long-term profiles indicate interests that are not subject to frequent changes over time. For example, consider a musician who uses the Web for her daily research. One day, she decides to go on vacation, and she uses the Web to look for hotels, airplane tickets, etc. Her user profile should reflect her music interests as long-term interests, and the vacation-related interests as short-term ones. Once the user returns from her vacation, she will resume her music-related research, and the vacation information in her profile should eventually be forgotten. Because they can change quickly as users change tasks, and less information is collected, short-term user's interests are generally harder to identify and manage than long-term interests. In general, the goal of user profiling is to collect information about the subjects in which a user is interested, and the length of time over which they have exhibited this interest, in order to improve the quality of information access and infer user's intentions.

As shown in Figure 8, the user profiling process generally consists of three main phases. First, an information collection process is used to gather raw information about the user. The second phase focuses on user profile construction from the user data. The final phase, in which a technology or application exploits information in the user profile in order to provide personalized services.

8. Regional distributed crawler

8.1 Regional Crawler Method

In this method, the crawling strategy is based on users' interests and needs in certain domains. These needs and interests are determined according to common characteristics

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

of the users like geographical location, age, membership and job. Regional Crawler uses these interests as basic data for crawling strategy. In the other words, people in the same region are more likely to search for similar subjects and ignore the other categories that may be important for people in other areas. For example, people in Iran are usually searching for information about soccer and Middle East news, but in the U.S users are more likely to search for baseball events. Even people in a CS department usually look for similar information, (computer science articles for example), so the region could even be defined as small as a LAN. The more a document contains common interests of different domains; the more is its chance for getting crawled.

8.2 Searching and user profile:

The Architecture of most Agent-Based search is based on a Three-Layer Model. The main idea of this Three-Layer model is to divide the internet structure into three layers and devote some particular activities to each layer.

According to figure 9, the requesters are the users who enter the query into the system and have an individual unique user profile. User profile contains the user interests and the results of previous searches. Providers section also contains the services and information of the providers that are being searched for pages related to users' queries. Intermediaries are responsible for matching the users' requests with the information available from the providers or information which have become accessible by the other users according to their user profile.

8.3 Regional Crawler and personalization benefits

As we explained before, current distributed and Agent-Based search engines are usually constructed based on a Three-Layer structure. Generally the main structure of a Personal Agent in most of Agent-Based search engines is just like figure 10.

Search Agent will search the internet and User Interface Agent acts as an interface between the user and the whole Personal Agent structure and enables the collaboration between the user and Agents for searching and entering the queries. Middle Agent plays the most important role in this architecture. It's a bridge between users and providers in the way that providers would announce the services that they provide and users will ask for their needs on the other hand and the Middle Agent would act as a Match Maker between those groups. The Advantage of this methodology is that some user profiles would be devoted to the users which show their needs, interests and past search results. When a query entered by a user and got ready for the search. Middle Agent would get the query and specify the subject of it, then it would search through the user profiles and User Agents to find a similar user according to the public profiles and get information from its past search results for the same or similar queries and send these results as an answer for the user who has entered the query. By adding the Regional Crawler as a Regional Agent to the above architecture we would have Figure 11:

Regional Agent is responsible for collecting the users' interests of a specific region. Users with similar User Profile will be gathered together by Reinforcement Learning methods or Supervised learning depending on the Middle-Agent architecture. Regional Agent will search for users with similar interests and gather them in a unique public agent. Then we devote a special crawler for each regional agent and ask it to crawl the web in the way that it can satisfy the users' interests. This means that the crawler should look for the pages related to region interests before the other topics available on the web. Since the crawlers are in cooperation with Search Agents, Regional Agent will ask Search Agents to update the important web pages (look for important new pages) by announcing the user interests and needs to them. The important point in this architecture is that by implementing the RL methods, regions domain will be unlimited and as an example two Fans of a particular soccer club in two different locations of the world would be in the same region. So by adding a regional Agent to the Architectures above, we expect the important pages from the users (user agents) perspective become updated more frequently [14].

9. Conclusion

Search engine and web information retrieval field still in developing circle not stopped at any station until user's need not curb and World Wide Web expand. Despite new and effective solutions for web information retrieval system are suitable and solved many problems in the past, now, they consider bad and generated problems. For example, WordNet tool that widely used in proposed system, it not suitable for proximity search. The proposed system handle some problems such as: low precision and recall, lack of personalization of information and limited customization to individual users, vocabulary, user search behavior, query formulation, information overload, speed, resources consuming. We will continue to wait problems of new solutions and newer solutions for the previous.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

References

[1] Ricardo Baeza-Yates, Berthier Ribeiro-Neto, "modern information retrieval", ISBN 0-201-39829-X, chapter 10 " user interfaces and visualization" pages 257:

[2] Marti Hearst, "search user interfaces" ,ISBN 9780521113793, copyright @ 2009, chapter 3 "models of information seeking process".

[3] Djoerd Hiemstra, "information retrieval models" published in Goker, A., and Davies, J. " Information Retrieval: Searching in the 21st Century. John Wiley and Sons, Ltd., ISBN-13: 978-0470027622, November 2009.

[4] Changwoo Yoon, "Domain – specific knowledge-base information retrieval model using knowledge reduction", a dissertation presented to the graduated school of the university of Florida in partial fulfillment of the requirements for the degree of doctor of philosophy, 2005.

[5] Jyotsna Gharat, Jayant Gadge, "web information retrieval using WordNet", international journal of computer applications (0975-8887), volume 56 – no.13, October 2012.

[6] Diana Inkpen, "Information Retrieval on the Internet", 2008

[7]usabilityetc.com/articles/information-retrieval-concept-matching/, last visit May 28, 2013

[8] Chia-Hui Chang, Ching-Chi Hsu, "Integrating Query Expansion and Conceptual Relevance Feedback for Personalized Web Information Retrieval", *Computer Networks volume 30(1-7):621-623*.

[9] Query expansion, http://en.wikipedia.org/wiki/Query_expansion, last visit June 1, 2013

[10] WordNet, http://en.wikipedia.org/wiki/Wordnet, last visit June 3, 2013

[11] Christopher D. Manning, Prabhakar Raghavan and Hinrich Schütze, "Introduction to Information Retrieval", ISBN: 0521865719, Cambridge University Press. 2008

[12] Harald Reiterer, Thomas M. Mann, "Visual Information Retrieval for the Web", Proceedings of the Ninth International Conference on Human-Computer Interaction, 2001.

[13] Ed H. Chi, "A Taxonomy of Visualization Techniques using the Data State Reference Model", Proceeding INFOVIS '00 Proceedings of the IEEE Symposium on Information Visualization 2000 page 69, ISBN:0-7695-0804-9

[14] Pirooz Chubak, Milad Shokouhi, "Designing a Regional Crawler for Distributed and Centralized Search Engines" ,http://ausweb.scu.edu.au/ aw04/papers/refereed/ shokouhi/ paper.html, last visit June 13, 2013

[15] Vladislav Shkapenyuk, Torsten Suel, " Design and Implementation of a High-Performance Distributed Web Crawler"

[16] Rajashree Shettar, Dr. Shobha G, "Web Crawler On Client Machine", Proceedings of the International Multi-Conference of Engineers and Computer Scientists 2008 Volume II IMECS 2008, 19-21 March, 2008, Hong Kong.

[17] Margaret rouse, "distributed search" april 2008, http://whatis.techtarget.com/definition/distributed-search", last visit June 23, 2013

[18] Mamoon H. Mamoon, Hazem M. El-Bakry, Amany A. Salama, " Interactive Visualization of Retrieved Information", International Journal of Knowledge Engineering and Research, Vol 2 Issue 4 April 2013 ,ISSN 2319 – 832X

[19] Odysseas Papapetrou, George Samaras, "Minimizing the Network Distance in Distributed Web Crawling", Springer-Verlag Berlin Heidelberg 2004, LNCS 3290, pp. 581–596, 2004.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

A Comparative Study of Replication Techniques in Grid Computing Systems

Sheida Dayyani
Department of Computer Engineering
Sheikh Bahaei University
Isfahan, Iran

Mohammad Reza Khayyambashi
Department of Computer Engineering
University of Isfahan
Isfahan, Iran

***Abstract*—Grid Computing is a type of parallel and distributed systems that is designed to provide reliable access to data and computational resources in wide area networks. These resources are distributed in different geographical locations, however are organized to provide an integrated service. Effective data management in today`s enterprise environment is an important issue. Also, Performance is one of the challenges of using these environments. For improving the performance of file access and easing the sharing amongst distributed systems, replication techniques are used. Data replication is a common method used in distributed environments, where essential data is stored in multiple locations, so that a user can access the data from a site in his area. In this paper, we present a survey on basic and new replication techniques that have been proposed by other researchers. After that, we have a full comparative study on these replication strategies. Also, at the end of the paper, we summarize the results and points of these replication techniques.**

Keywords-comparative study; distributed environments; grid computing; data replication

I. INTRODUCTION

Computing infrastructure and network application technologies have come a long way over the past years and have become more and more detached from the underlying hardware platform on which they run. At the same time computing technologies have evolved from monolithic to open and then to distributed systems [1].

Nowadays, there is a tendency of storing, retrieving, and managing different types of data such as experimental data that are produced from many projects [2]. This data plays a fundamental role in all kinds of scientific applications such as particle physics, high energy physics, data mining, climate modeling, earthquake engineering and astronomy, to cite a few, manage and generate an important amount of data which can reach terabytes and even petabytes, which need to be shared and analyzed [3], [4], [5].

Storing such amount of data in the same location is difficult, even impossible. Moreover, an application may need data produced by another geographically remote application. For this reason, a grid is a large scale resource sharing and problem solving mechanism in virtual organizations and is suitable for the above situation [6], [7], [8]. In addition, users can access important data that is available only in several locations, without the overheads of replicating them locally. These services are provided by an integrated grid service platform so that the user can access the resource transparently and effectively [2], [6]. Managing this data in a centralized location increases the data access time and hence much time is taken to execute the job. So to reduce the data access time, "Replication" is used [3], [4].

The replication is the process of creation and placement of the copies of entities software. The phase of creation consists in reproducing the structure and the state of the replicated entities, whereas the phase of placement consists in choosing the suitable slot of this new duplication, according to the objectives of the replication. So, replication strategy can shorten the time of fetching the files by creating many replicas stored in appropriate locations [9], [10]. By storing the data at more than one site, if a data site fails, a system can operate using replicated data, thus increasing availability and fault tolerance. At the same time, as the data is stored at multiple sites, the request can find the data close to the site where the request originated, thus increasing the performance of the system. But the benefits of replication, of course, do not come without overheads of creating, maintaining and updating the replicas [11].

There is a fair amount of work on data replication in grid environments. Most of the existing work focused on mechanisms for create, decision and delete replicas. The purpose of this document is to review various replication techniques and compare these techniques which have been presented by other researches in different distributed architectures and topologies.

The rest of this paper is organized as follows. In the second section, we present an overview of grid systems, types of grids and topologies that exist for grid systems. The third section describes replication scenario, challenges and parameters of evaluating replication techniques. Section four takes a closer look on basic and new existing data replication strategies in grid environment. In section five, we present a comparative study on the replication techniques that were discussed in the previous Section. Finally, section six will be reserved for the conclusion and a summary of discussed replication techniques results.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

II. GRID SYSTEMS

A large number of scientific and engineering applications require a huge amount of computing time to carry out their experiments by simulation. Research driven by this has promoted the exploration of a new architecture known as "The Grid" for high performance distributed application and systems [12]. In [13], Foster defines the Grid concept as "coordinated resource sharing and problem solving in dynamic, multi-institutional virtual organizations". There are different types and topologies of Grid developed to emphasize special functions that will be defined in the two next sections.

A. Types of Grid

Grid computing can be used in a variety of ways to address various kinds of application requirements and it has three primary types. Of course, there are no hard boundaries between these grid types and often grids may be a combination of two or more of these [14]. Types of grids are summarized below:

- **Computational grid:** Computational grid is focused on setting aside resources specifically for computing power. Such as most of the machines are high-performance servers [14].
- **Scavenging grid:** Scavenging grid is most commonly used with large numbers of desktop machines that are scavenged for available CPU cycles and other resources. Owners of the desktop machines are usually given control over when their resources are available to participate in the grid [14].
- **Data grid:** Data grid is a collection of geographically distributed computer resources that these resources may be located in different parts of a country or even in different countries [10]. For example, you may have two universities doing life science research, each with unique data. A grid connects all these locations and enables them to share their data, manage the data, and manage security issues such as who has access to which data [15], [16].

B. Grid Topologies

In this section we present an overview of major grid topologies. The performance of replication strategies is highly dependent on the underlying architecture of grid [17], [18].

Hierarchical and tree models are used where there is a single source for data and the data has to be distributed among collaborations worldwide [17], [18]. The Figure 1 and Figure 2, shows the hierarchical and tree models respectively.

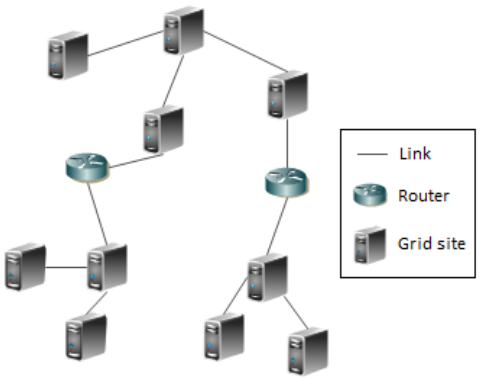


Figure 1. An example of Hierarchical topology.

A tree topology also has shortcomings. The tree structure of the grid means that there are specific paths to the messages and files can travel to get to the destination. Furthermore, data transference is not possible among sibling nodes or nodes situated on the same tier [17], [18].

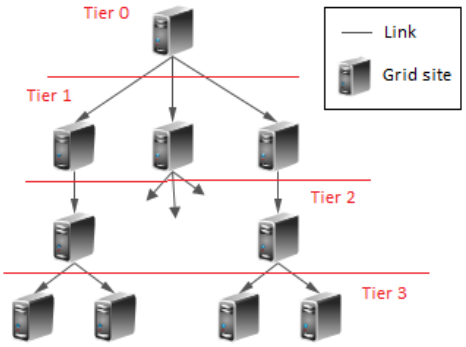


Figure 2. An example of Tree topology.

Peer to Peer (P2P) systems overcome these limitations and offer flexibility in communication among components. A P2P system is characterized by the applications that employ distributed resources to perform functions in a decentralized manner. From the viewpoint of resource sharing, a P2P system overlaps a grid system. The key characteristic that distinguishes a P2P system from other resource sharing systems is its symmetric communication model between peers, each of which acts as both a server and a client [17], [18]. The Figure 3, shows an example of the P2P structure.

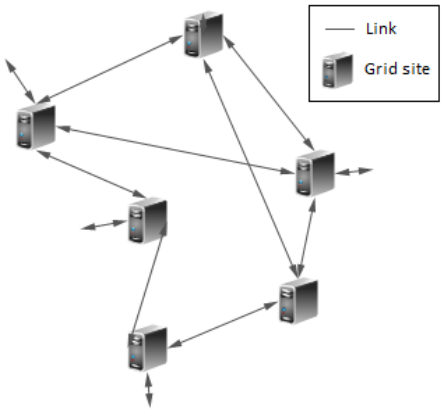


Figure 3. An example of Peer to Peer topology.

Hybrid Topology is simply a configuration that contains an architecture consisting of any combination of the previous mentioned topologies. It is used mostly in situations where researches working on projects want to share their results to further research by making it readily available for collaboration [17], [18]. A hybrid model of a hierarchical grid with peer linkages at the edges is shown in Figure 4.

A hybrid topology can carry features of both tree and P2P architectures and thus can be used for better performance of a replication strategy [15].

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

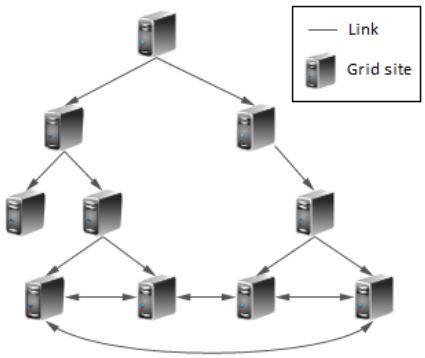


Figure 4. An example of Hybrid topology .

III. DATA MANAGEMENT IN GRIDS

An important technique for data management in grid systems is the replication technique. Data replication is characterized as an important optimization technique in Grid for promoting high data availability, low bandwidth consumption, increased fault tolerance, and improved scalability. The goals of replica optimization is to minimize file access times by pointing access requests to appropriate replicas and pro-actively replicating frequently used files based on access statistics gathered.

Generally, replication mechanism determines which files should be replicated, when the new replicas should be created and where the new replicas should be placed [4], [9], [15]. In the rest of this section, we discuss about data replication scenario, challenges and parameters.

A. Data Replication Scenario

The main aims of using replication are to reduce access latency and bandwidth consumption. The other advantages of replication are that it helps in load balancing and improves reliability by creating multiple copies of the same data [4], [15].

Replication schemes can be classified as static and dynamic. In **static replication**, a replica persists until it is deleted by users or its duration is expired. The drawback of static replication is evident when client access patterns change greatly in the Data. Static replication can be used to achieve some of the above mentioned goals but the drawback with static replication is that it cannot adapt to changes in user behavior. The replicas have to be manually created and managed if one were to use static replication. But, in **dynamic replication**, replica creation, deletion and management are done automatically. Dynamic strategies have the ability to adapt to changes in user behavior [19].

Various combinations of events and access scenarios of data are possible in a distributed replicated environment. The three fundamental questions any replica placement strategy has to answer are as follow that Depending on the answers, different replication strategies are born [4], [15]:

- When the replicas should be created?
- Which files should be replicated?
- Where the replicas should be placed?

B. Data Replication Challenges

Using replication strategies in grid environment may cause some challenges. The four important challenges in replicated environments are as follow [11]:

- Time of creation of a new replica: If strict data consistency is to be maintained, performance is severely affected if a new replica is to be created. As sites will not be able to fulfill request due to consistency requirements.
- Data Consistency: Maintaining data integrity and consistency in a replicated environment is of prime importance. High precision applications may require strict consistency of the updates made by transactions.
- Lower write performance: Performance of write operations can be dramatically lowered in applications requiring high updates in replicated environment, because the transaction may need to update multiple copies.
- Overhead of maintenance: If the files are replicated at more than one site, it occupies storage space and it has to be administered. Thus, there are overheads in storing multiple files.

C. Data Replication Evaluation

Almost all the replications strategies try to reduce the **access latency** thus reducing the job response time and hence increase the performance of the grids. Similarly almost all the replication strategies try to reduce the **bandwidth consumption** to improve the availability of data and performance of the system. The target is to keep the data as close to the user as possible, so that data can be accessed efficiently. Some of the replication strategies explicitly target to provide a **balanced workload** on all the data servers. This helps in increasing the performance of the system and provides better response time. With more number of replicas in a system the cost of maintaining them becomes an overhead for the system. Some of the strategies aim to make only an optimal number of replicas in the data grid. This ensures that the storage is utilized in an optimal way and the **maintenance cost** of replica is minimized. Some strategies target the strategic placement of the replicas along with an optimal number of replicas. The **strategic placement** of replicas is a very important factor because it is integrated with few other very important factors. For example, if the replicas are placed on the optimal locations it helps to optimize the workload of different servers. It is also related with the cost of the maintenance. If a strategy goes on replicating a popular file blindly, it will create too many replicas thus increasing the burden for the system as replica maintenance costs will become too high [20].

Job execution time is another very important parameter. Some replication strategies target to minimize the job execution time with optimal replica placement. The idea is to place the replicas closer to the users in order to minimize the response time, and thus the job execution time. This will increase the throughput of the system [20]. Only a few replication strategies have considered replication as an option to provide **fault tolerance** and **quality assurance**. All replication strategies use subset of these parameters [20].

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

IV. REPLICATION TECHNIQUES

The role of a replication strategy is to identify when a replica should be created, where to place replicas, when to remove replicas and how to locate the best replica [21].

Several replication replacement strategies have been proposed in the past and they are the basics of other replication algorithms. Details of some important basic and new replication algorithms are as follows:

- **NO Replication strategy** will not create replica and therefore, the files are always accessed remotely. One example of the implemented strategy is the SimpleOptimizer algorithm [22], which never performs replication; rather it reads the required replica remotely. SimpleOptimizer algorithm is simple to implement and performs the best relative to other algorithms in terms of the storage space usage, but performs the worst in terms of job execution time and network usage [15].
- **Best client** creates replica at the client that has generated the most requests for a file, this client is called the best client. At a given time interval, each node checks to see if the number of requests for any of its file has exceeded a threshold, then the best client for that file is identified [15].
- **Cascading Replication** supports tree architecture. The data files generated in the top level and once the number of accesses for the file exceeds the threshold, then a replica is created at the next level, but on the path to the best client, and so on for all levels, until it reaches to the best client itself [15].
- **Plain Cashing:** The client that requests a file stores a copy locally. If these files are large and a client has enough space to store only one file at a time, then files get replaced quickly [15].
- **Cashing plus Cascading** combines cascading and plain cashing strategies. The client caches file locally, and the server periodically identifies the popular files and propagates them down the hierarchy. Note that the clients are always located at the leaves of the tree but any node in the hierarchy can be a server. Specifically, a Client can act as a Server to its siblings. Siblings are nodes that have the same parent [15].
- **Fast Spread:** In this method a replica of the file is stored at each node along its path to the client. When a client requests a file, a copy is stored at each tier on the way. This leads to a faster spread of data. When a node does not have enough space for a new replica it deletes the least popular file that had come in the earliest [15].
- **Least Frequently Used (LFU)** strategy always replicates files to local storage systems. If the local storage space is full, the replica that has been accessed the fewest times is removed and then releases the space for new replica. Thus, LFU deletes the replica which has less demand (less popularity) from the local storage even if the replica is newly stored [23].
- **Least Recently Used (LRU)** strategy always replicates files to local storage system. In LRU strategy, the requested site caches the required replicas, and if the local storage is full, the oldest replica in the local storage is deleted in order to free the storage. However, if the oldest replica size is less than the new replica, the second oldest file is deleted and so on [23].
- **Proportional Share Replica (PSR)** policy is an improvement in Cascading technique. The method is a heuristic one that places replicas on the optimal locations by assuming that the numbers of sites and the total replicas to be distributed are already known. Firstly an ideal load distribution is calculated and then replicas are placed on candidate sites that can service replica requests slightly greater than or equal to that ideal load [24].
- **Bandwidth Hierarchy Replication (BHR)** is a novel dynamic replication strategy which reduces data access time by avoiding network congestions in a data grid network. With BHR strategy, we can take benefits from "network-level locality" which represents that required file is located in the site which has broad bandwidth to the site of job execution. BHR strategy was evaluated by implementing in OptorSim simulator and the results show that BHR strategy can outperform other optimization techniques in terms of data access time when hierarchy of bandwidth appears in Internet. BHR extends current site-level replica optimization study to the network-level [25].
- **Simple Bottom-Up (SBU)** and **Aggregate Bottom-Up (ABU)** are two dynamic replication mechanisms that are proposed in the multi-tier architecture for data grids. The SBU algorithm replicates the data file that exceeds a pre-defined threshold for clients. The main shortcoming of SBU is the lack of consideration to the relationship with historical access records. For the sake of addressing the problem, ABU is designed to aggregate the historical records to the upper tier until it reaches the root. The results shown improvements against Fast Spread strategy. The values for interval checking and threshold were based on data access arrival rate, data access distribution and capacity of the replica servers [16].
- **Multi-objective approach** is a method exploiting operations research techniques that is proposed for replica placement. In this method, replica placement decision is made considering both the current network status and data request pattern. The problem is formulated in p-median and p-center models to find the p replica placement sites. The p-center problem targets to minimize the max response time between user site and replica server whereas the p-median model focuses on minimizing the total response time between the requesting sites and the replication sites [26], [27].
- **Weight-based dynamic replica replacement** strategy calculates the weight of replica based on the access time in the future time window on the last access history. After that, calculate the access cost which embodies the number of replicas and the current bandwidth of the network. The replicas with high weight will be helpful to improve the efficiency of data access, so they should be retained and then the replica with low weight will not make sense to the rise of data access efficiency, and therefore, should be deleted. The access history defines based on the zipf-like distribution [28].
- **Latest Access Largest Weight (LALW)** is a dynamic data replication mechanism. LALW selects a popular file for replication and calculates a suitable number of copies and grid sites for replication. By associating a different weight to each historical data access record, the importance of each record is differentiated. A more recent data access record has a larger

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

weight. It indicates that the record is more pertinent to the current situation of data access [29].

- **Agent-based replica placement algorithm** is proposed to determine the candidate site for the placement of replica. For each site that holds the master copies of the shared data files will deploy an agent. The main objective of an agent is to select a candidate site for the placement of a replica that reduces the access cost, network traffic and aggregated response time for the applications. Furthermore, in creating the replica an agent prioritizes the resources in the grid based on the resource configuration, bandwidth in the network and insists for the replica at their sites and then creates a replica at suitable resource locations [7].
- **Adaptive Popularity Based Replica Placement (APBRP)** is a dynamic replica placement algorithm, for hierarchical data grids which is guided by "file popularity". The goal of this strategy is to place replicas close to clients to reduce data access time while still using network and storage resources efficiently. The effectiveness of APBRP depends on the selection of a threshold value related to file popularity. APBRP determines this threshold dynamically based on data request arrival rates [30].
- **Efficient Replication strategy** is a replication strategy for dynamic data grids, which take into account the dynamic of sites. This strategy can increase the file availability, improved the response time and can reduce the bandwidth consumption. Moreover, it exploits the replicas placement and file requests in order to converge towards a global balancing of the grid load. This strategy will focus on read-only-access as most grids have very few dynamic updates because they tend to use a "load" rather than "update" strategy.

There are three steps provided by this algorithm, which are:

1. Selection of the best candidate files for replication; Selected based on requests number and copies number of each files.
2. Determination of the best sites for files placement which are selected in the previous step; Selected based on requests number and utility of each site regarding to the grid.
3. Selection of the best replica; Taking account the bandwidth and the utility of each site [31].

- **Value-based replication strategy (VBRS)** is proposed to decrease the network latency and meanwhile to improve the performance of the whole system. In VBRS, threshold was made to decide whether to copy the requested file, and then solve the replica replacement problem. VBRS has two steps; At the first steps, the threshold will be calculated to decide whether the requested file should be copied in the local storage site. Then at the second stage, the replacement algorithm will be triggered when the requested file needs to be copied at the local storage site does not have enough space. The replica replacement policy is developed by considering the replica's value which is based on the file's access frequency and access time. The experiment results show that the effectiveness of VBRS algorithm can reduce network latency [32].
- **Enhance Fast Spread (EPS)** is an enhanced version of Fast Spread for replication strategy in the data grid. This strategy was proposed to improve the total of response time and total bandwidth consumption. Its takes into account some criteria such as the number and frequency of requests, the size of the replica and the last time the replica was requested. EFS strategy keeps only the important replicas while the other less important replicas are replaced with more important replicas. This is achieved by using a dynamic threshold that determines if the requested replica should be stored at each node along its path to the requester [33].
- **Predictive hierarchical fast spread (PHFS)** is a dynamic replication method in multi-tier data grid environments which is an improve version of common fast spread. The PHFS tries to forecast future needs and pre-replicates the min hierarchal manner to increase locality in accesses and improve performance that consider spatial locality. This method is able to optimize the usage of storage resources, which not only replicates data objects hierarchically in different layers of the multi-tier data grid for obtaining more localities in accesses. It is a method intended for read intensive data grids. The PHFS method use priority mechanism and replication configuration change component to adapt the replication configuration dynamically with the obtainable condition. Besides that, it is developed on the basis of the concept that users who work on the same context will request some files with high probability [34].
- **Dynamic Hierarchical Replication (DHR)** is a dynamic replication algorithm for hierarchical structure that places replicas in appropriate sites. Best site has the highest number of access for that particular replica. This algorithm minimizes access latency by selecting the best replica when various sites hold replicas. The replica selection strategy of DHR algorithm, selects the best replica location for the users running jobs by considering the replica requests that waiting in the queue and data transfer time. It stores the replica in the best site where the file has been accessed most, instead of storing files in many sites [35].
- **Modified Latest Access Largest Weight (MLALW)** is a dynamic data replication strategy. This strategy is an enhanced version of Latest Access Largest Weight strategy. MLALW deletes files by considering three important factors:

1. Least frequently used replicas
2. Least recently used replicas
3. The size of the replica

MLALW stores each replica in an appropriate site in the region that has the highest number of access in future for that particular replica. The experiment results show that MLALW strategy gives a better performance compared to the other algorithms and prevents unnecessary creation of replica which leads to efficient storage usage [36].

V. COMPARATIVE STUDY

In this section, we present a full comparative study on the replication techniques that were discussed in the previous section.

These twenty two replication strategies are compared in the Table 1, Table 2 and Table 3.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

TABLE I. COMPARATIVE STUDY ON REPLICATION TECHNIQUES (A)

Replication technique	Method	Performance metric	Topology	Scalability	Used storage	Simulator	Year	Additional feature
Best Client [15]	Replicates file to site that generates maximum number of requests	Response time , Bandwidth conservation	Tree structure (top-down)	Medium	Low	A grid simulator using PARSEC	2001	Need to compute number of request for each file
Cascading [15]	If number of requests exceeds threshold then replica trickles down to lower tier	Response time , Bandwidth conservation	Tree structure (top-down)	Medium	Medium	A grid simulator using PARSEC	2001	Need to define a threshold for number of requests
Cashing [15]	A requesting client receives the file and stores a replica of it locally	Response time , Bandwidth conservation	Tree structure (top-down)	Medium	High	A grid simulator using PARSEC	2001	_____
Cascading plus Cashing [15]	Joining two replication techniques : Cashing and cascading techniques	Response time , Bandwidth conservation	Peer to Peer structure	High	Medium	A grid simulator using PARSEC	2001	Need to define a threshold for number of requests
Fast Spread [15]	If a client requests a file then a replica of file stores at each node along the path toward the client	Response time , Bandwidth conservation	Tree structure (top-down)	Medium	High	A grid simulator using PARSEC	2001	Need to storing request history to avoid of double replicating
Least Frequently Used (LFU) [23]	Always replicates files to local storage , if no space : delete least accessed files	Job execution time	Flat	Low	High	Optorsim	2003	Need to files access history
Least Recently Used (LRU) [23]	Always replicates files to local storage , if no space : delete oldest file in the storage	Job execution time	Flat	Low	High	Optorsim	2003	Need to files access history

TABLE II. COMPARATIVE STUDY ON REPLICATION TECHNIQUES (B)

Replication technique	Method	Performance metric	Topology	Scalability	Used storage	Simulator	Year	Additional feature
Proportional Share Replication (PSR) [24]	Calculates an ideal workload and distributes replicas	Mean of response time	Tree structure (top-down)	Medium	High	NS2 network simulator (modified)	2004	Need to define ideal workload
Bandwidth Hierarchy Replication (BHR) [25]	Replicates files which are likely to be used frequently within the region in near future	Total job execution time	Hierarchy structure	High	Medium	Optorsim	2004	Need to define network-level locality and regions
Simple Bottom-Up (SBU) [16]	Creates replicas as close as possible to the clients that request the data files with high rates exceeding the pre-defined threshold	Replication frequency, Bandwidth cost, Response time	Tree structure (bottom-up)	Medium	Low	DRepSim (a multi-tier grid simulator)	2005	Need to process records in the access history individually
Aggregate Bottom-Up (ABU) [16]	Aggregates the history records to the upper tier step by step till it reaches the root	Replication frequency, Bandwidth cost, Response time	Tree structure (bottom-up)	Medium	Low	DRepSim (a multi-tier grid simulator)	2005	Need to access history
Multi-objective approach [26], [27]	Reallocates replicas to new candidate sites if a performance metric degrades significantly over best k-time periods	Average response time	Tree structure (top-down)	Medium	Medium	Optorsim	2006	Need to calculate replica relocation cost

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

Weight-based replication [28]	Calculates the weight of replica based on the access time in the future time window, based on the last access history	Effective network usage, Mean job execution time	Flat	Low	Medium	Optorsim	2008	Need to access history that define based on zip-like distribution
Least Access Largest Weight (LALW) [29]	Selects a popular file for replication and calculates a suitable number of copies and grid sites for replication	Network usage, Mean job execution time	Hierarchy structure	High	Medium	Optorsim	2008	Need to find out a popular file and suitable site
Agent based replication [7]	By an agent for each site that holding the master copies, select a candidate site for the placement of replica that exceeds the conditions	Execution time test, Data availability test	Flat	Low	Low	GridSim	2009	Need to define agents

TABLE III. COMPARATIVE STUDY ON REPLICATION TECHNIQUES (C)

Replication technique	Method	Performance metric	Topology	Scalability	Used storage	Simulator	Year	Additional feature
Adaptive Popularity Based Replica Placement (APBRP) [30]	Selects a threshold value related to file popularity and places replicas close to clients to reduce data access time while still using network and storage resource efficiency	Storage cost, Average bandwidth cost, Job execution time	Tree structure	Medium	Medium	Optorsim	2010	Need to determines threshold value dynamically, based on data request arrival rates
Efficient replication strategy [31]	Takes into account the dynamic of sites. Exploits the replicas placement and file request in order to converge towards a global balancing of grid load	Response time, Effective Network Usage	Flat	Low	Medium	Optorsim	2010	Need to considering dynamicity of sites
Value Based Replication Strategy (VBRS) [32]	Calculates the ideal threshold to decide whether the file should be copied or not. Chooses the replica that should be replaced based on the values of the local replicas	Mean job time, Effective Network Usage	Flat	Low	Low	Optorsim	2010	Need to define threshold
Enhanced Fast Spread (EFS) [33]	Uses a dynamic threshold that determines if the requested replica should be stored at each node along its path to the requester. Keeps only the important replicas while other less important replicas are replaced with more important replicas	Total response time, Total bandwidth consumption	Flat	Low	Medium	An event-driven simulator written in java	2011	Need to frequency of requests, the size of the replica and the last time that the replica was requested
Predictive Hierarchical Fast Spread (PHFS) [34]	Tries to forecast future needs and pre-replicates the min hierarchical manner. Uses the hierarchical replication to optimize the utilization of resources	Average access latency	Tree structure	Medium	Medium	Optorsim	2011	Need to considering spatial locality and using predictive methods
Dynamic Hierarchical Replication (DHR) [35]	Selects best replica when various sites hold replicas. Places replicas in appropriate sites that has the highest number of access for that particular replica	Mean job execution time	Hierarchy structure	High	Low	Optorsim	2012	Need to access history
Modified Least Access Largest Weight (MLALW) [36]	Stores each replica in an appropriate site. Deletes files by considering least frequently used replicas, least recently used replicas and the size of the replica factors	Effective network usage, Mean job execution time	Hierarchy structure	High	Low	Optorsim	2012	Need to LRU lists of replicas, LFU lists of replicas and access history

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

VI. CONCLUSION

Replication is a technique used in grid environments that helps to reduce access latency and network bandwidth utilization. Replication also increases data availability thereby enhancing system reliability. This technique appears clearly applicable to data distribution problems in large scale scientific collaborations, due to their globally distributed user communities and distributed data sites.

In this paper, a review and a comparative study has been done on basic and new replication techniques that have been implemented in grids. After a brief introduction, an overview of grid systems, types of grids and grid topologies were presented in Section 2. In Section 3, replication scenario, challenges and ways of evaluating replication techniques were described. In Section 4, a closer look was taken on twenty two of the various existing data replication strategies. In Section 5, a full comparative study was presented on the replication techniques that were discussed in Section 4. And finally, in this section, a table is presented that shows the results of discussed replication techniques.

Table 4 shows the summary and some results of replication techniques that discussed in Section 5.

TABLE IV. SUMMARIZES THE MAJOR RESULTS OF REPLICATION TECHNIQUES IN GRIDS

Replication technique	**Results and Points**
Best Client [15]	▪ Faster average response time than No Replication strategy ▪ Not good overall performance ▪ Not suitable for grid
Cascading [15]	▪ Has an small degree of locality ▪ Not good performance for random access pattern
Cashing [15]	▪ Similar performance as cascading ▪ High response time
Cascading plus Cashing [15]	▪ Client can act as server for sibling ▪ Better performance than cascading ▪ Better performance than cashing
Fast Spread [15]	▪ Consistent performance ▪ High I/O and CPU load ▪ High storage request ▪ Good performance for random access pattern
Least Frequently Used (LFU) [23] Least Recently Used (LRU) [23]	▪ Upgrades overall performance ▪ Upgrades utilization of replica ▪ Better performance than No Replication strategy
Proportional Share Replication (PSR) [24]	▪ Load sharing among replica sites ▪ Better results over cascading technique
Bandwidth Hierarchy Replication (BHR) [25]	▪ Maximizes network-level locality ▪ Good scalability ▪ Better total job times than LRU and LFU
Simple Bottom-Up (SBU) [16] Aggregate Bottom-Up (ABU) [16]	▪ Better results over Fast Spread technique
Multi-objective approach [26], [27]	▪ Good performance in dynamic environments ▪ Dynamic maintainability when performance metric degrades
Weight-based replication [28]	▪ Better performance than LRU and LFU ▪ Has not tested in the real grid systems
Least Access Largest Weight (LALW) [29]	▪ Increases the effective network usage ▪ Better job execution time and effective network usage than LRU, LFU and BHR
Agent based replication [7]	▪ Admissible aggregated response time and data transfer time

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

Adaptive Popularity Based Replica Placement (APBRP) [30]	▪ Improves access time from the client`s perspective ▪ Better performance than Best client, Cascading, Fast Spread, ABU and LRU
Efficient replication strategy [31]	▪ Improves the response time ▪ Increases data availability ▪ Reduces bandwidth consumption
Value Based Replication Strategy (VBRS) [32]	▪ Decreases network latency ▪ Improves performance of the hole system
Enhanced Fast Spread (EFS) [33]	▪ Improves total of response time ▪ Improves total bandwidth consumption ▪ Enhanced version of Fast Spread for replication strategy in data grid
Predictive Hierarchical Fast Spread (PHFS) [34]	▪ Optimizes the utilization of resources ▪ Decreases access latency in multi-tier data grids ▪ Improved version of common Fast Spread ▪ Lower latency and better performance compared with common Fast Spread
Dynamic Hierarchical Replication (DHR) [35]	▪ Prevents unnecessary creation of replica ▪ Efficient storage usage ▪ Minimizes access latency
Modified Least Access Largest Weight (MLALW) [36]	▪ Modified version of LALW strategy ▪ Better performance than LRU, LFU, BHR, LALW and DHR

REFERENCES

[1] K. Sashi and A.S. Thanamani, "A new replica creation and placement algorithm for data grid environment", in *International Conference on Data Storage and Data Engineering (DSDE)*, pp. 265–269, 2010.

[2] M. Li and M. Baker, "*The grid core technologies"*, *John Wiley & Sons*, 2005.

[3] N. Mohd. Zin, A. Noraziah, A. Che Fauzi, and T. Herawan, "Replication Techniques in Data Grid Environments", in *Intelligent Information and Database Systems*, vol. 7197, Eds. Springer Berlin, Heidelberg, pp. 549–559, 2012.

[4] K. Ranganathan and I. Foster, "Decoupling computation and data scheduling in distributed data-intensive applications", in *11th IEEE International Symposium on High Performance Distributed Computing*, pp. 352–358, 2002.

[5] S. Naseera and K.V.M. Murthy, "Agent Based Replica Placement in a Data Grid Environement", in *First International Conference on Computational Intelligence, Communication Systems and Networks*, pp. 426–430, 2009.

[6] F.B. Charrada, H. Ounelli, and H. Chettaoui, "Dynamic period vs static period in data grid replication", in *International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC)*, pp. 565–568, 2010.

[7] W. Hoschek, J. Jaen-Martinez, A. Samar, H. Stockinger, and K. Stockinger, "Data management in an international data grid project", in *Grid Computing*, Springer, pp. 77–90, 2000.

[8] A. Chervenak, I. Foster, C. Kesselman, C. Salisbury, and S. Tuecke, "The data grid: Towards an architecture for the distributed management and analysis of large scientific datasets", in *Network Computing Application*, vol. 23, no. 3, pp. 187–200, 2000.

[9] S. Venugopal, R. Buyya, and K. Ramamohanarao, "A taxonomy of data grids for distributed data sharing, management, and processing", in *Acm Computing Surveys*, vol. 38, no. 1, p. 3, 2006.

[10] K. Ranganathan, A. Iamnitchi, and I. Foster, "Improving data availability through dynamic model-driven replication in large peer-to-peer communities", in *2nd IEEE/ACM International Symposium on Cluster Computing and the Grid*, pp. 376–376, 2002.

[11] S. Goel and R. Buyya, "Data replication strategies in wide area distributed systems", in *Enterprise Service Computing: From Concept to Deployment*, vol. 17, 2006.

[12] R. Buyya, D. Abramson, and J. Giddy, "An architecture for a resource management and scheduling system in a global computational grid", in *The Fourth International Conference/Exhibition on High Performance Computing in the Asia-Pacific Region*, vol. 1, pp. 283–289, 2000.

[13] I. Foster, C. Kesselman, and S. Tuecke, "The Anatomy of the Grid: Enabling Scalable Virtual

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

Organizations", in *International Journal of High Performance Computing Applications*, vol. 15, no. 3, pp. 200–222, 2001.

[14] B. Jacob, "Grid computing: what are the key components", *IBM Developer Works*, 2003.

[15] K. Ranganathan and I. Foster, "Identifying dynamic replication strategies for a high-performance data grid", *Grid Computing*, pp. 75–86, 2001.

[16] M. Tang, B.S. Lee, C.K. Yeo, and X. Tang, "Dynamic replication algorithms for the multi-tier Data Grid", in *Future Generation Computer Systems*, vol. 21, no. 5, pp. 775–790, 2005.

[17] L. Zhu and I.D.R. Frost, "Metadata Management in Grid Database Federation", in *ACM Transactions on Database Systems*, 2007.

[18] Q. Rasool, J. Li, G. Oreku, E. Munir, and D. Yang, "A Comparative Study of Replica Placement Strategies in Data Grids", in *Advances in Web and Network Technologies, and Information Management*, vol. 4537, Springer Berlin, Heidelberg, pp. 135–143, 2007.

[19] X. Dong, J. Li, Z. Wu, D. Zhang, and J. Xu, "On Dynamic Replication Strategies in Data Service Grids", in *11th IEEE International Symposium on Object Oriented Real-Time Distributed Computing (ISORC)*, pp. 155 –161, 2008.

[20] T. Amjad, M. Sher, and A. Daud, "A survey of dynamic replication strategies for improving data availability in data grids", in *Future Generation Computer Systems*, vol. 28, no. 2, pp. 337–349, 2012.

[21] K. Sashi and A.S. Thanamani, "Dynamic replication in a data grid using a Modified BHR Region Based Algorithm", in *Future Generation Computer Systems*, vol. 27, no. 2, pp. 202–210, 2011.

[22] J. Gwertzman, "M. seltzer: The Case for Geographical Push-Cashing," in *5th Conference on Hot Topics in Operating systems*, Orcas Island, USA, 1995.

[23] W.H. Bell, D.G. Cameron, A. P. Millar, L. Capozza, K. Stock-inger, and F. Zini, "Optorsim: A grid simulator for studying dynamic data replication strategies", in International Journal of High Performance Computing Applications, vol. 17, no. 4, pp. 403–416, 2003.

[24] J. Abawajy, "Placement of File Replicas in Data Grid Environments", in *Computational Science (ICCS)*, vol. 3038, Springer Berlin, Heidelberg, pp. 66–73, 2004.

[25] S.M. Park, J.H. Kim, Y.B. Ko, and W.S. Yoon, "Dynamic data grid replication strategy based on Internet hierarchy", in *Grid and Cooperative Computing*, pp. 838–846, 2004.

[26] R.M. Rahman, K. Barker, and R. Alhajj, "Replica placement design with static optimality and dynamic maintainability", in *Sixth IEEE International Symposium on Cluster Computing and the Grid (CCGRID)*, vol. 1, pp.4, 2006.

[27] R.M. Rahman, K. Barker, and R. Alhajj, "Replica placement in data grid: a multi-objective approach", in *Grid and Cooperative Computing (GCC)*, Springer, pp. 645–656, 2005.

[28] W. Zhao, X. Xu, N. Xiong, and Z. Wang, "A weight-based dynamic replica replacement strategy in data grids", in *IEEE Asia-Pacific Services Computing Conference,(APSCC)*, pp. 1544–1549, 2008.

[29] R.S. Chang and H.P. Chang, "A dynamic data replication strategy using access-weights in data grids", in *The Journal of Supercomputing*, vol. 45, no. 3, pp. 277–295, 2008.

[30] M. Shorfuzzaman, P. Graham, and R. Eskicioglu, "Adaptive popularity-driven replica placement in hierarchical data grids", in *The Journal of Supercomputing*, vol. 51, no. 3, pp. 374–392, 2010.

[31] F.B. Charrada, H. Ounelli, and H. Chettaoui, "An efficient replication strategy for dynamic data grids", in *International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC)*, pp. 50–54, 2010.

[32] W. Zhao, X. Xu, Z. Wang, Y. Zhang, and S. He, "Improve the performance of data grids by value-based replication strategy", in *Sixth International Conference on Semantics Knowledge and Grid (SKG)*, pp. 313–316, 2010.

[33] M. Bsoul, A. Al-Khasawneh, E.E. Abdallah, and Y. Kilani, "Enhanced fast spread replication strategy for data grid", in *Journal of Network and Computer Applications* , vol. 34, no. 2, pp. 575–580, 2011.

[34] L.M. Khanli, A. Isazadeh, and T.N. Shishavan, "PHFS: A dynamic replication method, to decrease access latency in the multi-tier data grid", in *Future Generation Computer Systems*, vol. 27, no. 3, pp. 233–244, 2011.

[35] N. Mansouri and G.H. Dastghaibyfard, "A dynamic replica management strategy in data grid", in *Journal of Network and Computer Applications*, vol. 35, no. 4, pp. 1297–1303, 2012.

[36] N. Mansouri, "An Effective Weighted Data Replication Strategy for Data Grid", in *Australian Journal of Basic and Applied Sciences*, vol. 6, no. 10, pp. 336–346, 2012.

AUTHORS PROFILE

Sheida Dayyani was born in Isfahan, Iran 1989. She received the B.Sc. degree in Computer Software Engineering from Sheikh Bahaei University (SHBU), Esfahan, Iran in 2011. Currently, she is a student of Master of Science at Department of Computer Engineering, Sheikh Bahaei University, Esfahan, Iran. Her research areas are Grid Computing, Data Replication and Scheduling algorithms.

Mohammad Reza Khayyambashi was born in Isfahan, Iran in 1961. He received the B.Sc. degree in Computer Hardware Engineering from Tehran University, Tehran, Iran in 1987. He received his M.Sc. in Computer Architecture from Sharif University of Technology (SUT), Tehran, Iran in 1990. He got his Ph.D. in Computer Engineering, Distributed Systems from University of Newcastle upon Tyne, Newcastle upon Tyne, England in 2006. He is now working as a lecturer at the Department of Computer, Faculty of Engineering, University of Isfahan, Isfahan, Iran. His research interests include Distributed Systems, Networking, Fault Tolerance and E-Commerce. He has published in these areas extensively.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

Parallel Implementation of the Single Source Shortest Path Algorithm on CPU–GPU Based Hybrid System

Dhirendra Pratap Singh
Dept. of Computer Science and Engineering
MANIT
Bhopal, India

Nilay Khare
Dept. of Computer Science and Engineering
MANIT
Bhopal, India

***Abstract*—Single source shortest path (SSSP) calculation is a common prerequisite in many real world applications such as traveler information systems, network routing table creation etc., where basic data are depicted as a graph. To fulfill the requirements of such applications, SSSP calculation algorithms should process their data very quickly but these data are actually very large in size. Parallel implementation of the SSSP algorithm could be one of the best ways to process large data sets in real time. This paper proposes two different ways of parallel implementation of SSSP calculation on a CPU-GPU (Graphics Processing Unit)-based hybrid machine and demonstrates the impact of the highly parallel computing capabilities of today's GPUs. We present parallel implementations of a modified version of Dijkstra's famous algorithm of SSSP calculation, which can settle more than one node at any iteration. This paper presents a comparative analysis between both implementations. We evaluate the results of our parallel implementations for two Nvidia GPUs; the Tasla C2074 and the GeForce GTS 450. We compute the SSSP on graph having 5.1 million edges in 191 milliseconds. Our modified parallel implementation shows the three-fold improvement on the parallel implementation of simple Dijkstra's algorithm.**

Keywords-Graph Algorithm; Compute Unified Device Architecture (CUDA); Graphics Processing Unit (GPU); Parallel Processing.

I. INTRODUCTION

Graphs are the best way to represent the data in many real world fields such as computer networks [1, 2], commodity flow networks [3], road networks [4, 5], VLSI design [6, 7, 8], and Robotics [9] etc. The calculation of the SSSP is the most frequent operation on these real world graphs to fulfill the requirements of applications implemented on them. Most graphs representing the data of real time applications have millions of nodes and edges, so many parallel SSSP algorithms have been implemented to solve it in a practical time on the machines like PRAM, CRAY super-computer, and dynamically reconfigurable coprocessor. In this paper we present two parallel SSSP implementations for GPU-based, very cost-effective and highly parallel platform.

Here we give some basics of directed weighted graphs and notations which are used to define the shortest path algorithm in this paper. A graph is represented as a collection of nodes, and links between these nodes called edges, with some attribute related to each edge called the weight of the corresponding edge. If each edge of graph has a fixed starting (source) and ending (destination) node, then such a graph is called a directed graph. The shortest path problem finds a path between two nodes of a weighted directed graph such that the sum of the weights of the edges creating this path is minimal. The single source shortest path problem computes the shortest paths from single source node to all other nodes in a graph.

Let an order pair G = (V, E) represent a graph, V is set of nodes and E represents the set of edges, where |V| and |E| are the number of nodes and number of edges in the graph respectively. Each edge should have a non-negative weight; for an edge (x, y) Ɛ E it is represented as l(x, y). The objective of SSSP calculation is to find the minimum weighted paths from the source node to all other nodes of the graph, minimum weight is denoted by W(v) for a node v Ɛ V. During the execution of the shortest path algorithm a node is called settled if its node weight reaches W(v). Most of the serial shortest path algorithms maintain a tentative weight for each node; let δ (v) represent the tentative weight of node v Ɛ V, which is always infinity or the weight of a path from the source node to node v. Tentative weights are optimized by edge relaxation, i.e. for an edge (x, y) Ɛ E, set δ (y) to be the minimum of δ (y) and δ (x) + l(x, y). In SSSP algorithms tentative weight of node v Ɛ V is optimized until it reaches to W(v).

The rest of this paper is organized into eight sections. Section 2 summaries the work done in the area of parallel SSSP implementation. The CUDA programming model is presented in section 3, and then in section 4 our graph representation is given. Serial modified Dijkstra's algorithm is discussed in section 5. Section 6 presents the parallel modified Dijkstra's algorithm and its different versions on the GPU using CUDA. Section 7 presents the results and performance analysis of our implementations using various graphs. Finally, the Conclusion is discussed in Section 8.

II. RELATED WORK

The Dijkstra's algorithm [10] and the Bellman-Ford algorithm [11, 12] are the two most famous serial SSSP algorithms. To speed up the SSSP calculation process for real world applications there are number of parallel

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

implementations have been proposed for it, which can be divided into two groups. First are those which make parallel the internal operations of the serial SSSP algorithm, and others divide the actual graph into sub-graphs and achieve the parallelization by executing the serial SSSP program for each sub-graph on different machines. A. Crauser et al. [13] implemented a PRAM-based parallel modified Dijkstra's algorithm, which selects a threshold in each iteration and implemented the parallel relaxation of outgoing edges of nodes which satisfy the condition against the threshold. G. Brodal et al. [14] parallelized the queue operation of Dijkstra's algorithm by using a parallel priority queue. J. R. Crobak et al. [15] defined a new algorithm which maintains a list of nodes with their tentative weight in a bucket array, and during each iteration it removes all nodes of the first non-empty bucket and relaxes the outgoing edges of these nodes in parallel. M. Papaefthymiou and J Rodrigue [16] implemented the parallel Bellman-Ford algorithm with some modifications. Y. Tang et al. [17] partitioned the graph and ran a serial SSSP algorithm for each partition on a different machine and then the boundary nodes of adjacent sub-graphs exchange the message and correct their weights. A. Fetterer and S. Shekhar [18] represented the graph in two layers: layer one represents the partitioned sub-graphs and layer two summarizes the boundary graph. First, they parallel ran the SSSP for each sub-graph and updated the node weights in the boundary graph accordingly, and finally ran the SSSP on the boundary graph. These implementations were done on different types of machine such as PRAM, CRAY supercomputers which are very expensive.

Today's GPUs provide us a highly parallel computation platform at very low cost, GPUs are used as co-processors with CPUs for parallel implementation of computer intensive operations of an algorithm, so a number of parallel SSSP algorithms are implemented to run on GPUs as well. Harish et al. [19] have given a parallel implementation, Bellman-Ford's algorithm, where each iteration checks the weight change for any node in the previous iteration; if it is true then crates |V| threads one for each node and each thread relaxes the outgoing edges of its assigned node. S. Kumar et al. [20] have implemented a parallel version of a modified Bellman-Ford algorithm, which can accept negative weighted edges as well and shown good performance for dense graphs. They have created |E| threads, one for each edge, and used just one kernel for both edge relaxation and termination checks. P. J. Martín et al. [21] have shown different ways for the parallel implementation of Dijkstra's algorithm. Basically, each iteration of these implementations finds those queued nodes, which have a weight equal to the current minimum node weight, and relax their outgoing edges in parallel. Relaxation takes place on threads running for these nodes as they have created one for each node.

III. CUDA PROGRAMMING MODEL

In 2007, NVIDIA released a parallel programming interface enabling its GPUs to be used for general purpose computing. This was called compute unified device architecture (CUDA) [22]. It is an extension of the C programming language with some restrictions, and it uses the CPU and GPU simultaneously for execution of code. The CPU executes the serial part of an algorithm, and the data parallel compute intensive jobs are assigned to the GPU. The GPU works on the SIMD (Single Instruction Multiple Data) model in which the same set of instructions are executed on different processing units to work on different data items. GPU uses the thread model to achieve high parallel performance; a large number of threads are created which are mapped on different cores of the GPU.

NVIDIA's GPUs have one or more multiprocessors (MP), which are themselves a collection of multiple independent processing elements (PE) or cores. GPUs have multiple levels of memory for each PE, a fast, private register memory, shared memory which is accessible to all PEs present in any MP, and global, constant and local memories which are present on device DRAM are accessible to all PEs of the GPU. Global and local memories are read/write memory and the constant memory is read only memory. In CUDA [23] we define a set of instructions under a function called kernel and these instructions are executed by all threads. To manage the large number of threads, the threads are grouped to create blocks as shown in Figure1; a block is the group maximum possible number of threads which can assigned to cores under an MP. Multiple threads can be assigned to a core and similarly multiple blocks can be assigned to an MP and each thread gets a unique thread ID in an MP. Blocks can be further grouped to form a grid, where each block gets a unique Block ID. These Thread IDs and Block IDs are used by a thread to uniquely identify the data item on which it has to work. Figure 1 represents the CUDA programming model, here TH denotes a thread, and LM and R represent the local and register memory respectively.

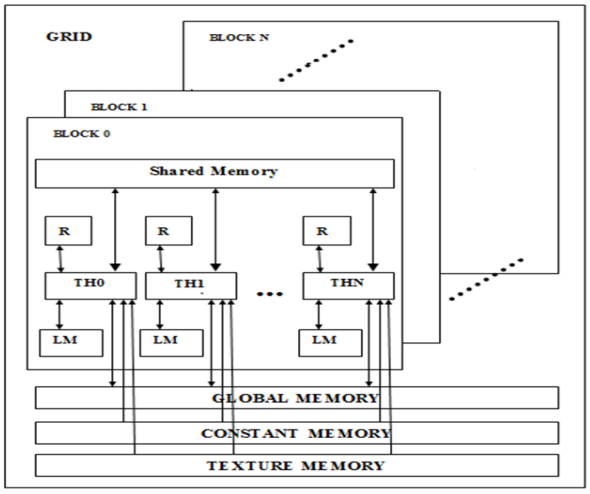


Figure 1. CUDA programming model.

IV. GRAPH REPRESENTATION

We use the adjacency list representation of a graph similar to the Harish et al. [19] method but with a little modification. For our graph representation we have defined three arrays, Node of size |V| and Edge and Weight of size |E|. Each index of the array Node represents a node number and an array value at that index is the start index of the corresponding node's adjacency list in the array Edge. The array Edge stores the adjacency list i.e. the destination node numbers of the outgoing

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

edges of a node, in ascending order of their edge weight. An edge weight is stored in array Weight at the same index as its edge destination node has on the array Edge. We use these three arrays to explain our implementations. Figure 3 shows the adjacency list representation of the graph in the figure 2. Second algorithm uses one more array Edge_SN of size |E|, which stores the source node of each edge of the graph at the same index where edge destination node is stored in array Edge.

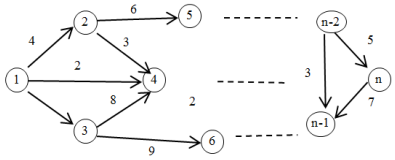


Figure 2. Example graph.

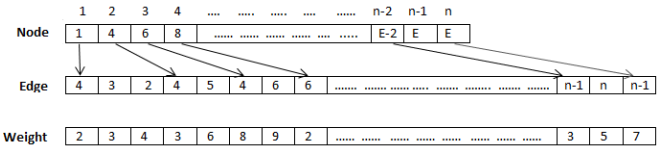


Figure 3. Adjacency list representation of example graph.

V. MODIFIED DIJKSTRA ALGORITHM

Dijkstra's [10] is a label setting SSSP algorithm, defined for a positive edge weighed directed graph. It initializes the source node weight to zero and all other nodes to infinity. It divides the node set of a graph into three groups. First is the group of nodes got their minimum weight, is called, settled nodes. Second group have nodes whose weight is neither minimum nor infinity, which is called queued nodes. Last group is the group of nodes whose weight is still infinity, which is called unreached nodes. In Dijkstra's algorithm each iteration removes a queued node with minimum weight and shifts it to the settled node group and relaxes the node's outgoing edges. These operations are repeated until the all nodes not become the part of settled group.

Dijkstra's algorithm selects only one queued node during any iteration but according to A. Crauser et al. [13] it could be possible that the multiple nodes of queued group have got their minimum weight, which can be transferred to a settled group simultaneously with relaxing their respective outgoing edges. But the problem with this idea is how to identify such queued nodes. Some solutions for this problem have been suggested, such as all those nodes for which $\delta(v) \leq L$, where L is $\min\{\delta(u) + l(u, z)$: $u \in V$ is queued and $(u, z) \in E$ such that $\forall(u, x) \in E$, $l(u, x)$ is minimum$\}$. Here, the minimum weighed outgoing edge of each node has been pre-computed during the initialization of the graph.

A modified Dijkstra's algorithm is defined in Algorithm 1. It initializes the source node weight by 0 and all other nodes by infinity and Thr_d is initialized by 0. In each iteration of step 6 this algorithm calculates the latest value of Thr_d with the help of all those nodes which are present in the group of queued nodes. After calculation of the current Thr_d value all those queued nodes whose weight is less than or equal to the Thr_d value are removed from the queue and their outgoing edges are relax. After the relaxation of any edge, its destination node is added to the group of queued nodes. In the Algorithm 1 edge(x, y) and weight[x, y] represents an edge and its weight respectively, where x is the source node and y is the destination node of that edge.

Algorithm 1: Modified Dijkstra Algorithm(Graph G (V, E), Source node)

Create an array node_weight of size |V|, a variable INFINITY with a very large number assigned to it and a variable Thr_d to store the threshold value
Begin
[1] **for** all node n **do**
[2] node_weight[n] =INFINITY
[3] **End for**
[4] node_weight[Source node]=0,Thr_d=0
[5] Add the Source node in queue
[6] **while** (Thr_d < INFINITY) **do**
[7]Thr_d=INFINITY
[8]**for** all queued node m **do**
[9] **for** first edge(m, p) , where p is unsettled **do**
[10]Thr_d=min(Thr_d, node_weight[m] + weight[m, p])
[11] **end for**
[12] **end for**
[13] **for** all queued node m **do**
[14] **if**(node_weight[m]<=Thr_d) **then**
[15]**for** all edge [m, p] **do**
[16]node_weight[p]=min(node_weight[p], node_weight[m] + weight[m, p])
[17] Add node p in queue
[18] **End for**
[19] **End if**
[20] **End for**
[21] **End while**
END

VI. PARALLEL IMPLEMENTATION

In this section we explain two different approaches for the parallel implementation of threshold based modified Dijkstra's algorithm on a GPU. Method 1 is called node based implementation because it creates |V| threads during the call of relax kernel and the method 2 is called edge based implementation as it crates |E| threads to call the relax kernel.

A. Method 1(Node based implementation)

The first approach to parallel implementation of threshold based Dijkstra's algorithm in the CUDA environment is defined in Algorithm 2. It uses the three kernels for its basic operations. The first kernel is the INITIALIZATION defined in Algorithm 3; it assigns the initial distance value of each node from the source node and initializes the values of the Flag array corresponding to the each node. The second kernel is the THRESHOLD defined in algorithm 4 which calculates the new threshold value with the help of all those nodes which are present in the queue. The third kernel is the RELAX defined in Algorithm 5, for relaxation of the outgoing edges of those nodes whose current distance from the source node is less than or equal to the current threshold value.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

First step of Algorithm 2 assigns the initial weight to each node and the corresponding flag value in Mask array. For this initialization it creates |V| threads to call the kernel INITIALIZATION, one thread for each node. In the next steps Thr_d is initialized with zero and then the algorithm recalculates the new Thr_d value inside the loop. If in any iteration Thr_d value remains unchanged, then the loop will be terminated because the algorithm sets the Thr_d to infinity whenever it comes inside the loop.

Algorithm 2: Node_based_SSSP (Graph G (V, E), Source node)

Create an array Node_weight of size |V|, a Boolean array Mask of size |V|, a variable INFINITY with a very large number assigned to it and a variable Thr_d to store the threshold value.
Begin
[1] INITIONALIZATION(Node_weight, Mask, source node) for all nodes of the graph in parallel
[2]Thr_d=0
[3]**while** (Thr_d< INFINITY) **do**
[4]Thr_d= INFINITY
[5]THRESHOLD(Node, Node_ weight, edge weight, mask, Thr_d, INFINITY) for all nodes of the graph in parallel
[6]RELAX(Node, Node_weight, Edge, Weight, Mask, Thr_d) for all nodes of the graph in parallel
[7]**End while**
End

Algorithm 3:INITIALIZATION(Node_weight, Mask, Source node)

Begin
[1]id=getThreadID
[2]Node_weight[id]=INFINITY
[3]Mask[id]=0
[4]**if** (id=source node) **then**
[5]Node_weight[id]=0
[6]Mask[id]=1
[7]**End if**
End

For threshold calculation Algorithm 2 creates |V| threads to call the THRESHOLD kernel, one for each node. This is a minimum calculation process which is inherently serial, but still we do it in a single step. P. J. Martín et al. [21] have shown the minimum calculation in two steps. First find the minimum for each CUDA block and then calculate the global minimum value from all these blocks' minimum values. We have also tried to implement a similar minimum calculation but it does not give any performance gain compared to the minimum calculation in a single step because it is necessary to synchronize all the threads in any block, which adds a time overhead.

Kernel THRESHOLD calculates a new Thr_d value at any iteration. Here, each thread checks that its assigned node's Mask is not set and its Node weight is less than infinity. A node which satisfies the previous conditions can participate in a minimum Thr_d value calculation. Out of all outgoing edges of this node, the edge with the minimum weight and whose destination node is not settled will be selected. The sum of this node's weight and its selected edge's weight sets a Thr_d value. This threshold calculation is an ATOMIC operation because it could be possible that multiple threads will try to update the value of the Thr_d variable at the same time.

Algorithm 4: THRESHOLD (Node, Node_ weight, edge weight, mask, Thr_d, INFINITY)

Begin
[1]id=getThreadID
[2]**if**(Maks[id]!=1 AND Node_weight < INFINITY) **then**
[3]f**or** all m successor of id **do**
[4]**if**(Mask[m]!=1) **then**
[5]**Begin ATOMIC**
[6]**if**(Thr_d > node_weight[id] + weight[id,m]) **then**
[7]Thr_d= node_weight[id] + weight[id,m]
[8]**End if**
[9]**End ATOMIC**
[10]**End if**
[11]**End for**
[12]**End if**
End

After the new threshold calculation the Algorithm 2 calls the RELAX kernel with creating |V| threads, one for each node of the graph. Each thread of the RELAX kernel checks that its corresponding node's Mask is not set and its weight is less than the current Thr_d value, then sets the node's Mask value to 1 and relax all its outgoing edges. Relaxation is performed in an atomic manner to avoid a read/write conflict.

Algorithm 5: RELAX(Node, Node_weight, Edge, Weight, Mask, Thr_d)

Begin
[1]Id=getThreadID
[2]**if**(Maks[id]!=1 AND Node_weight< Threshold) **then**
[3]Mask[id]=1
[4]**for** all nodes m Ɛ V and successor of node id **do**
[5]**Begin ATOMIC**
[6]**if**((Node_weight[m] >Node_weight[id] + weight[id, m]) **then**
[7]Node_weight[id] = Node_weight[id] + Weight[id, m]
[8]**End if**
[9]**End ATOMIC**
[10]**End for**
[11]**End if**
End

This algorithm creates |V| threads at the call of the RELAX kernel, one for each node, so during the relaxation each node has to check all its outgoing edges one by one. This serial part of the relax function can be a big overhead if a node has thousands of outgoing edges. To avoid this serialization we have proposed our second method.

B. Method 2(Edge based implementation)

Edge based implementation of modified Dijkstra's algorithm is defined in Algorithm 6. This implementation works similar to the first implementation up to the calculation of new value of threshold, but the relax operation is different here. It creates |E| threads during the call of the relaxation kernel and each thread works for one edge of the graph. If the source node of an edge has node weight less than or equal to

Identify applicable sponsor/s here. *(sponsors)*

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

the current threshold value then Algorithm 6 relaxes this edge. In the first implementation, relaxation of the all outgoing edges of any node takes place serially in a thread, but this second implantation removes the serialization and so we achieve a greater degree of parallelization here.

Algorithm 6: Edge_based_SSSP (Graph G (V, E, W), Source node)

Create an arrary Node_weight of size |V|, a Boolean array Mask of size |V|, a variable INFINFITY and assign it a very large number and a variable Thr_d to store the threshold value.
Begin
[1] INITIONALIZATION(Node_weight, Mask) for all nodes of the graph in parallel
[2]Thr_d=1
[3]**while** (Thr_d< INFINITY) **do**
[4]Thr_d= INFINITY
[5]THRSHOLD(Node, Node_ weight, edge weight, mask, Thr_d, INFINITY) for all nodes of the graph in parallel
[6]RELAX_2(Node, Node_weight, Edge, Weight, Edge_SN, Mask, Thr_d) for all edges of the graph in parallel
[7]**End while**
End

The relaxation operation for the second implementation is defined in Algorithm 7, named RELAX_2. In this implementation Weight[x] represents the weight of edge number x. Algorithm 6 creates |E| threads during the call of the RELAX_2 kernel, one thread for each edge of the graph. Each thread executing the RELAX_2 kernel checks that its corresponding edge source node's weight is less than the Thr_d and its Mask is not set. Then it sets its Mask to 1 and relaxes the edge. Relaxation of this edge is done by an ATOMIC operation to avoid a read/write conflict between different threads.

Algorithm 7: RELAX_2(Node, Node_weight, Edge, Weight, Edge_SN, Mask, Thr_d)

Begin
[1]id=getThreadID
[2]**if**(Maks[Edge_SN[id]]!=1 AND Node_weight[Edge_SN] < Thr_d) **then**
[3]Maks[Edge_SN[id]]=1
[4]**Begin ATOMIC**
[5]**if**((Node_weight[Edge[id]] > Node_weight[Edge_SN[id]] + Weight[id]) **then**
[5]Node_weight[Edge[id]] = Node_weight[Edge_SN[id]] + Weight[id])
[7]**End if**
[8]**End ATOMIC**
[9]**End if**
END

VII. PERFORMANCE ANALYSIS

To evaluate the performance of our parallel modified Dijkstra's implementations we use the different types of standard graphs available on Stanford graph database [24].These graphs are tested graphs with different graph properties are already calculated. The standard graphs used are the web graphs, the computer network graphs, the social network graphs and the citation graphs. These are all directed graphs and edge weights 1 to 10 are randomly assigned at the time of pre-processing to store them on our defined data structure. These graphs have 6 K to 80 K vertices and 20 K to 1.5 lakh edges. The degree of these graphs is so low that they are considered as sparse graphs. To test our algorithm on some large graphs we use the graphs having 0.1 M to 1 M nodes and 1 M to 5.1 M edges of varying out-degrees.

A. Experimental Setup

To evaluate the performance of our implementations we use two different machines having different software and hardware configurations. Setups of the both machines are shown in table1.

TABLE I. EXPERIMENTAL SETUP

SETUP 1	SETUP2
CUDA 4.1	CUDA 5.0
NVIDIA GeForce GTS 450 GPU	NVIDIA Tesla C2075 GPU
Compute Capability 2.1	Compute Capability 2.0
192 Cores and 4 Multiprocessors	448 Cores and 14 Multiprocessor
1 GB Dedicated GPU Memory	4 GB Dedicated GPU Memory
Intel Core i5 CPU @ 3.20 Ghz	2 x CPU Intel HEX(6) Core Xeon X 5660, 2.8GHz,
2 GB RAM	24 GB RAM
Windows 7 Professional x86	Windows 7 Professional 64-bit OS
Visual Studio Professional 2008	Visual Studio Professional 2010

B. Results

In this section we show the results of our parallel implementations of modified Dijkstra's algorithm and one previous parallel implementation of simple Dijkstra's algorithm [19] for small and large graphs on both the setups. The calculation time of an algorithm is affected by the size and degree of the graph as well as the processing complexity of the algorithm. Our edge-based implementation has reduced the complexity of the relax function from O(|V|) to O(1), because it has removed the condition where a settled node was relaxing all of its outgoing edges one by one. In the case of a very sparse graph there is no great difference in the results of either implementation because in these graphs out-degree of nodes are in O(1), which makes the both implementations very much similar. In the case of dense graphs, the edge-based implementation gives better results as its each threads just processes one edge but the node-based implementation can have O(|V|) edges to process.

Results are shown in the form of figures with x and y axes. Here, the x-axis shows the size of the test graphs in terms of the number of edges and the y-axis shows the computation time of the algorithms in milliseconds.

Figures 4 and 5 show the results of parallel Dijkstra's implementation (S_Dij) and our parallel node-based (M_N_Dij) and parallel edge-based (M_E_Dij) implementations of modified Dijkstra's algorithm in both setup 1 and setup 2 respectively for small graphs with 3 to 5 average out-degree for the edges. They show that for a graph with 60K nodes and 0.14M edges our first node-based algorithm gives

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

the processing time of 9.1 ms and the edge-based one gives 6 ms for a Telsa C2075 graphics card. Our Edge based implementation is giving approximately three time speedup as compare to parallel implementation of Dijkstra's algorithm for small graphs.

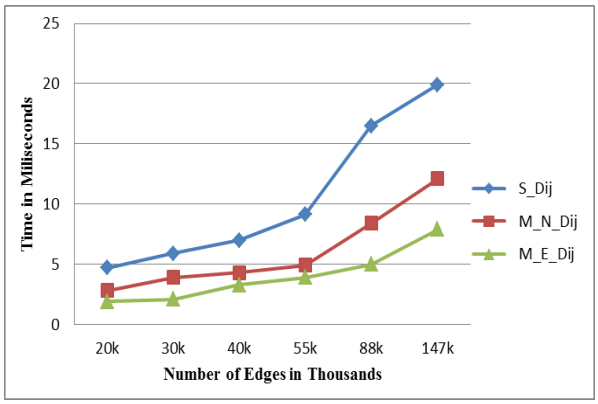


Figure 4. SSSP algorithms timing for small graphs in setup 1.

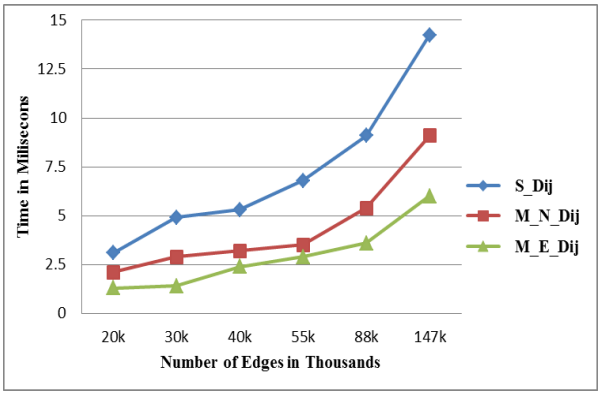


Figure 5. SSSP algorithms timing for small graphs in setup 2.

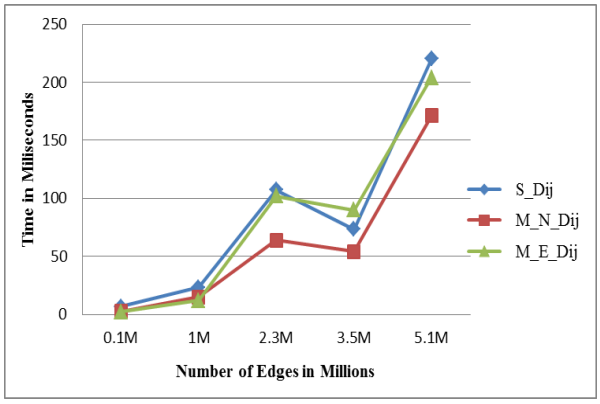


Figure 6. SSSP algorithms timing for large graphs in setup 2.

Figures 6 show the results of parallel implementation of Dijkstra's algorithm and our both implementations for large graphs having average out-degree 5 to 10 on setup 2. It shows that as the graph size and edge density increases our edge based implementation is not giving better results as compared to the node based implementation, because we have to create too many threads for small job and system is wasting its time in scheduling of unused threads. From Figure 6 it is clear that all algorithms are taking relatively more time to processes the graph having 2.3 million edges, because the number of levels in shortest path sub tree for this graph is much higher as compare to other the graphs.

VIII. Conclusion

In this paper, we have shown parallel and efficient versions of Dijkstra's SSSP algorithm on a GPU-based machine using the CUDA programming model. We have shown that after decreasing the processing complexity of the algorithm, it gives better performance. We have also removed the problem present in previous threshold-based algorithm, which was also considering the settled destination node in the threshold calculation. This problem is removed by the addition of the concept of run time minimum weighted outgoing edge selection for any node. We have tested our algorithm on different types and sizes of graphs and our node based implementation has processed the graph of 5.1 million edges in 191 milliseconds. In best case we got 3x speed-ups as compared to the previous GPU-based implementation of Dijkstra's algorithm.

To get more performance gain we can use the different levels of memory available on a GPU, such as fast register memory, to store the edge weight. We can extend our implementation to machines having multiple GPUs and further we can implement it on cluster where each cluster node has a GPU.

References

[1] Nuno M. Garcia, Przemyslaw Lenkiewicz, Mário M. Freire and Paulo P. Monteiro, "On the Performance of Shortest Path Routing Algorithms for Modeling and Simulation of Static Source Routed Networks – an Extension to the Dijkstra Algorithm", Second International Conference on Systems and Networks Communications (ICSNC 2007), IEEE, 2007.

[2] Kun Tan, Qian Zhang and Wenwu Zhu, "Shortest Path Routing in Partially Connected Ad Hoc Networks", Global Telecommunications Conference, GLOBECOM '03. IEEE, 2003.

[3] Lifeng Lai and Siu-Wai Ho, "Simultaneously Generating Multiple Keys and Multi-Commodity Flow in Networks",Information Theory Workshop (ITW), IEEE , pp. 627 - 631, 2012 .

[4] Qingquan Li,Zhe Zeng, and Bisheng Yang,"Hierarchical model of road network for route planning in vehicle navigation systems", Intelligent Transportation Systems Magazine, IEEE,Vo. 1, no. 2, pp.20-24, 2009.

[5] Qing Song and Xiaofan Wang, "Efficient Routing on Large Road Networks Using Hierarchical Communities", IEEE Trans. on Intelligent Transportation Systems, Vo.12, No.1, pp.132-140, 2011.

[6] Gaurav Trivedi, Sumit Punglia, H. Narayanan, "Application of DC Analyzer to Combinatorial Optimization Problems", 20th International Conference on VLSI Design (VLSID'07), IEEE, 2007.

[7] Francois R. Boyer and EL Mostapha Aboulhamid, "Optimal Design of Synchronous Circuits Using Software Pipelining Techniques", ACM Transactions on Design Automation of Electronic Systems,Vo. 6, pp. 516-532, 2001.

[8] Ching-Wei Yeh, Chung-Kuan Cheng and Ting-Ting Y. Lin,"A Probabilistic Multicommodity-Flow Solution to Circuit Clustering Problems", ICCAD-92, IEEE, 1992.

[9] Fajie Li, Reinhard Klette and Sandino Morales, "An Approximate Algorithm for Solving Shortest Path Problems for Mobile Robots or Driver Assistance", Intelligent Vehicles Symposium, IEEE, 2009.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

[10] E. W. Dijkstra, "A note on two problems in connexion with graphs", Numerische Mathematik 1, 269–271, 1959.

[11] R. E. Bellman, "On a routing problem", Quarterly of Applied Mathematics, 16: 87-90, 1958.

[12] L. R. Ford Jr., and D. R. Fulkerson, "Flows in Network", Princeton University Press, 1962.

[13] A. Crauser, K. Mehlhom, U. Meyer and P. Sanders, "A parallelization of dijkstra's shortest path algorithm", LNCS 1450, pp. 722-731, 1998.

[14] G. Brodal, J. Traff and C. D. Zarolingis, A parallel priority data structure with applications, 11th International Parallel Processing Symposium, IEEE, 1997.

[15] J. R. Crobak, J. W. Berry, K. Madduri and D A. Bader, "Advanced shortest paths algorithms on a massively-multithreaded architecture", Parallel and Distributed Processing Symposium, IEEE, 2007.

[16] M. Papaefthymiou and J Rodrigue,. "Implementing parallel shortest-paths algorithms", DIMACS Series in Discrete Mathematics and Theoretical Computer Science, pp. 59-68, 1994.

[17] Y. Tang, Y. Zhang and H. Chen, A parallel shortest path algorithm based on graph-partitioning and iterative correcting, 10th IEEE International Conference on High Performance Computing and Communications, IEEE, 2008.

[18] A. Fetterer and S. Shekhar, A performance analysis of hierarchical shortest path algorithms, Ninth IEEE International Conference on Tools with Artificial Intelligence,IEEE, 1997.

[19] P. Harish and P. J. Narayanan, 'Accelerating large graph algorithms on the GPU using CUDA", in High Performance Computing – HiPC 2007, S. Aluru, M. Parashar et al. (Eds.), Springer Berlin Heidelberg 2007, pp. 197-208.

[20] S. Kumar, A. Misra, R. S. Tomar, "A Modified Parallel Approach to Single Source Shortest Path Problem for Massively Dense Graphs Using CUDA", Int. Conf. on Computer & Comm. Tech. (ICCCT), IEEE, 2011.

[21] P. J. Martín, R. Torres, and A. Gavilanes, 'CUDA Solutions for the SSSP Problem", in Computational Science – ICCS 2009, G. Allen et al. (Eds.), Springer-Verlag Berlin, Heidelberg, 2009, pp. 904–913.

[22] J. Nickolls, I. Buck, M. Garland, and K. Skadron, "Scalable Parallel Programming with CUDA," ACM Queue, vol. 6, no. 2, pp. 40-53, 2008.

[23] NVIDIA Corporation, CUDA C programming guide (2013), http://docs.nvidia.com/cuda/pdf/CUDA_C_Programming_Guide.pdf.

[24] Jure Leskovec, Stanford Large Network Dataset Collection, Stanford University, http://snap.stanford.edu/data/.

Analysis Of The Methodology Required For The Simulation Of Handover Failure In GSM Network

Syed Foysol Islam
Faculty of Engineering
University Of Development Alternative (UODA)
Dhaka, Bangladesh

Fahmi Ahmed
Faculty of Engineering
University Of Development Alternative (UODA)
Dhaka, Bangladesh

***Abstract*—This research paper shows the methodology needed for the simulation of call drop & handover failure in GSM network tele-traffic through OMNeT++ simulation tool under Windows platform. It measures design conditions and minimum quality standards should provide for operation and simulates call drop and hand over failure in GSM tele-traffic. The simulator has been programmed in OMNeT++, is a discrete event simulator focused on research of wired or wireless networks.**

Keywords-Call drop; Handover; Wireless network; Simulator;

I. INTRODUCTION

GSM is a digital mobile telecommunication standard in most of the world. It was created by CEP (European Conference of Postal and Telecommunications) and developed by ETSI (European Telecommunications Standards Telecommunications) as the standard for European mobile telecommunication. ETSI is responsible for laying down different standards; GSM digitizes the data, and then send the data through a channel, it works in different frequency bands which were defined by the CEP, the 890MHz - 915MHz bands for the mobile stations, 935MHz - 960MHz bands for base stations. Currently, GSM works also on 1800 MHz and 1900 MHz bands. Moreover, it also includes data support GPRS (General Packet Radio Service) and EDGE (Enhanced Data Rates for GSM Evolution). GPRS is a service for sending and receiving data packets at high speeds from 56Kbps to 114 Kbps, while EDGE stresses with speeds ranging from 110-130 Kbps to a peak speed 473 Kbps. GSM technology provides crisp and clear voice calls, different types of mobile devices based on this technology and allows communication between users in different locaions around the world. However, this research simulates a GSM cellular mobile telephone system, through OMNeT simulation tool under Windows platform. It measures design conditions and minimum quality standards should provide for operation and simulates call drop and hand over failure in GSM tele-traffic. OMNET is a discrete event simulator focused on research of wired or wireless networks. Through this tool, we can measure the networks behavior under minimum quality standards such voice service such as Web accessibility, service accessibility and integrity of service. As the Methodology is developed, it implements the following steps: Analysis, where the variables of entrance and exit are defined. Formulation, considers what it wants to simulate. Data, the type of distribution is moderate probabilistic of GSM system. Implementation is realized in the simulation in different scenes from the results of the distribution. Verification and Validation verify that how much it approaches the real world. And, finally, Results and Conclusions, where the analysis becomes appropriate. OMNET contains graphical publishers Scalars and Plovers which generates the result in graphical form, therefore facilitating the analysis of the simulation.

II. SIMULATION TOOLS FOR METHODOLOGY

When deciding to simulate a system, we should be able to choose the right simulation tool (simulator) as appropriate from the papers, journals and their functions, interfaces and specifications:

NS2: Network Simulator is a discrete event simulator developed by the UC Berkeley to model IP-type networks. The simulation takes into account the structure (topology) of the network and same packet traffic in order to create a kind of diagnosis that shows the behavior obtained by having a network with certain characteristics. On the other hand, it implements protocols such as TCP and UDP. It is possible to make them behave like a traffic FTP, Telnet, Web, CBR and VBR. It also handles various mechanisms that generate queue in routers: Drop Tail, RED CQB and Dijkstra's algorithm. Currently, the NS project is part of a project being developed VIN Tools to visualize the results of a simulation, for example, a graphical interface.

The graph of a simplified view of NS is shown below:

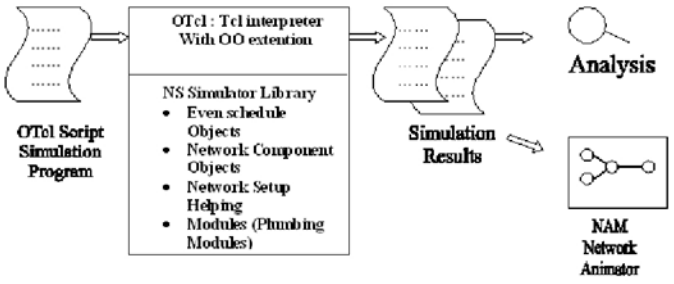


FIGURE-1: Function of Vista Simulation [1]

As we can see, it starts with a script that comes OTCL do encodes the user to simulate. INPUT is the only one that gives the user schedule. The rest is the internal processing of NS. The simulation is a file that can be quite uncomfortable to read

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

or analyze without using a special application that can be displayed using an interface graph. The script is a TCL file written in object-oriented, i.e. OTCL that has various internal components shown in the table in the middle of the Figure. These components are configured network topology, events that load the necessary functions for simulation, planning to start or terminate traffic of a particular package, among other things. NS2 simulation tool is a discrete event focused on network research. NS provides simulation for TCP, routing and multicasting on wired or wireless networks. Furthermore, NS2 is used in educational settings to simulate simple networks that help to understand different protocols and observe how it is produced sending packets between nodes, and other functions. To define a NS2 simulation using a scripting language called TCL the various elements of a network NS2 also has a graphical interface for viewing the simulations called NAM (Network Animator). NAM in turn, provides a graphic editor for viewing results [2].

OMNET: It is a tool implemented to simulate objects and modular discrete events in communications networks. It also has a lot of tools and an interface that can be run on platforms such as Windows and UNIX distribution for the development of this project implemented in the Windows XP platform. Omnet makes use of various compilers such as C + +, in this case Visual Basic 6.0 can be used. Moreover, we can say that Omnet is a free version for academic purposes. OMNEST commercial version is developed by Omnet Global, Inc. OMNeT + + [3].

III. METHODOLOGY OF DEVELOPMENT

A. System Defination:

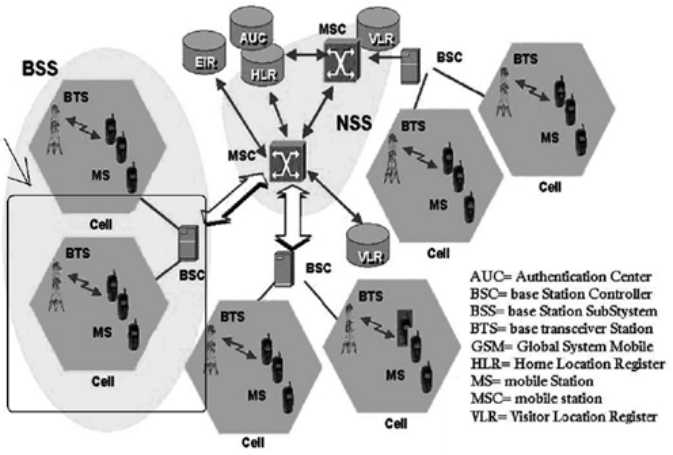


FIGURE-2: Elements of GSM System [4].

The purpose is to design the methodology for the radio interface of the GSM network which is comprised of mobile stations (MS), base transceiver stations (BTS) and the air interface which will be responsible for communication between them. To create these components the same GSM network is to be declared by modules. The GSM module is defined in a module called Compound Module, while its components: MS, BTS, AIR, and BSC are simple modules defined in Simple Module. Communication between these GSM network elements is done through a series of messages entering and leaving each module, through gates (Gates). For making a call, the messages have to pass through the Interface Radio (AIR). This module enables communication between the BTS and the Mobile Station.

Types of Messages:

Messages MS□BTS:

CONN_REQ	Connection request, checks for connection to BTS.
CHECK_LINE	Verification of the BTS, by the mobile.
CHECK_BTS	Availability of the BTS.
MOVE_CAR	Mobile Exchange.
MS_DATA	The BTS receives the data from the MS.
DISC_REQ	Disconnect Request.

Messages BTS□MS:

CONN_ACK	Connection received or accepted by the BTS.
CHECK_MS	MS Access.
FORCE_DISC	Disconnect.
BTS_DATA	The MS receives the data from the BTS.
DISC_ACK	Disconnect Request received.

This is done by processing messages to and from the module Air. This module receives the message that identifies the type of message it is, reviews and responds message, transfer the message from either the MS to the BTS or vice versa.

B. Analysis:

Input Variables:

Number of mobile stations; Number of base stations; Location of mobiles in the cell; Base station location in the cell; Speed mobile in the cell; Width of the cell; Length of cell; Power; Duration of the simulation; Average time of call; Simulation Time.

Output Variables:

Number of calls made by mobile phone; Number of calls answered; Number of missed calls; Number of dropped calls.

C. Formulation of Model:

It is a simulation of a cellular telephone system, GSM technology, for measuring minimum quality standards recommended for voice service as:

• Attempt to uncompleted calls: this should be less than 1% of the total of attempted calls.

• Dropped calls: This value must be less than 2% of total calls.

Tests will be done from different scenarios, and, accordingly, measurements will be obtained for these quality parameters [5].

D. Codification:

The process of codification can be done through Class Diagram and Modules. Moreover Class Diagram consists of Inheritance, Association, Mobile Station, BTS, GSMSIM, Air. On the other hand Module associates Interface GSMSIM, Air Interface, Mobile Workstation, Base Stations.

• CLASS DIAGRAM:

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

Inheritance:

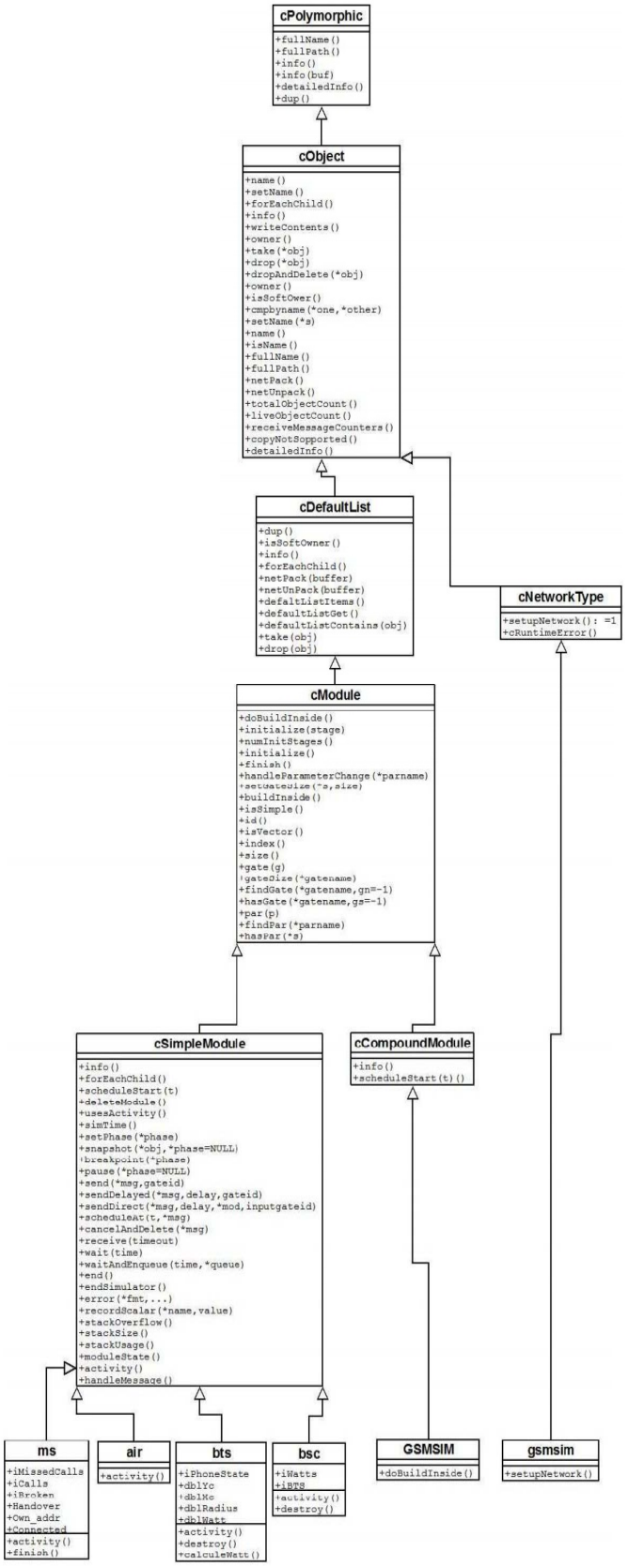


FIGURE-3: Diagram Class-Inheritance [5], [8].

Association:

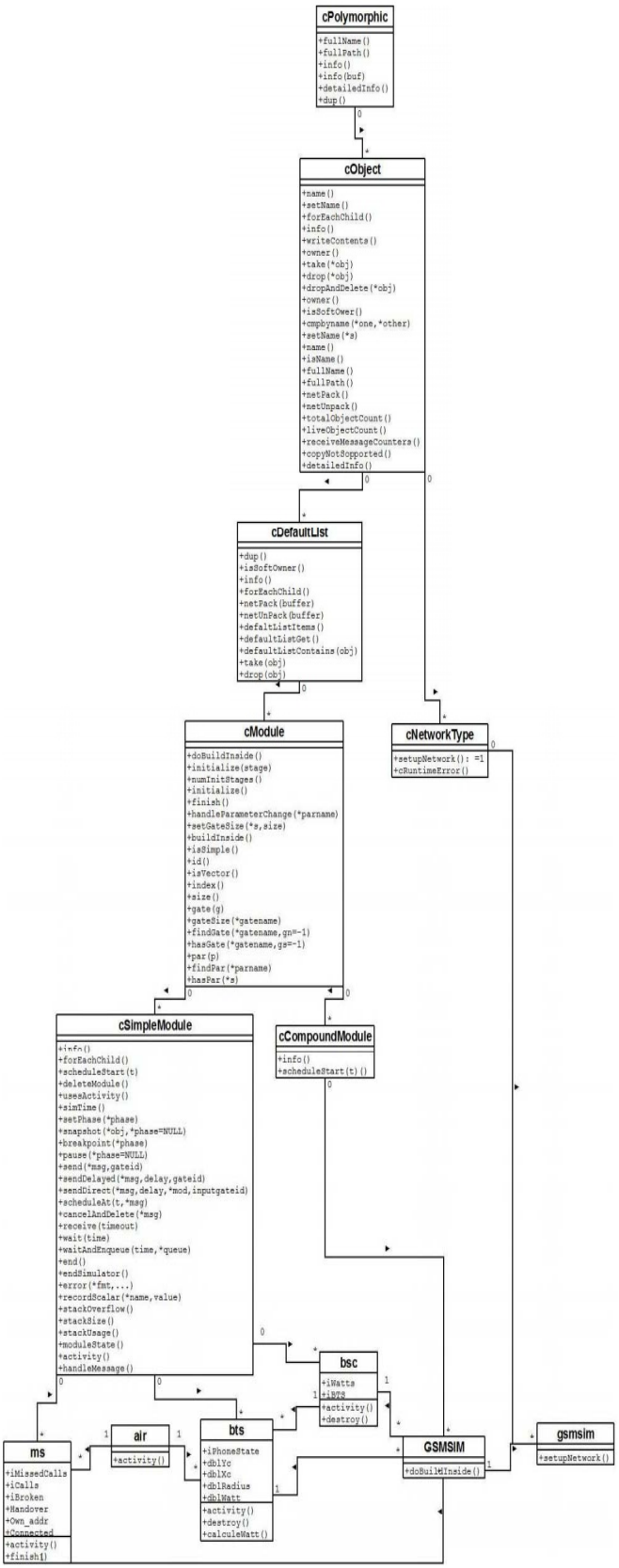


FIGURE-4: Diagram Class-Association [5], [8].

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

CLASS MS: Descend Simple Module class.

Attributes:

Missed Calls = integer and is used to keep the number of calls lost; Calls = integer and is used to store the number of actual calls; Broken = integer and is used to store the number of dropped calls; Own_addr = integer, indicates which interface (Air) is connected; Connected = integer, indicates that BTS are connected.

Methods:

Activity () = controls all activity of MS and is responsible for managing and processing the messages received and sent by the MS; Finish () = MS processing is complete and the file saved in the statistics actual number of calls, lost and fallen and calculates percentages.

CLASS BTS: Descend Simple Module class.

Attributes:

Phone State = integer, is a dynamic array that contains the state of each mobile connected to the BTS; Dbl Xc type = real, contains the X coordinate in the position of the BTS; Type = real dblYc contains the Y coordinate of the position of the BTS; Dbl Radius = real rate is the radius of the cell (in meters). An estimated multiplying the power by the multiplicity factor of it; Dbl Watt = real rate is the work of the BTS power is a parameter that the user enters.

Methods:

Activity () = controls all activity of the BTS, is responsible for managing and processing the messages received and sent by the BTS; Destroy () = the destroyer class and free the memory space of the array mobile connected to BTS; Calculate Watt () = is the function used to calculate the power of work of BTS, is based on the coordinates of the BTS and MS. When the MS is within the area BTS coverage this function returns a value greater than 0, otherwise returns a -1.

CLASS GSMSIM: Descend Compound Module class.

Methods:

Do Build Inside () = is responsible for configuring the modules in MS, BTS, BSC and AIR. Ask the user input data of each module, each module graph, and connects the modules through the Gates.

CLASS GSMSIM: Descend Network Type class.

This class implements the GSM module. Create an instance of the class and GSMSIM initializes the entire system through the method setup Network (); Classes communicate with one another via message passing using methods send () and receive () Simple Module class.

CLASS AIR: Descend Simple Module class.

Methods:

Activity () = controls all activities of messages between the BTS and MS are responsible for managing and processing between them.

• MODULES:

Interface GSMSIM:

Module Interface (GSMSIM)

/ / Parameters:

Parameter (number_ms, ParType_Numeric ParType_Const)

Parameter (number_bts, ParType_Numeric ParType_Const)

Parameter (xwidth, ParType_Numeric ParType_Const)

Parameter (ydepth, ParType_Numeric ParType_Const)

End Interface.

Air Interface:

Module Interface (Air)

/ / Gates:

Gate (from_ms [] GateDir_Input)

Gate (from_bts [] GateDir_Input)

Gate (to_ms [] GateDir_Output)

Gate (to_bts [] GateDir_Output)

End Interface.

Mobile Workstation:

Module Interface (BTS)

/ / Parameters:

Parameter (xc, ParType_Numeric)

Parameter (c, ParType_Numeric)

Parameter (slots, ParType_Numeric)

Parameter (watt, ParType_Numeric)

Parameter (phones, ParType_Numeric)

/ / Gates:

Gate (from_, GateDir_Input)

Gate (from_bsc, GateDir_Input)

Gate (to air, GateDir_Output)

Gate (to_bsc, GateDir_Output)

End Interface.

Base Stations:

Module Interface (MS)

/ / Parameters:

Parameter (xc, ParType_Numeric)

Parameter (c, ParType_Numeric)

Parameter (vx, ParType_Numeric)

Parameter (vy, ParType_Numeric)

Parameter (num bts, ParType_Numeric)

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

Parameter (timeout, ParType_Numeric)

/ / Gates:

Gate (from_, GateDir_Input)

Gate (to air, GateDir_Output)

End Interface.

IV. RESULT

Whether at work, university or just on the street, the use of GSM between people is increasing every day. Different systems and models of mobile telephony phones have created and developed to meet this great demand, so provide better and innovative services to users. For the design of the GSM network, specifically Radio Network should be taken consider the following factors: coverage, capacity and quality.

Coverage:

It is the area where each BTS must ensure the service call. This is determined from several factors, such as minimum area required for obtain license for operation, and traffic demand areas coverage to ensure quality service, meaning that there is continuity service.

Capacity:

This corresponds to the design of base stations required to enhance ability to undertake traffic in the areas of greatest demand and high concentration of users. A BTS, with an average capacity given by three sectors with 12 TRX by each sector, each TRX can use 8 timeslots, which are the channels that are available for voice traffic. That is if you have 2 TRX then be counted with 16 timeslots, which will make 16 calls at once. Importantly, the power of a BTS must be between 5 Watts and 32 Watts minimum and maximum value.

Quality:

For areas affected by the propagation phenomena or areas not covered, must be enhanced to signal level in the network. This project will be study the minimum quality standards in terms of missed calls and falls, which should provide in the GSM radio interface [6]. With this information, you create different scenarios, where they will face the minimum quality standards mentioned above. These settings are configured, which then of one number significant simulation, choosing and handle the configuration that comes closest to reality and ensuring service quality [7].

First Stage :

MS Number: 50	Number of missed calls: 25
BTS Number: 1	Number of dropped calls: 0
Dimension of the Cell: 1	Total calls: 310
Transmission power of BTS: 4 dBm	Percentage calls answered: 91.94%
Number of calls answered: 285	Percentage missed calls: 8.06%.

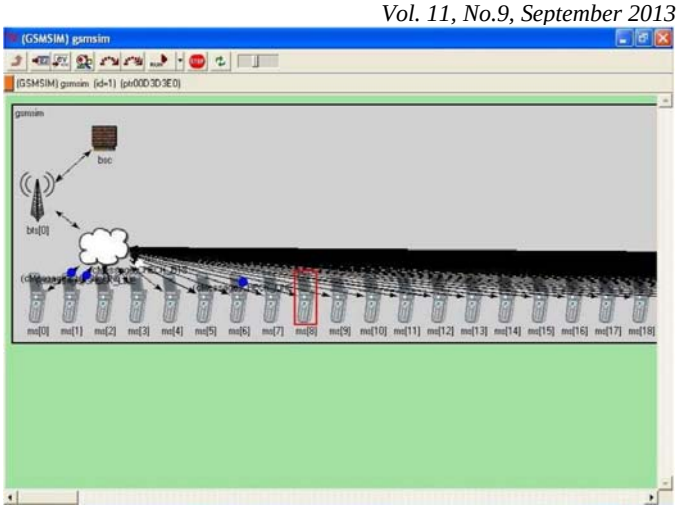


FIGURE-5: Simulation with 50 users

Second Stage :

MS Number: 50	Number of dropped calls: 10
BTS Number: 3	Total calls: 260
Dimension of the Cell: 4	Percentage calls answered: 80.77%
Transmission power of BTS: 7 dBm	Percentage missed calls: 15.38%.
Number of calls answered: 210	Percentage dropped calls: 2.69%.
Number of missed calls: 40	

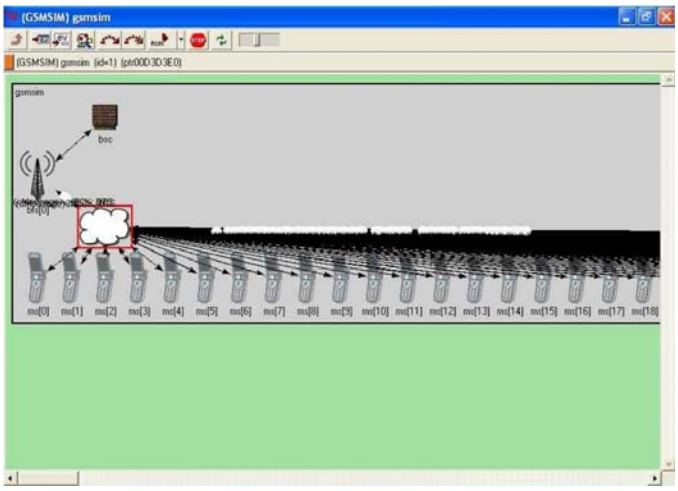
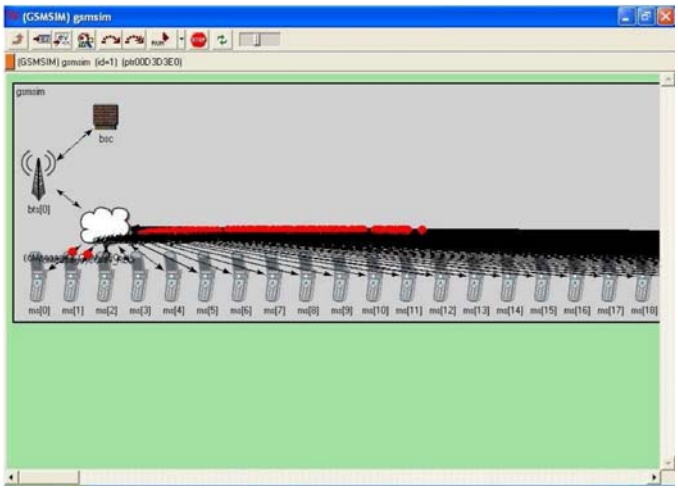


FIGURE-6: Simulation with 50 users

Third Stage :

MS Number: 85	Number of dropped calls: 12
BTS Number: 7	Total calls: 514
Dimension of the Cell: 9	Percentage calls answered: 91.44%
Transmission power of BTS: 7 dBm	Percentage missed calls: 6.23%
Number of calls answered: 470	Percentage dropped calls: 2.33%.
Number of missed calls: 32	

http://sites.google.com/site/ijcsis/
ISSN 1947-5500



Fourth Stage :

MS Number: 100	Number of dropped calls: 22
BTS Number: 20	Total calls: 842
Dimension of the Cell: 4	Percentage calls answered: 92.64%
Transmission power of BTS: 7 dBm	Percentage missed calls: 4.75%
Number of calls answered: 780	Percentage dropped calls: 2.61%.
Number of missed calls: 40	

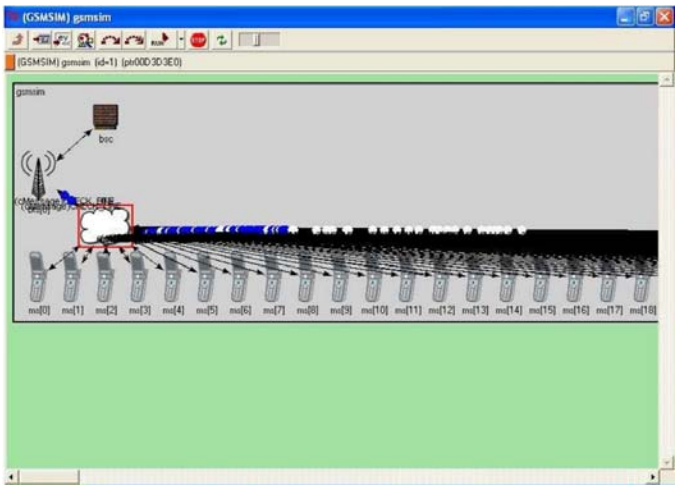


FIGURE-8: Simulation with 100 users

After creating different simulation environments, a series of Output Vector Data's are obtained instantly. From each simulation the percentage of calls made, served, lost and falls are also found.

REFERENCES

[1] Herrera, Jose Miguel. Technical University Federico Santa María (Valparaíso - Chile). NS2 -Network Simulator. (Cited April 28., 2007).

[2] Turegano, Javier Molina (mushrooms). NS-2 Simulator. linuxalbacete.org/web/content/view/149/31 /. (Accessed on January 14, 2013-20.15 GMT)

[3] The theoretical foundation of this issue was taken on the OMNeT home page: http: / / www.omnetpp.org, since it is the only source of information about this Tool. Therefore, the concepts mentioned in this theme is adapted from the posted there by André Varga, creator of the tool and the GSM module. (Accessed on February 25, 2013-20.45 GMT)

[4] Figure extracted from http://www.cisco.com. (Accessed on May 25, 2013- 23.50 GMT).

[5] Rico Martinez, Monica Andrea. Protocol Specification for Quality Management in Networks Telephony in Colombia. Bogotá, 2006. Master Thesis (Telecommunications Engineering). University St. Thomas Aquinas. School of Telecommunications.

[6] Robert, E. Shannon. "Systems Simulation". Trillas Mexico. 1997. P 419

[7] MOBILE PHONE. Technology Project and Administrative. Annex 3.6, http://www.subtel.cl/prontus_procesostarifarios/site/artic/20070205/asocfile/20070205104103/3_6_1a_anexo_pub_mem_proytec.pdf /.(Accessed on June 25, 2013-16.45 GMT).

[8] Mario César Romero Sánchez and Arturo Stanford Anichiarico, Specification for cellular system technology. UNIVERSIDAD EL BOSQUE. Faculty of Engineering Systems, Bogotá, November 15, 2008.

AUTHORS PROFILE

Syed Foysol Islam
MSc Engg in Electrical Engineering (BTH, Sweden)
BSc, MSc in Computer Science (Rajshahi University, Bangladesh)
Assistant Professor, Department of ETE
University of Development Alternative (UODA)

Fahmi Ahmed
MEngg in Telecommunication (AIUB, Bangladesh)
BSc in Electronic and Telecommunication Engg (UODA, Bangladesh)
Lecturer, Department of ETE
University of Development Alternative (UODA)

Web Usage Mining through Efficient Genetic Fuzzy C-Means

Deepak Kumar Niware
Research Scholar
Department of Computer Science & Engg.
TIT, Bhopal (INDIA)

Dr. Setu Kumar Chaturvedi
Professor
Department of Computer Science & Engg.
TIT, Bhopal (INDIA)

***Abstract*- In process of knowledge discovery from any web-log dataset, most widely and extensively used clustering algorithm for this purpose is Fuzzy c-means (FCM) algorithm because the data of web-log is unsupervised dataset. Due to sensitivity of FCM, it can be easily trapped in a local optimum, and it is also depends on initialization. In this paper we present use of Genetic algorithm in Fuzzy c-means algorithm to select initial center point for clustering in FCM. The purpose of this paper is to provide optimum initial solution for FCM with the help of genetic algorithm to reduce the error rate in pattern creation.**

Keywords: Fuzzy C-means, Genetic Algorithm, Web log mining, Web usage mining, Web mining.

I. INTRODUCTION

Database is used for keeping huge amount of data in a formatted manner, but data can also be in unformatted manner too, therefore it is suitable to apply data mining task for making intelligent business decisions. Web usage mining is a type of web mining which deals with the log files. It is also known as Web log mining. In application of web mining like Personalization, System Improvements, Modification of Web Site, Business Intelligence, Characterization of use etc. all can only be possible through web usage mining [6]. Clustering is one of the major data mining tasks and aims at grouping the data objects into meaningful classes (clusters) such that the similarity of objects within clusters is maximized, and the similarity of objects from different clusters is minimized [1]. Cluster can be viewed as subset of dataset, on the basis of these cluster, we can classify cluster technique as : Hard (Crisp) clustering methods are based on classical set theory, and require that an object either does or does not belong to a cluster. Hard clustering means partitioning the data into a specified number of mutually exclusive subsets. Fuzzy clustering methods, however, allow the objects to belong to several clusters simultaneously, with different degrees of membership. Objects on the boundaries between several classes are not forced to fully belong to one of the classes, but rather are assigned membership degrees between 0 and 1 indicating their partial membership. Fuzzy c-means clustering involves two processes: the calculation of cluster centers and the assignment of points to these centers using a form of Euclidian distance. This process is repeated until the cluster centers stabilize. The algorithm is similar to k-means clustering in many ways but it assigns a membership value to the data items for the clusters within a range of 0 to 1. So it incorporates fuzzy set's concepts of partial membership and forms overlapping clusters to support it [2]. A genetic algorithm (GA) is a search technique used in computing to find exact or approximate solutions to optimization and search problems. Genetic algorithms are a particular class of evolutionary algorithms (EA) that use techniques such as inheritance, mutation, selection, and crossover. In section 2 we shows some related work on Genetic algorithm and FCM, in section 3 we discuss the problem related with FCM, in section 4 overview of proposed method, in section 5 we present experiment setup and result, in last section 6 we shows the result and conclusion.

II. RELATED WORK

In [3] propose a novel hybrid genetic algorithm (GA) that finds a globally optimal partition of a given data into a specified number of clusters. They hybridize GA with a classical gradient descent algorithm used in clustering viz., K-means algorithm. Hence, the name genetic K-means algorithm (GKA). They define K-means operator, one-step of K-means algorithm, and use it in GKA as a search operator instead of crossover. They also define a biased mutation operator specific to clustering called distance-based-mutation. Using finite Markov chain theory, and prove that the GKA converges to the global optimum. It is observed in the simulations that GKA converges to the best known optimum corresponding to the given data in concurrence with the convergence result. It is also observed that GKA searches faster than some of the other evolutionary algorithms used for clustering.

In [1] present a clustering algorithm based on Genetic k-means paradigm that works well for data with mixed numeric and categorical features. They worked to modified description of cluster center to overcome the numeric data only limitation of Genetic k-mean algorithm and provide a better characterization of clusters.

Pareto-based multi objective evolutionary algorithm rule mining method based on genetic algorithms is in [5]. Predictive accuracy, comprehensibility and interestingness are used as different objectives of the association rule mining problem. Specific mechanisms for mutations and crossover operators together with elitism have been designed to extract interesting rules from a transaction database.

III. PROBLEM STATEMENT

The process of web usage mining model falls into four sections as source data collection phase, data pretreatment phase, pattern mining phase and pattern analysis phase is shown in Fig-1[7].

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

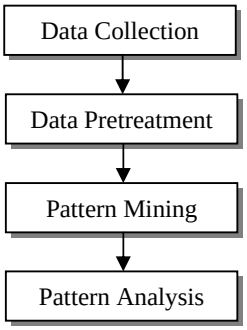


Figure -1

Pattern mining phase deals with making good clusters for the pattern analysis phase, each phase in web usage mining depend upon the previous phase for producing quality result. In this paper we are using Web log data which is huge and uncertain in nature. Due to the nature of web log data Fuzzy c-means algorithm which is inherited from k-means algorithm is used for clustering, because it is best suited for these types of data clustering. Pattern analysis is also depends on goodness of created cluster. In FCM, the cluster center which is chosen initially is not optimized solution. And pattern analysis is depended upon the cluster. The challenge is of better cluster center selection for the FCM. Because, if initial created cluster center is not optimized then rest cluster center will also not good. In this paper we proposed a Genetic Fuzzy c-mean algorithm, Genetic algorithm is used for the optimum solution for the cluster center in FCM.

IV. PROPOSED METHOD

In this paper we proposed to combine two method Genetic algorithms which is used to local optimum solution. And other is Fuzzy c-means algorithms used for clustering in unsupervised data for knowledge discovery.

A. Genetic Algorithm

A **genetic algorithm (GA)** is a search heuristic that mimics the process of natural evolution. This heuristic is routinely used to generate useful solutions to optimization and search problems. Genetic algorithms belong to the larger class of evolutionary algorithms (EA), which generate solutions to optimization problems using techniques inspired by natural evolution, such as Initialization, mutation, selection, and crossover [10].

a) Initialization stage

The search space of all possible solutions is mapped onto a set of finite strings. Each string (called chromosomes) has a corresponding point in the search space. The algorithm starts with the initial solutions that are selected from a set of configurations in the search space called population using randomly generated solutions or by applying special algorithms. Each of the initial solutions (called an initial population) is evaluated using a user defined fitness function. A fitness function exists to numerically encode the performance of the chromosome.

b) Selection stage

A set of individuals that have high scores in the fitness function is selected to reproduce itself. Such a selective process results in the best-performing chromosomes in the population to occupy an increasingly larger proportion of the population over time. From the selected set of individuals, some progeny is generated by applying different genetic operators (i.e. crossover, mutation).

c) Crossover stage

One site crossover and two site crossover are the most common ones adopted. In most crossover operators, two strings are picked from the mating pool at random and some portions of the strings are exchanged between the strings. Crossover operation is done at string level by randomly selecting two strings for crossover operations. A one site crossover operator is performed by randomly choosing a crossing site along the string and by exchanging all bits on the right side of the crossing site as shown in Fig. 2.

String1: 011|**01100**	String1: 011|*11001*
String2: 011|*11001*	String2: 011|**01100**
Before Crossover	After Crossover

Figure 2: One site crossover operation

String1: 011|**011**|00	String1: 011|*110*|00
String2: 011|*110*|01	String2: 011|**011**|01
Before Crossover	After Crossover

Figure 3: Two-site crossover operation

In one site crossover, a crossover site is selected randomly (shown as vertical lines). The portion right of the selected site of these two strings is exchanged to form a new pair of strings. The new strings are thus a combination of the old strings. Two site crossover is a variation of the one site crossover, except that two crossover sites are chosen and the bits between the sites are exchanged as shown in Fig. 3. One site crossover is more suitable when string length is small while two site crossover is suitable for large strings. The underlying objective of crossover is to exchange information between strings to get a string that is possibly better than the parents.

d) Mutation stage

Mutation operates on a single chromosome: one element is chosen at random from the chain of symbols, and the bit string representation is changed with another one [11].

e) Termination

The terminating condition of algorithm can be controlled by the convergence degree of solution, and the inheritance can be controlled by the evolution algebra.

B. Fuzzy c-means Clustering

Fuzzy C-Mean (*FCM*) is an unsupervised clustering algorithm that has been applied to wide range of problems involving feature analysis, clustering and classifier design. One of the widely used clustering methods is the Fuzzy c-means (FCM) algorithm developed by Bezdek [9]. Fuzzy c-means partitions set of n objects $o = \{o_1, o_2, .. , o_n\}$ in R^d

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

dimensional space into c (1 < c < n) fuzzy clusters with Z = {z_1, z_2, ... , z_c} cluster centers or centroids. The fuzzy clustering of objects is described by a fuzzy matrix μ with n rows and c columns in which n is the number of data objects and c is the number of clusters. μ_{ij}, the element in the ith row and jth column in μ, indicates the degree of association or membership function of the ith object with the jth cluster. The characters of μ are as follows [8]:

$$\mu_{ij} \in [0,1] \quad \forall i = 1,2,\dots,n;\ \forall j = 1,2,\dots,c \tag{1}$$

$$\sum_{j=1}^{c} \mu_{ij} = 1 \quad \forall i = 1,2,\dots,n \tag{2}$$

$$0 < \sum_{i=1}^{n} \mu_{ij} < n \quad \forall j = 1,2,\dots,c \tag{3}$$

The objective function of FCM algorithm is to minimize the Eq.(4):

$$J_m = \sum_{j=1}^{c}\sum_{i=1}^{n} \mu_{ij}^{m} d_{ij} \tag{4}$$

Where

$$d_{ij} = \left|\left|o_i - z_j\right|\right| \tag{5}$$

in which, m (m > 1) is a scalar termed the weighting exponent and controls the fuzziness of the resulting clusters and d_{ij} is the Euclidian distance from object o_i to the cluster center z_j. The z_j, centroid of the jth cluster, is obtained using Eq. (6).

$$z_j = \frac{\sum_{i=1}^{n} \mu_{ij}^{m} o_i}{\sum_{i=1}^{n} \mu_{ij}^{m}} \tag{6}$$

The FCM algorithm is iterative and can be stated as follows:

Algorithm: *Fuzzy c-means*

Step 1. Select m (m > 1); initialize the membership function values μ_{ij}, i = 1, 2, ... , n; j = 1, 2, . . . , c.

Step 2. Compute the cluster centers z_j, j = 1, 2, ... , c, according to Eq. (6).

Step 3. Compute Euclidian distance d_{ij}, i = 1, 2,..... , n; j = 1, 2, ... , c.

Step 4. Update the membership function μ_{ij}, i = 1, 2, .. , n; j = 1, 2, ... , c according to Eq. (7).

$$\mu_{ij} = \frac{1}{\sum_{k=1}^{c}\left(\frac{d_{ij}}{d_{ik}}\right)^{\frac{2}{m-1}}} \tag{7}$$

Step 5. If not converged, go to step 2.

Several stopping rules can be used. One is to terminate the algorithm when the relative change in the centroid values becomes small or when the objective function, Eq. (4), cannot be minimized more. The FCM algorithm is sensitive to initial values and it is likely to fall into local optima.

V. Experiment and Result

Web log dataset used in this experiment is Microsoft Server log file, having 22 attributes in it. We have tested proposed method on two web log dataset having 259 KB and 559 KB size. We experimented on Pentium Dual Core 1.80 GHz and 1GB RAM with 160 GB HDD machine having Window XP Service Pack 3 with MATLAB Version 7.8. Result table and graphs are as follows.

TABLE-I

Method	Threshold	Error Rate	Time	Iteration
FCM	0.1	50.391234	59.191740	2000
FCM	0.2	51.036754	62.939058	1000
FCM	0.3	51.682274	62.104490	667
FCM	0.4	52.327794	62.372756	500
FCM	0.5	52.973314	62.084189	400
FCM	0.6	53.618834	62.155665	333
FCM	0.7	54.264354	61.654710	286
FCM	0.8	54.909874	60.510172	250
FCM	0.9	55.555394	60.662062	222
GFCM	0.1	19.317537	61.104803	2001
GFCM	0.2	19.539618	63.751756	1001
GFCM	0.3	19.761700	64.026603	668
GFCM	0.4	19.983781	63.583065	501
GFCM	0.5	20.205862	64.041465	401
GFCM	0.6	20.427943	63.856052	334
GFCM	0.7	20.650024	62.092531	287
GFCM	0.8	20.872105	63.207093	251
GFCM	0.9	21.094186	62.714621	223

Table 1: Analysis Report of FCM and GFCM with weblog1 dataset

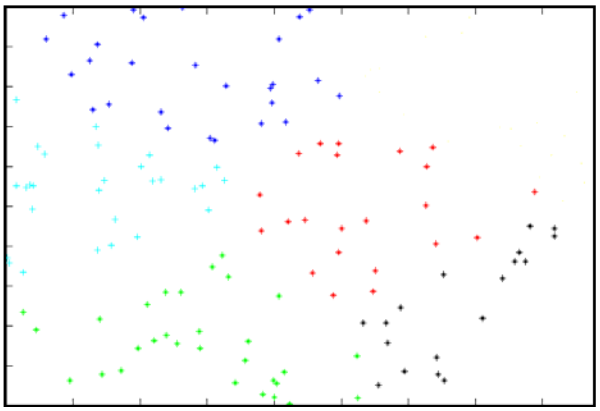


Figure 4: FCM Method of Dataset 1

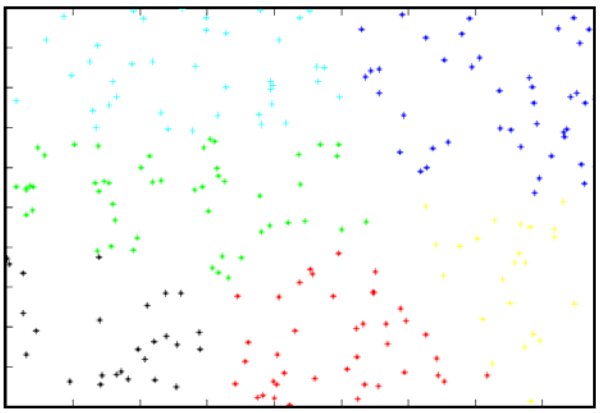
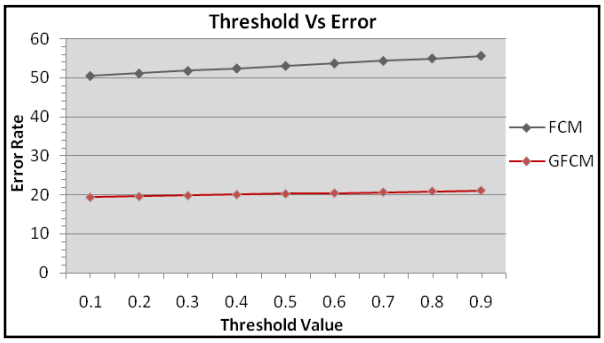
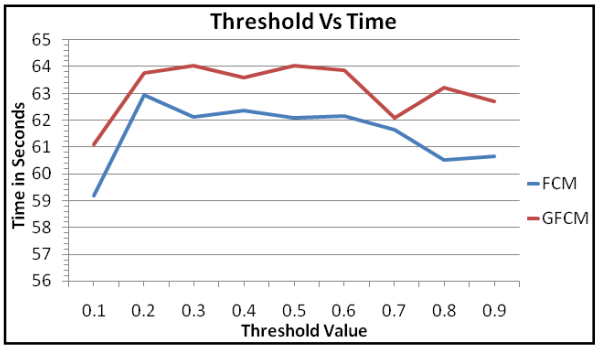


Figure 5: GFCM Method of Dataset 1

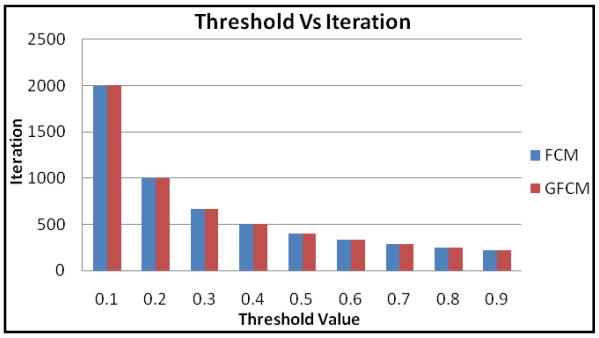
http://sites.google.com/site/ijcsis/
ISSN 1947-5500



Graph 1.1: Threshold Vs Error of Dataset 1.



Graph 1.2: Threshold Vs Time of Dataset 1.



Graph 1.3: Threshold Vs Iteration of Dataset 1.

VI. CONCLUSTION

In the above shown graphs, in the FCM method as the value of threshold value increases the error rate also increases. In proposed GFCM method also error rate increased as the threshold value increases. But it is clearly shown in graph, if we comparing both methods that at same value of threshold value GFCM reduce the error rate more than 50% of the error rate of FCM. In graph of FCM, increases rate of error rate is high but in graph of GFCM, the increase rate of error rate is much low as compared to FCM. It shows that data loss in GFCM is low and the content of cluster is increased which is important parameter of evaluation. The other parameter of evaluation like time it is also shown in graph that time is varying because time is dependent upon CPU time other processes and time rate of increases and decrease is also not depend on threshold value. And in iteration parameter it is approximately same in both methods. Graphs, experiment and analysis concludes that GFCM is more efficient method for pattern recognition and cluster creation in web usage mining.

REFERENCES

[1] Dharmendra K Roy , Lokesh K Sharma; "Genetic k-means clustering algorithm for mixed numeric and categorial datasets"; In proceeding of "International Journal of Artificial Intelligence & Applications (IJAIA)" Vol.1,No.2, page 23-28,2010.

[2] Raju G, Binu Thomas, Sonam Tobgay , Th. Shanta Kumar; "Fuzzy Clustering Methods in Data Mining-A comparative Case Analysis"; In "International Conference on Advanced Computer Theory and Engineering", IEEE, page: 489-493, 2008.

[3] K. Krishna, M. Narasimha Murty; "Genetic K-Means Algorithm"; In "IEEE Transactions on systems, man, and cybernetics-part b: cybernetics" Vol. 29, No. 3, page: 433-439, JUNE 1999.

[4] Dharmendra K Roy, Lokesh K Sharma;"Genetic k-Means clustering algorithm for mixed numeric and categorical datasets"; In "International Journal of Artificial Intelligence & Applications"; Vol.1, No.2, page: 23-28, April 2010.

[5] Peter P. Wakabi-Waiswa, Venansius Baryamureeba; "Extraction of interesting association rules using genetic algorithms"; In "International Journal of Computing and ICT Research", Vol. 2 No. 1, Page: 26-33, June 2008.

[6] "Jaideep Srivastava, Robert Cooley, Mukund Deshpande, Pang-Ning Tan"; "SIGKDD Explorations 2000" ACM SIGKDD, Volume 1, Issue 2 - page 12-23, Jan 2000.

[7] Ya-Xiu Yu, Xin-Wei Wang; "Web Usage Mining Based on Fuzzy Clustering"; In "International Forum on Information Technology and Applications"; IEEE Computer Society, page: 268-271, 2009.

[8] H. Izakian, A. Abraham;"Fuzzy C-means and fuzzy swarm for fuzzy clustering problem"; In "Expert Systems with Applications 38(2011)"; Elsevier, page: 1835-1838, 2011.

[9] J. Bezdek, "Fuzzy Mathematics in Pattern Classification," Ithaca: Cornell University, 1973.

[10] http://en.wikipedia.org/wiki/Genetic_algorithm.

[11] Mu-Jung Huang, Hwa-Shan Huang, Mu-Yen Chen; "Constructing a personalized e-learning system based on genetic algorithm and case-based reasoning approach"; In "Expert Systems with Applications (2006)" Elsevier, 2006.

[12] T. Velmurugan, T. Santhanam; "Performance Evaluation of K-Means and Fuzzy C-Means Clustering Algorithms for Statistical Distributions of Input Data Points"; In "European Journal of Scientific Research" Vol.46 No.3, page: 320-330, 2010.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

Clustering Technique Based Outlier Detection Method for Document Classification

Deepti Jain
SATI Vidisha,
Vidisha (M.P.) India
.

Dr. Bhupendra Verma
Director TIT Bhopal,
Bhopal (M.P.) India
.

Dr. R. C. Jain
Director SATI Vidisha,
Vidisha (M.P.) India
.

ABSTRACT

In this paper, we propose a clustering based techniques to capture outliers for document classification and apply K-means clustering algorithm to divide the dataset into clusters. The points lying near centroid of the cluster are not probable candidate for outlier and prune out such points from each cluster then calculate a distance based outlier score for remaining points. The computations calculate to the outlier score reduces considerably due to the pruning of some points. Based on the outlier score declare the top n points with the highest score as outliers after that classification technique is applied for categorization. The experimental results using actual dataset demonstrate that even though the number of computations is fewer, the proposed method performs better than the obtainable method

Keywords: **Outlier; Cluster; Distance-based; Classification**.

1. INTRODUCTION

Outlier is a data point that does not conform to the normal points characterizing the data set. Detecting outliers has important applications in data cleaning as well as in the mining of abnormal points for fraud detection, intrusion detection, marketing, network sensors, email spam detection, stock market analysis. Finding anomalous points among the data points is the basic idea to find out an outlier. Outlier detection signals out the objects mostly deviating from a given data set [1, 2, 3, 4, 5, 6]. The problem of detecting outliers has been extensively studied in the statistics community [7, 8, 6]. Typically, the user has to model the data points using a statistical distribution, and points are determined to outliers depending on how they appear in relation to the postulated model. The main problem with these techniques is that in many situations, user might not have the enough knowledge about the underlying data distribution.

In particular, distance-based techniques use the distance function for relating each pair of objects of the data set. Distance-based definitions [4, 9, 10] represent a useful tool for data analysis [3, 11, 12]. These definitions are computationally efficient, since distance-based outlier scores are monotonic non-increasing functions of the portions of the database already explored. In recent years, several algorithms have been proposed to fast detecting distance based outliers [13, 14, 9, 15]. Some of them are every efficient in terms of CPU cost, while some other are mainly interested in the I/O cost.

Several measures are used to find out the deviation of a point from other points which tells the outlierness of a point. As we know that the number of outliers in a dataset is very few, it is redundant to calculate these measures for all points. By removing the points which are probably not outliers, we can reduce the computation time.

In this work, we identify the points which are not outliers using clustering and distance functions, and prune out those points. Next, we calculate a distance-based measure for all remaining points, which is used as a parameter to identify a point to be an outlier or not. We assume that there are n outliers in data set, and top n points will be reported as outliers by our method. In our work, we have used Local Distance-based Outlier Factor (LDOF) [16, 14, 17] as a measure to identify an outlier.

2. RELATED WORK

Knorr and Ng [4, 18] were the first to introduce distance-based outlier detection techniques. An object p in a data set DS is a DB(q,dist)-outlier if at least fraction q of the objects in DS lie at a greater distance than dist from p. This definition is well accepted, since it generalizes several statistical outlier test.

Ramaswamy et.al. [9, 19] proposed the extension of the above definition. All the points are ranked based on the outlier score. Given two integers kn and w, an object p is said to be an outlier, if less than w objects have higher value for Dk than p, where Dk denotes the distance of the kth nearest neighbor of the object p.

Subsequently, Angiulli and Pizzuti [7, 16] proposed a method to determine the outliers by considering the whole neighborhood of the objects. All the points are ranked basing on the sum of the distances from the k-nearest neighbors, rather than considering solely the distance to the kth nearest neighbor. The above three definitions are closely related.

Breunig et.al. [2] proposed a Local Outlier Factor(LOF) for each object in the data set, indicating its degree of outlierness. This is the first concept of an outlier which also quantifies how outlying an object is. The outlier factor is a local

in the sense that only a restricted neighborhood of each object is taken into account. Since the LOF value of an object is obtained by comparing its density with those in its neighborhood, it has strong modeling capability than a distance based scheme, which is based only on the density of the object itself. Note that the density based scheme does not explicitly categorize the objects into either outliers or non- outliers (If desired, a user can do so by choosing a threshold value to separate the LOF values of the two classes) [20].

Zhang et.al. [17] proposed a local distance-based outlier detection method to find outliers from the data set. The local distance-based outlier factor (LDOF) of an object determines the degree to which the object deviates from its neighborhood. Calculating LDOF for all points in the data set, makes overall complexity O(N 2), where N is the number of points in the data set [21].

Clustering methods like CLARANS [22], DBSCAN [15,23], BIRCH [6, 24] and CURE [25, 26] may detect outliers. However, since the main objective of a clustering method is to find clusters, they are developed to optimize clustering and not to optimize outlier detection. The definitions of outlier used are subjective to the clusters that are detected by these algorithms. While definitions of distance-based outliers are more objective and independent of how clusters in the input data are identified [27].

While obtainable work on outliers focuses only on the identification aspect, the work in [11] also attempts to provide intentional knowledge, which is basically an explanation of why an identified outlier is exceptional.

3. Local Distance-Based Outlier Factor

LDOF (Local Distance-based Outlier Factor) has used in this work, which tells how much a point is deviating from its neighbors. The high ldof value of a point indicates that the point is deviating more from its neighbors and probably it may be an outlier. The factor ldof is calculated as follows [17]:

$ldof$ of p: The local distance-based outlier factor of p is defined as:

$$Idof(p) := \frac{\overline{d_p}}{\overline{D_p}}$$

$\bar{d}$ (KNN distance of p): Let Npbe the set of k-nearest contains less number of points than the required num neighbors of object p (excluding p). The k-nearest neighbors distance of p equals the average distance from p to all objects in Np . More formally, let dist(p, q) ≥ 0 be a distance measure between objects p and q. The k-nearest neighbors distance of object p is defined as:

$$\bar{\imath}_p := \frac{1}{k} \sum_{q \in N_p} dist(p,q)$$

$\bar{D}p$ (KNN inner distance of p): Given Np of object p, the k-nearest neighbors inner distance of p is defined as the average distance among objects in Np

$$\bar{D}_p := \frac{1}{k(k-1)} \sum_{q.q \in N_p.q = q} dist(q.q)$$

Calculating ldof values for all points is computationally expensive, since the complexity of this algorithm is O(N 2) [17]. In our next section, the reduction based computation approach is proposed for detecting the outliers by pruning some points which are probably not the outliers.

4. Proposed Work

In this, the proposed algorithm, this is an improvement over LDOF (Local distance based outlier factor). The main shortcoming is computationally expensive with the LDOF algorithm. This is because for each point p in the data set DS , one has to compute ldof . Since we are interested in the only outliers which are very few in numbers, the ldof computations for all the points are of little use and can be altogether avoided. We use K-means algorithm to cluster the data set. Once clusters are formed, we calculate radius of each cluster. Prune the points whose distance from the centroid is less than the radius of the respective clusters. After that for each unpruned points in every cluster we calculate the ldof. We report the top-n points with high ldof value as outliers.

A.Outline

According to LDOF, The proposed idea has derived in the pruning-based algorithm is to first cluster the data set into clusters, and then prune the points in different clusters if determined that they cannot be outliers. Since n (number of outliers) will typically be very small, this additional preprocessing step helps to eliminate a significant number of points which are not outliers. The **Algorithm1** describes our method to find out outliers. We briefly describe the steps need to be performed by our pruning based algorithm.

1) **Generating clusters**: Initially, we cluster the entire dataset into c clusters using K-means clustering algorithm and calculate radius of each cluster.

2) **Clusters having less number of points**: If a cluster contains less number of points than the required number of outliers, the radius pruning is avoided for that cluster.

3) **Pruning points inside each cluster:** Calculate distance of each point of a cluster from the centroid of the cluster. If the distance of a point is less than the radius of a cluster, the point is pruned.

4) **Computing outlier points:** Calculate ldof for all the points that are left unpruned in all the clusters. Then n points with high ldof values are reported as outliers.

The complexity of K-means algorithm is c * it* N , here c is the number of clusters to be formed, it is the number of iterations and N is the number of data points. The total computation of our method is c * it* N + c* np + (w * N)^2

Algorithm 1 Outlier Detection Algorithm

Input: DS : DataSet, c: required number of clusters, it: number of iterations, n: number of outliers

```
1:  Set Y ← Kmeans(c, it, DS)
2:  for each cluster C_j ∈ Y do
3:      Radius_j ← radius(C_j)
4:  end for
5:  if |C_j| > n then
6:      for each point p_i ∈ C_j do
7:          if distance(p_i, o_j) < Radius_j then
8:              prune(p_i)
9:          else
10:             Add p_i to U
11:         end if
12:     end for
13: else
14:     for each point p_i ∈ C_j do
15:         Add p_i to U
16:     end for
17: end if
18: for each point p_i ∈ U do
19:     calculate ldof(p_i)
20: end for
21: Sort the points according to their ldof(p_i) values.
22: First n points with highest ldof(p_i) values are
    the desired outliers
```

Where np represents average number of points in each cluster and w indicates the fraction of data point that we have after pruning, which is around 0.4. Number of clusters c depends on the number of outliers which is very small compare to N. Since c and it are small, so the total computation of our method is less than N^2.

In the next section experimental analysis has proposed which presents the comparative analysis between LDOF and PLDOF method.

5. Experimental Setup

(i) Dataset : Two commonly used datasets, 20-Newsgroups, and Reuters-21578 WebPages were used in our experiments [23]. For each document, we extract features from its title and body separately for Co-Training algorithm, and extract single feature vector from both title and body for all algorithms. Stop words are eliminated and stemming is performed for all features. For body features, all words with low document frequency (set to less than three in the experiments) are removed. TFIDF [13] is then used to index both titles and bodies, where IDF is calculated based on the training dataset. The words that appear only in the test dataset but not in the train dataset are discarded.

From the 20-Newsgroups dataset2 only select the five comp.* conversation groups, which forms a very confusing but evenly distributed dataset for classification, i.e. there are approximately 1000 articles in each group. We choose 80% of each group as training set and the remaining 20% as test set, which give us 4000 train examples and 1000 test examples. The dataset Reuters-21578 is downloaded from Yiming Yang's homepage3. We use the Moderate split to form the training set and test dataset, where there are 7769 training examples and 3019 test examples.

(ii) Experiment Results: We used three evaluation measures (Recall, Precision, and F1) as the bases of comparison, where F1 is computed based on the following equation:

$$F_1 = \frac{2 \times recall \times precision}{recall + precision}$$

Precision and recall are widely used for evaluation measures in IR and ML, where according to below matrix:

Precision=a/(a+b)

Recall=a/(a+c)

Iteration	Relevant	Irrelevant
Documents Retrieved	**A**	**B**
Documents not Retrieved	**C**	**D**

(iii) Result Analysis: In this section, we compare the outlier detection performance of our PLDOF method with the LDOF outlier detection existing method and their classification performance evaluation. To further validate our approach, we repeat the experiment 10 times with a different number of outliers (randomly extracted from objects). Each time, we perform 10 independent runs, and calculate the average detection precision over the k range from 10 to 50.

These results are implemented through Vb.Net 2003 based source code and graphs has designed by Microsoft Excel 2003 Which are given below analysis. The Figures have designed between k range and Precision of LDOF and PLDOF method.

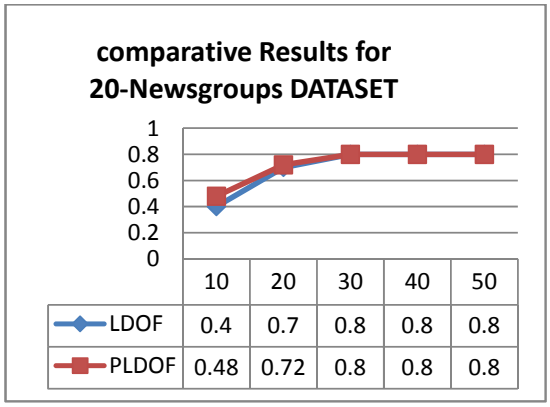
The Figure 1 observed that how precision is varying with top n points and neighborhood size k. As shown in the Figure 1, the precision of our method is better than the LDOF method [17]. When we consider top-20 points, both the methods detect all the outliers. We have also conducted an experiment by varying the neighborhood size k. We have observed that both the methods are at per and precision reaches its upper limit at k=30. From Table I we observed that even though we prune out around 57% of points from the data set we are able to get precision at par with the precision of LDOF. Due to the

reduction of around 57% of points, the computation cost is reduced drastically, which is a positive aspect of pruning based LDOF method in table I.

Table I PRUNING RATIO FOR 20-Newsgroups DATASET

	Percentage of	Precision	
	data points pruned	LDOF	PLDOF
10	57.18	0.4	0.48
20	55.52	0.7	0.72
30	55.52	0.8	0.8
40	53.59	0.8	0.8
50	54.14	0.8	0.8

Figure 1 Comparison of precision of LDOF and PLDOF method 20-Newsgroups DATASET

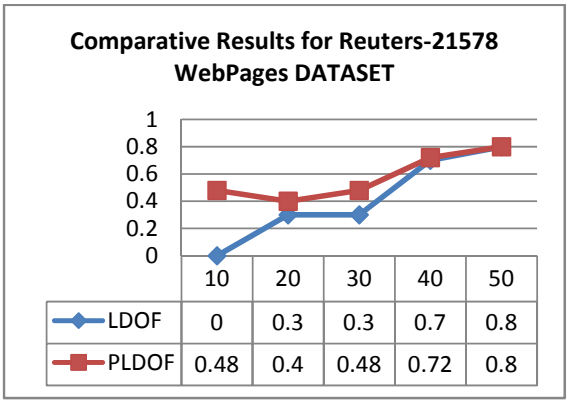


In this next experiment, we use a dataset which has been used to find localization site of protein. The data set contains 1484 records (objects), each with attributes (1 name + 8 real-valued input features). In this experiment we use all 1474 records as normal objects and added 10 records into normal objects as outliers. We perform similar type of experiment that we have conducted for 20-Newsgroups Dataset and observed similar trends. The experiment results are presented in Figure 2 and Table II. It is observed that we could able to prune out more than 60% of points from original data set without losing any outlier detection performance.

Table II PRUNING RATIO FOR Reuters-21578 WebPages DATASET

K	Percentage of	Precision	
	data points pruned	LDOF	PLDOF
10	59.03	0	0.48
20	59.57	0.3	0.4
30	59.03	0.3	0.48
40	60.24	0.7	0.72
50	60.10	0.8	0.8

Figure 2 Comparison of precision of LDOF and PLDOF method for Reuters-21578 WebPages DATASET



6. CONCLUSION

In this paper, we proposed an efficient outlier detection method. We first identify some points which are not probable candidates for outliers by using the radius of each cluster and we remove those points from the dataset. Due to the reduction in the size of the data set, the computation time reduced considerably. We used a local distance-based outlier factor to measure the degree to which an object deviates from its neighborhood. The precision of detecting outliers of our method is at per or higher than the existing methods though we pruned out some of the points. Experimental results show classification the web pages with acceptable accuracy. Only extremely minimal degradation in the performance, but the memory and time requirement very less [28]. For the project category, it is interpreted from the results that even a single good quality summary is enough to attain acceptable classification accuracy.

One limitation of our algorithm is that with the constant arrival of new emails, the same procedure of clustering, meta-feature addition, and classification, should be applied again for the whole dataset, a rather time consuming, and computationally expensive process. A suggestion would be to use proposed clustering instead of the static clustering algorithm used now. The proposed clustering is a method that deals with the problem of updating clusters without frequently performing complete re-clustering. This would be a more suitable way for maintaining clusters in the typical, dynamic environment of spam filtering. Another issue about our algorithm is its rather naive approach to clustering that may not capture all the meta-information possible hidden in the dataset.

More sophisticated clustering methods have been proposed in the literature that focus on incorporating prior knowledge into the clustering process; conceptual clustering, topic-driven

clustering, just to name a few. These methods are based in the idea that it is possible to use explicitly available domain knowledge to constrain or guide the clustering process. In our case, the class labels of the training set can constitute the domain knowledge and be used as guidance to a clustering algorithm. Another issue that needs to be discussed is the representation of the extra knowledge derived from clustering, i.e. the representation of the clusters.

7. REFERENCES

[1] A. Arning, R. Agrawal and P. Raghavan, A linear method for deviation detection in large databases, Pages 164–169, 1996.

[2] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander. Lof, identifying density-based local outliers. *SIGMOD Rec Page* 93–104, 2000.

[3] W. Jin, A. K. H. Tung, and J. Han. Mining top-n local outliers in large databases. In *KDD 01, Proceedings of the seventh ACM SIGKDD international conference on Knowledge dis- covery and data mining*, pages 293–298, 2001.

[4] E. M. Knorr and R. T. Ng. Algorithms for mining distance- based outliers in large datasets. In *Proc. 24th Int. Conf. Very Large Data Bases, VLDB*, pages 392–403, 1998.

[5] S. Papadimitriou, H. Kitagawa, P. Gibbons, and C. Faloutsos. Loci: fast outlier detection using the local correlation integral. In *Data Engineering, 2003. Proceedings. 19th International Conference on Data Engineering*, pages 315–326, 2003.

[6] B. V and L. T. *Outliers in Statistical Data*. John Wiley and Sons, New York, 1994

[7] F. Angiulli, S. Basta, and C. Pizzuti. Distance-based detection and prediction of outliers. *IEEE Transactions on Knowledge and Data Engineering*, Pages 145–160, 2006.

[8] F. Angiulli and C. Pizzuti. Fast outlier detection in high dimensional spaces. In *PKDD '02, Proceedings of the 6th European Conference on Principles of Data Mining and Knowledge Discovery*, pages 15–26, 2002.

[9] S. Ramaswamy, R. Rastogi, and K. Shim. Efficient algorithms for mining outliers from large data sets. pages 427–438, 2000.

[10] Guru, D.S., Harish, B.S. and Manjunath, S. 2010.Symbolic representation of text documents. In Proceedings of Third Annual ACM Bangalore Conference.

[11] E. M. Knorr and R. T. Ng. Finding intensional knowledge of distance-based outliers. In *VLDB '99: Proceedings of the 25th International Conference on Very Large Data Bases*, pages 211–222, 1999.

[12] John, G.H., Kohavi, R., Pfleger, K. 1994. Irrelevant features and the subset selection problem. In: Proceedings of International Conference on Machine Learning. 121-129.

[13] A. Ghoting, S. Parthasarathy, and M. Otey. Fast mining of distance-based outliers in high-dimensional datasets. *Data Mining and Knowledge Discovery*, Page 349–364, 2008.

[14] E. M. Knorr, R. T. Ng, and V. Tucakov. Distance-based outliers: algorithms and applications. *Pages* 237–253, 2000.

[15] Y. Tao, X. Xiao, and S. Zhou. Mining distance-based outliers from large databases in any metric space. *KDD 06*, 20, 2006.

[16] F. Angiulli and C. Pizzuti. Outlier mining in large high- dimensional data sets. *IEEE Transactions on Knowledge and Data Engineering*, Page 203–215, 2005.

[17] K. Zhang, M. Hutter, and H. Jin. A new local distance-based outlier detection approach for scattered real-world data. In *PAKDD '09: Proceedings of the 13th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining*, pages 813–822, 2009.

[18] Tang, B., Shepherd, M., Heywood, M.I. and Luo, H.2005. Comparing dimension reduction techniques for document clustering. In Lecture Notes in Computer Science on Advances in Artificial Intelligence. vol 3501, 292-296.

[19] Isa, D., Lee, L.H., Kallimani, V.P., Rajkumar, R. 2008. Text document preprocessing with the bayes formula for classification using the support vector machine. IEEE Transactions on Knowledge and Data Engineering, vol. 20 23-31.

[20] Elnahrawy, E.M. 2002. Log-based chat room monitoring using text categorization: A comparative study. In International Conference on Information and Knowledge Sharing, Acta Press Series, 111-115.

[21] Dinesh, R., Harish, B.S., Guru, D.S. and Manjunath, S.2009. Concept of status matrix in classification of text documents. In Proceedings of Indian International Conference on Artificial Intelligence, India, 2071 – 2079.

[22] R. T. Ng and J. Han. Efficient and effective clustering methods for spatial data mining. pages 144–155, 1994.

[23] M. Ester, H.P. Kriegel, and X. Xu. A database interface for clustering in large spatial databases, In *Proceedings of 1st International Conference on Knowledge Discovery and Data Mining (KDD-95)*, Page 45-56, 1995.

[24] T. Zhang, R. Ramakrishnan, and M. Livny. Birch: an efficient data clustering method for very large databases Pages 103–114, 1996.

[25] S. Guha, R. Rastogi, and K. Shim. CURE, An efficient clustering algorithm for large databases. *SIGMOD Rec.*, Page 73–84, 1998.

[26] Bock, H.H., Diday, E. 1999. Analysis of symbolic Data. Springer Verlag.

[27] Bezdek, J.C.1981. Pattern Recognition with Fuzzy Objective Algorithms. Kluwer Academic Publisher.

[28] Guru, D.S. and Kiranagi, B. B. 2005. Multivalued type dissimilarity measure and concept of mutual dissimilarity value for clustering symbolic patterns. Journal of Pattern Recognition, vol. 38(1), 151-156.

Author's profile

Deepti Jain has received her B.Tech in Computer Science and Engineering from Barkatullah University, Bhopal and MTech in computer science and engineering from Rajiv Gandhi Technical university Bhopal (M.P.),India. She is persuing Ph.D in computer science and Engineering from R.G.P.V. Bhopal(M.P.),India under the guidance of Dr Bhupendra verma and Dr RC jain .Her field of research is Data Mining.

Dr. Bhupendra Verma is the Director of Technocrafts Institute of Technology, Bhopal (M.P.), India. He has done B.E, M.Tech and Ph.D in Computer Science and Engineering from Rajiv Gandhi Prodyogiki Vishwavidyalaya, Bhopal (M.P.) India.He has 20 years of teaching experience. At TIT ,,Bhopal he has been involved in various Academic, Technical and industrial collaborative activities for the students, and also involved intensively in research activities at UG and PG level. Dr. Verma has guided about 25 M. Tech dissertations and presently 6 students are perusing Ph. D under his guidance at RGPV Bhopal. He has published 70 research papers in journals and conferences. He has delivered expert lectures on various topics in computer engineering at various institutes. Presently he is Director of Technocrats Institute of Technology (Excellence) Bhopal.

Dr. R. C. Jain is the Director of Samrat Ashok Technological Institute, Vidisha, (M.P.) India. He has 39 years of teaching/research experience .He is actively engaged in research .He is editor -in-chief of SATI Journal of science and Technology Vidisha and member of editoral board and Reviewer of various Journals.He has published approx 200 research papers in national,international journals and conferences .He has produce 20 Ph.Ds in various stream of Engineering and technology and 10 scholars are working for their Ph.Ds under him.His field of Specialization are Fuzzy Systems ,DIP,Mobile Computing,Adhoc Networks,Data Mining and Data Warehousing.

Hybrid Storage Architecture: A survey

Tejaswini Apte
Symbiosis Institute of Computer Studies and Research
Pune, INDIA

Dr. Maya Ingle and Dr. A.K. Goyal
Devi Ahilya Vishwavidyalaya, Indore
Indore, INDIA

Abstract-Database design requirement for large scale OLAP applications differs from small-scale database programs. Database query and update performance is highly dependent on the storage design techniques. Two storage design techniques have been proposed in the literature namely; a) Row-Store architecture and b) Column-Store architecture. This paper studies and combines the best aspect of both Row-Store and Column-Store architectures to better serve an ad-hoc query workload. The performance is evaluated against TPC-H workload.

General Terms: Performance, Design

Keywords: Statistics

I. INTRODUCTION

Database management systems (DBMSs) are pervasive in all applications. Hence, practitioners aim at optimal performance by database tuning. However, administrating and optimizing relational DBMSs are costly tasks [42]. Therefore researchers are developing self-tuning techniques to continuously and automatically tune relational DBMSs [21, 41]. In recent years, various approaches came up to fulfill new requirements for database applications, e.g., Column-Store (CS) to improve analytical query performance [1, 2, 30, 46]. CS are well-suited for the on-line analytical processing (OLAP) domain, whereas RS are originally designed for the on-line transaction processing (OLTP) domain [6].

The I/O behaviour between main-memory and secondary storage is often a dominant factor in overall database system performance. At the same time, recent architectural advances suggest that CPU performance for memory-resident data is also a significant component of the overall performance [14, 2, 3, 8]. In particular, the CPU cache miss penalty may be relatively high, and may have a significant impact on query response times [2]. However, new requirements to OLAP systems [32, 39, 43], e.g., real-time updates in data warehouses, soften the borders between OLAP and OLTP. Consequently, the decision process to find the suitable relational DBMS is more complex due to workload decisions for OLTP or OLAP. Therefore, modern database systems should be designed for both I/O performance and CPU performance. PAX model is a little modification of RS that employs CS of each block [47]. PAX would have much better cache performance than RS, while leaving the same I/O characteristics as CS. Fractured mirror approach, supple the optimizer to choose any of two storage architectures [48]. However for all types of queries neither of one is optimal.

In this paper we study the concept of PAX and Fractured Mirror for Hybrid Storage Architecture. The background related to our work and challenges are discussed in Section 2. Based on the workload mix (queries and updates), the complexity of queries, and the update frequency, application may choose more suitable storage architecture. Section 3 presents the search space for storage architecture. Section 4 discuss the Hybrid Storage Architecture, which is comprised of analysis with statistics and estimation. Experimental analysis with Hybrid Storage Architecture is presented in Section 5. Finally, we conclude in Section 6.

II. RELATED WORK AND CHALLENGES

In recent years, development of pure CS have improved drastically [1, 24, 35, 46]. The simulation of CS architecture under RS is carried through Index-Only Plans, hence pays performance penalty for RS [3]. To reduce the performance penalty, C-Store database architecture design has included different storage schemes for CS and RS [33]. Design advisors have developed pre-configurations for databases e.g., IBM DB2 Configuration Advisor [23]. Gathering and utilizing the statistics directly from relational DBMS to advise index and materialized view configurations have been discussed in [44, 45]. Two similar approaches are available in literature to illustrate the tuning process using constraints such as storage space threshold [8, 9]. However, these approaches operate on single systems instead of comparing two or more systems according to their architecture.

The importance of cache performance in query processing was studied in [49, 50], and PAX was proposed as a solution [4]. The notion of using CS for good disk bandwidth and cache performance is similar to the notion of building covering indices for the query at hand. But for ad-hoc query workloads, it may not be possible to build efficient covering indices. Using mirrors to optimize reads by distributing random seeks between the disks was first discussed in [51]. This technique was extended to optimize write performance. The notion of distorted mirrors, which worked at the granularity of disk blocks, cleverly managed the blocks in two partitions [52].

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

Formerly, the only architecture for (large-scale) relational OLAP systems was RS. Performance is governed by the design decisions of RS, which is driven by the predictable workloads. CS are faster than RS for OLAP workloads [3, 36]. CS is more preferred design for OLAP workloads. But, CS performs worse on tuple and update operations [4, 19, 29]. Compared to CS, RS performs better on tuple operations, thus, we assume that there are still research fields for both, RS and CS in the OLAP domain.

III. WORKLOAD PATTERNS

We analyze a given workload to select the optimal storage architecture. Due to the influence of operations on the performance, we map operations of a workload and their statistics to evaluable workload patterns. To analyze the influence of single operations, we identified three workload patterns concerning operations of queries in workloads. The three workload patterns are namely: tuple operations, aggregations and groupings, and join operations. First, for tuple operations pattern, RS directly process tuples instead of tuple reconstruction as in CS, which need costly tuple reconstruction to process tuples. To characterize tuple operations more precisely, the sub-patterns are identified namely: Order operation, Retrieval Operation, Projection. Second, for aggregation and grouping pattern, CS processes single columns except for grouping operations. For single column processing, CS performs well on aggregations. Third, join operations are basic but costly tasks for DBMS which significantly affect any relational DBMS. The CS supports positional or vector based join techniques while RS have to maintain costly structures, e.g. bitmap(join) indexes [25]. To use the advantages of mirrored architecture for a workload pattern, a relation is mirrored on CS and RS (Figure 1). Query optimizer would choose dynamically the most efficient and I/O beneficial storage for the given workload. The Hybrid Storage Architecture pays performance penalty for uneven query distribution and disk utilization. Random seeks may not be distributed between the mirrors, as RS and CS are not having similar performance for index lookups.

IV. HYBRID STORAGE ARCHITECTURE

We integrated our workload patterns and the statistics with Hybrid Storage Architecture (Figure 2). The parameters to be optimized namely: throughput, average query response time, and optimized load balance. According to given parameters, an important point for ranking alternatives is the evaluation function. Selection of optimal storage architecture is driven by query behavior and cost model for statistics generation. For Hybrid Storage Architecture, we differentiated between statistical and estimated decisions. The search space in statistical model is assumed to be without uncertainty. For Hybrid Storage Architecture, CS is applied to the attributes for ad-hoc queries and one mirrored copy of each relation is stored in a RS representation (Figure 2).

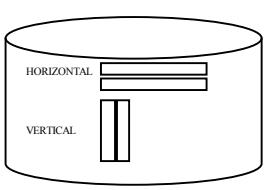


Figure 1: Fractured Mirror

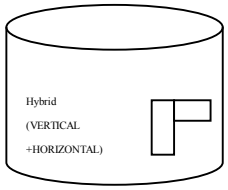


Figure 2: Hybrid Storage

A. Analysis with Statistics

For a Hybrid Storage Architecture model, we use statistics and query plans through relational DBMS [17]. We assume that statistics are available for both architectural designs (RS and CS), and the combined statistics can be used for Hybrid Storage Architecture. The architecture decision is based on the query evaluation cost [Equation 1]. Let the cost for given architecture is C(i,j),where i is storage architecture and j is the number of task T performed while executing the queries. The actual cost will be the minimum of three storage architectures.

Equation 1: $min \quad \sum C_{(i,j)}$ *with the constraints:*

$i \in \{CS, RS, Hybrid\ Storage\}$

$j \in T$

I. Either all or none of the task performed.

$\sum C_{ij} = (T=0 \text{ or } T=|T|)$

$$\forall i \in (CS, RS, Hybrid\ Storage)$$

$j \in T$

II. Frequency of task being performed has no limit.

$\sum C_{ij} = (T=\infty)$

$\forall i \in (CS, RS, Hybrid\ Storage)$

$j \in T$

Given an existing relational database system and a workload, we access and extract the statistics on operational costs [28]. The first constraint (I in Equation 1) ensures either all or none of the tasks are performed by an architecture. The second constraint (II in Equation 1) guarantees that all tasks are executed with no limitations on frequency. The extraction of cost values may be architecture-dependent and is part of the workload decomposition.

B. Analysis with estimation

For ad-hoc queries, the structure is not known but can be estimated. The multi-dimensionality appears due to different tasks within query plans. We use probability theory for sample workload statistics, to represent future workload and changes in the relational DBMS behavior. The estimation of all aspects can be combined into a cost function. We partition the tasks according to our workload structure TWL. As a result, we use the task set TWL=

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

{Join, Tuple Operations, aggregation and Grouping } where the elements are further refined. This enables us to estimate the cost of unknown workloads.

V. EXPERIMENTS ON TPC-H SUITE

To show the complexity of decisions, we introduce our example based on the TPC-H benchmark [37]. An experimental setup was built using Oracle 11g R2 with Red hat linux 5, 1GB of RAM, 80GB 2 HARD DISKS. The Buffer pool size set to 256 MB, with 16 KB block size. 10 GB of data is used for comparisons of these approaches. To demonstrate influences of a single operation to the query performance, we modify the number of returned attributes of the TPC-H queries Q2, and Q3. In other words, modifications to a single operation have different impacts. Hence, we state that a general decision regarding the storage architecture is not possible based on the query structure, e.g., SQL syntax. We used statistics to calculate evaluation cost of a single operation for a storage architecture [28].

TPC-H Query 2:

The query is having restrictions on only one dimension. The TPC-H query 2, which has rather unusual restrictions on the fact table as well; however the rationale for these Fact table restrictions seems reasonable. The query is meant to quantify the amount of revenue increase that would have resulted from eliminating certain company-wide discounts in a given percentage range for products shipped in a given year. This is a "what if" query to find possible increase in revenue. Since our lineorder table doesn't list shipdate, we will replace shipdate by orderdate in the flight.

select sum(lo_extendedprice*lo_discount) as revenue from lineorder, date where lo_orderdate = d_datekey and d_year = [YEAR] and lo_discount between [DISCOUNT] - 1 and
[DISCOUNT] + 1 and lo_quantity < [QUANTITY];

TPC-H Query 3:

For restriction on two dimensions, our query will compare revenue for some product classes, for suppliers in a certain region, grouped by more restrictive product classes and all years of orders; since TPC-H has no query of this description, we add it here.

select sum(lo_revenue), d_year, p_brand1 from lineorder, date, part, supplier where lo_orderdate = d_datekey and lo_partkey = p_partkey and lo_suppkey = s_suppkey and p_category = 'MFGR#12' and s_region = 'AMERICA' group by d_year, p_brand1 order by d_year, p_brand1;

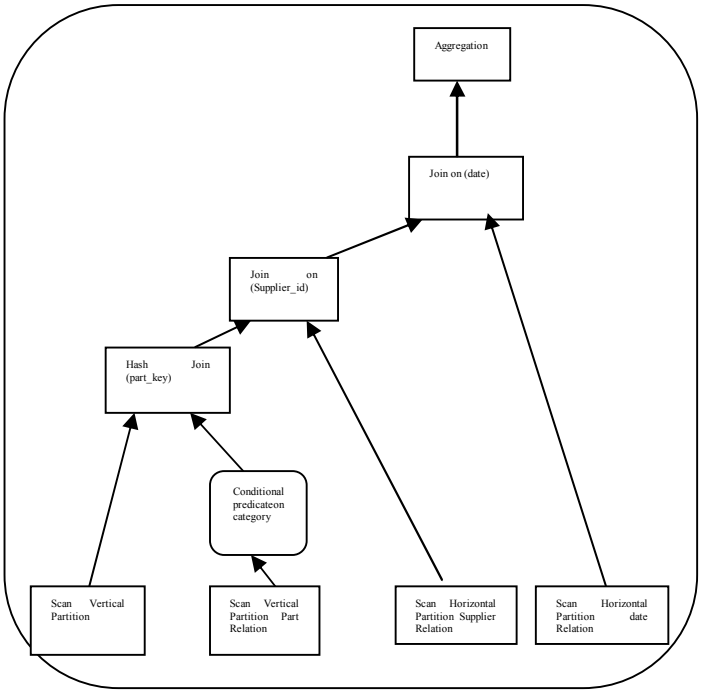


Figure 3: Hybrid Execution Plan

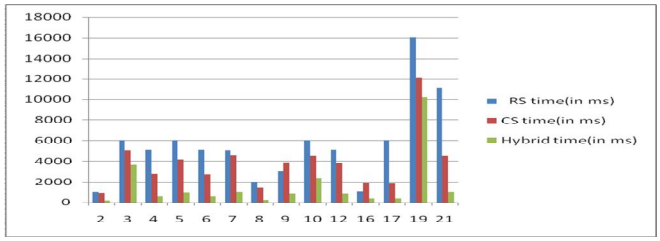


Figure 4: Execution time measure for Storage Architecture

VI. RESULT ANALYSIS AND CONCLUSION

In recent years, CS showed good results for OLAP applications, thus, CS (mostly) outperforms established RS. Nevertheless, new requirements arise in the OLAP domain that may not only be satisfied by CS, e.g., sufficient update processing. Thereby, the complexity of design process have increased. We chose Hybrid Storage Architecture based on workload patterns to minimize the complexity of design. The workload patterns contain all workload information, e.g., statistics and operation cost. As observed in Figure 4 the performance improvement for Hybrid Storage Architecture on average is 69%. Our model is based on cost functions (tuning objective) and constraints, but is developed on an abstract level, i.e., we can modularly refine or extend our model.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

REFERENCES

[1] D. J. Abadi, "Query execution in column-oriented database systems," Ph.D. dissertation, Cambridge, MA, USA, 2008, dviser: Madden, Samuel.

[2] D. J. Abadi, P. A. Boncz, and S. Harizopoulos, "Column oriented database systems," *PVLDB*, vol. 2, no. 2, pp. 1664–1665, 2009.

[3] D. J. Abadi, S. R. Madden, and N. Hachem, "Column-stores vs. rowstores: How different are they really?" in *SIGMOD '08*, 2008, pp. 967–980.

[4] D. J. Abadi, D. S. Myers, D. J. DeWitt, and S. Madden, "Materialization strategies in a column-oriented DBMS," in *ICDE '07*. IEEE, 2007, pp.466–475.

[5] A. Ailamaki, D. J. DeWitt, M. D. Hill, and M. Skounakis, "Weaving relations for cache performance," in *VLDB '01*. Morgan Kaufmann Publishers Inc., 2001, pp. 169–180.

[6] M. M. Astrahan, M. W. Blasgen, D. D. Chamberlin, K. P. Eswaran, J. Gray, P. P. Griffiths, W. F. K. III, R. A. Lorie, P. R. McJones, J. W.Mehl, G. R. Putzolu, I. L. Traiger, B. W. Wade, and V. Watson, "System R: Relational approach to database management," *TODS*, vol. 1, no. 2, pp. 97–137, 1976.

[7] B. Berendt and V. K¨oppen, "Improving ranking by respecting the multidimensionality and uncertainty of user rreferences," in *Intelligent Information Access*, ser. Studies in Computational Intelligence, G. Armano, M. de Gemmis, G. Semeraro, and E. Vargiu, Eds., 2010.

[8] N. Bruno and S. Chaudhuri, "To tune or not to tune? A lightweight physical design alerter," in *VLDB '06*. VLDB Endowment, 2006, pp.499–510.

[9] N. Bruno and S. Chaudhuri "An online approach to physical design tuning," in *ICDE '07*. IEEE, 2007, pp. 826–835.

[10] S. Chaudhuri and V. Narasayya, "Autoadmin "what-if" index analysis utility," in *SIGMOD '98*. ACM, 1998, pp. 367–378.

[11] S. Chaudhari, V. Narasayya, "Self-tuning database systems: A decade of progress," in *VLDB'07*. VLDB Endowment, 2007, pp. 3–14.

[12] V. Chvatal, *Linear programming*. W. H. Freeman and Company,September 1983.

[13] G. P. Copeland and S. N. Khoshafian, "A decomposition storage model," in *SIGMOD '85*. ACM, 1985, pp. 268–279.

[14] D. W. Cornell and P. S. Yu, "An effective approach to vertical partitioning for physical design of relational databases," *Trans. Softw. Eng.*,vol. 16, no. 2, pp. 248–258, 1990.

[15] C. Favre, F. Bentayeb, and O. Boussaid, "Evolution of data warehouses' optimization: A workload perspective," in *DaWaK '07*. Springer, 2007,pp. 13–22.

[16] J. Figueira, S. Greco, and M. Ehrgott, Eds., *Multiple criteria decision analysis: State of the art surveys*. Boston:Springer Science and Business Media, 2005. [Online]. Available: http://www.springerlink.com/content/978-0-387-23067-2

[17] S. J. Finkelstein, M. Schkolnick, and P. Tiberio, "Physical database design for relational databases," *TODS*, vol. 13, no. 1, pp. 91–128, 1988.

[18] R. Fourer, D. M. Gay, and B. W. Kernighan, *AMPL: A modeling language for mathematical programming*, 2nd ed. Duxbury Press, 2002.

[19] S. Harizopoulos, V. Liang, D. J. Abadi, and S. Madden, "Performance tradeoffs in read-optimized databases," in *VLDB '06*. VLDB Endowment,2006, pp. 487–498.

[20] M. Holze, C. Gaidies, and N. Ritter, "Consistent on-line classification of DBS workload events," in *CIKM '09*. ACM, 2009, pp. 1641–1644.

[21] IBM. (2006, June) An architectural blueprint for autonomic computing White Paper. Fourth Edition, IBM Corporation. [Online]. Available:http://www-01.ibm.com/software/tivoli/autonomic/

[22] Ingres/Vectorwise, "Ingres/VectorWise sneak preview on the Intel Xeon processor 5500 series-based platform," White Paper, September 2009.[Online]. Available: http://www.ingres.com/vectorwise/

[23] E. Kwan, S. Lightstone, K. B. Schiefer, A. J. Storm, and L. Wu, "Automatic database configuration for DB2 Universal Database: Compressing years of performance expertise into seconds of execution," in *BTW '03*. GI, 2003, pp. 620–629.

[24] T. Legler, W. Lehner, and A. Ross, "Data mining with the SAP NetWeaver BI Accelerator," in *VLDB '06*. VLDB Endowment, 2006,pp. 1059–1068.

[25] A. L¨ubcke, "Cost-effective usage of bitmap-indexes in DS-Systems," in *20th Workshop "Grundlagen von Datenbanken"*. School of Information Technology, International University in Germany, 2008, pp. 96–100.

[26] A. L¨ubcke, "Challenges in workload analyses for column and row stores," in *22nd Workshop "Grundlagen von Datenbanken"*, vol.581. CEUR-WS.org, 2010. [Online]. Available: http://ceur-ws.org/Vol-581/gvd2010 8 1.pdf

[27] A. L¨ubcke, I. Geist, and R. Bubke, "Dynamic construction and administration of the workload graph for materialized views selection," vol. 1,no. 3. DIRF, 2009, pp. 172–181.

[28] A. L¨ubcke, V. K¨oppen, and G. Saake, "A query decomposition approach for relational DBMS using different storage architectures," School of Computer Science, University Magdeburg, Tech. Rep., 2011, technical Report, Submitted for Publication.

[29] S. Manegold, P. Boncz, N. Nes, and M. Kersten, "Cache-conscious radixdecluster projections," in *Proceedings of the Thirtieth international conference on Very large data bases - Volume 30*, ser. VLDB '04. VLDB Endowment, 2004, pp. 684–695. [Online]. Available:http://portal.acm.org/citation.cfm?id=1316689.1316749

[30] H. Plattner, "A common database approach for OLTP and OLAP using an in-memory column database," in *SIGMOD '09*, 2009, pp. 1–2.

[31] K. E. E. Raatikainen, "Cluster analysis and workload classification,"*SIGMETRICS PER*, vol. 20, no. 4, pp. 24–30, 1993.

[32] R. J. Santos and J. Bernardino, "Real-time data warehouse loading methodology," in *IDEAS '08*, 2008, pp. 49–58.

[33] J. Schaffner, A. Bog, J. Kr¨uger, and A. Zeier, "A hybrid row-column OLTP database architecture for operational reporting," in *BIRTE '08*,2008.

[34] C. Schneeweiß, *Planung 1, Systemanalytische und entscheidungstheoretische Grundlagen*. Berlin: Springer, 1991.

[35] D. ´ Sle¸zak, J. Wr´oblewski, V. Eastwood, and P. Synak, "Brighthouse: An analytic data warehouse for ad-hoc queries," *PVLDB*, vol. 1, no. 2, pp.1337–1345, 2008.

[36] M. Stonebraker, D. J. Abadi, A. Batkin, X. Chen, M. Cherniack, M. Ferreira, E. Lau, A. Lin, S. Madden, E. J. O'Neil, P. E. O'Neil,A. Rasin, N. Tran, and S. B. Zdonik, "C-Store: A column-oriented DBMS," in *VLDB '05*. VLDB Endowment, 2005, pp. 553–564.

[37] Transaction Processing Performance Council. (2010, April) TPC BENCHMARK*TM* H. White Paper. Decision Support Standard Specification,Revision 2.11.0. [Online]. Available:http://www.tpc.org/tpch/spec/tpch2.11.0.pdf

[38] C. Turbyfill, "Disk performance and access patterns for mixed database workloads," *Data Eng. Bulletin*, vol. 11, no. 1, pp. 48–54, 1988.

[39] A. A. Vaisman, A. O. Mendelzon, W. Ruaro, and S. G. Cymerman,"Supporting dimension updates in an OLAP server," *Information Systems*,vol. 29, no. 2, pp. 165–185, 2004.

[40] D. von Winterfeldt and W. Edwards, *Decision analysis and behaviorresearch*. Cambridge University Press, 1986.

[41] G. Weikum, C. Hasse, A. Moenkeberg, and P. Zabback, "The COMFORT automatic tuning project, invited project review," *Information Systems*, vol. 19, no. 5, pp. 381–432, 1994.

[42] G. Weikum, A. C. Knig, A. Kraiss, and M. Sinnwell, "Towards selftuning memory management for data servers," *IEEE Data Eng. Bulletin*, vol. 22, pp. 3–11, 1999.

[43] Y. Zhu, L. An, and S. Liu, "Data updating and query in real-time data warehouse system," in *CSSE '08*, 2008, pp. 1295–1297.

[44] D. C. Zilio, J. Rao, S. Lightstone, G. M. Lohman, A. J. Storm, C. Garcia-Arellano, and S. Fadden, "DB2 Design Advisor: Integrated automatic physical database design," in *VLDB '04*. VLDB Endowment, 2004, pp.1087–1097.

[45] D. C. Zilio, C. Zuzarte, S. Lightstone, W. Ma, G. M. Lohman,R. Cochrane, H. Pirahesh, L. S. Colby, J. Gryz, E. Alton, D. Liang,and G. Valentin, "Recommending materialized views and indexes with IBM DB2 Design Advisor," in *ICAC '04*, 2004, pp. 180–188.

[46] M. Zukowski, P. A. Boncz, N. Nes, and S. Heman, "MonetDB/X100- a DBMS in the CPU cache," *Data Eng. Bulletin*, vol. 28, no. 2, pp.17–22, June 2005.

[47] J.Blakeley, W.J.McKenna, G.Graefe. Experiences building the open OODB Query Optimizer. Proceedings of ACM SIGMOD 1993.

[48] Cornell D.W., Yu P.S. *An Effective Approach to Vertical Partitioning for Physical Design of Relational Databases*.IEEE Transactions on Software Engg, Vol 16, No 2, 1990

[49] D.Bitton, J.Gray. Disk Shadowing. Proceedings of VLDB 1988.

[50]C.Orji, J.Solworth. Doubly-Distorted Mirrors. Proceedings of ACM SIGMOD 1993.

[51] Carey et al. Shoring up Persistent Applications. Proceedings of ACM SIGMOD 1994.

[52]A.Szalay et al. Designing and mining multi-terabyte astronomy archives: The Digital Sky Survey. Proceedings of ACM SIGMOD 2000.

[53] Ramamurthy R., Dewitt D.J., and Su Q. *A Case for Fractured Mirrors*. Proceedings of VLDB 2002.

Authors

Tejaswini Apte received her M.C.A(CSE) from Banasthali Vidyapith, Rajasthan. She is pursuing research in the area of Machine Learning from DAU, Indore,(M.P.),INDIA.

Presently she is working as an ASSISTANT PROFESSOR in Symbiosis Institute of Computer Studies and Research at Pune. She has 5 papers published in International/National Journals and Conferences. Her areas of interest include Databases, Data Warehouse and Query Optimization.

Mrs. Tejaswini Apte has a professional membership of ACM.

Maya Ingle did her Ph.D in Computer Science from DAU, Indore (M.P.) , M.Tech (CS) from IIT, Kharagpur, INDIA, Post Graduate Diploma in Automatic Computation, University of Roorkee, INDIA, M.Sc. (Statistics) from DAU, Indore (M.P.) INDIA.

She is presently working as PROFESSOR, School of Computer Science and Information Technology, DAU, Indore (M.P.) INDIA. She has over 100 research papers published in various International / National Journals and Conferences. Her areas of interest include Software Engineering, Statistical, Natural Language Processing, Usability Engineering, Agile computing, Natural Language Processing, Object Oriented Software Engineering. interest include Software Engineering, Statistical Natural Language Processing, Usability Engineering, Agile computing, Natural Language Processing, Object Oriented Software Engineering.

Dr. Maya Ingle has a lifetime membership of ISTE, CSI, IETE. She has received best teaching Award by All India Association of Information Technology in Nov 2007.

Arvind Goyal did his Ph.D in Physics from Bhopal University, Bhopal

(M.P.), M.C.M from DAU, Indore, M.Sc.(Maths) from U.P.

He is presently working as SENIOR PROGRAMMER, School of Computer Science and Information Technology, DAU, Indore (M.P.). He has over 10 research papers published in various International/National Journals and Conferences. His areas of interest include databases and data structure.

Dr. Arvind Goyal has lifetime membership of All India Association of Education Technology.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

APPLICATION OF COMBINED RADIAL BASIS FUNCTION AND ECHO STATE NEURAL NETWORK (RBFESNN) FOR MINING HAMILTON RATING SCALE DEPRESSION DATA

[1]Bhuvana R., [2]Purushothaman S., and [3]Rajeswari R.,

[1]Bhuvana R., Research Scholar, Department of Computer Science, VELS University, Chennai, INDIA-600117.	[2]Dr.Purushothaman S., Professor, PET Engineering College, Vallioor, INDIA-627117,	[3]Rajeswari R., Research scholar, Mother Teresa Women's University, Kodaikanal, INDIA-624101. .

Abstract-Mining of **depression data such as depressed mood, feelings of guilt, suicide, insomnia early, insomnia middle, insomnia late, work and activities, retardation, Psychomotor, agitation, anxiety, anxiety somatic, somatic symptoms, somatic symptoms general, genital symptoms, genital symptoms, insight, diurnal variations, depersonalization and decreolization, paranoid symptoms, obsessionals and compulsive symptoms have been collected based on the Hamilton rating scale for depression. This paper presents the implementation of neural network methods for depression data mining and diagnosis patients by using radial basis function (RBF) and Echo state neural network (ESNN). The output of RBF is given as input to ESNN network. A systematic approach has been developed to efficiently mine the depression data for proper diagnosis of the patients.**

Keywords: Hamilton Rating Scale Depression data, radial basis function (RBF), echo state neural network (ESNN)

[I]. INTRODUCTION

Depression is associated with high levels of co-morbidity with other conditions such as anxiety disorders, substance abuse and eating disorders. Depression is more common in adults than in children.

A list of the signs or symptoms of major depression:

a) Sadness, depressed mood, crying over seemingly minor setbacks.
b) Increased irritability, crankiness, difficulty being satisfied.
c) More easily frustrated, gives up quickly after initial failures.
d) Poor self-concept, low self-esteem, reluctance toward attempting endeavors.
e) Loss of interest in previously pleasurable activities.
f) Changes in appetite (decreased appetite most common) often signaled by rapid weight gain or loss.
g) Changes in sleep patterns (not enough or too much sleep).
h) Slowed, inhibited actions (slow, soft speech, slowed body movements).
i) Fatigue, loss of energy.
j) Poor concentration, attention and/or memory.
k) Thoughts or words about death or suicide.

[II]. RELATED WORKS

Jyoti, 2012, focused on depression analysis based on visual cues from facial expressions and upper body movements.

Danuta, 2012, states that family history of major depressive disorder (MDD) increases individuals'

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

vulnerability to depression and alters the way depression manifests itself.

[III]. MATERIALS AND METHODOLOGY

Depression [Danuta, 2012, Ahlberg, 2002] data is collected by using questionnaire. The data has been collected from the patients based on the Hamilton rating scale (HRS) [Hedlund, 1979]. The HRS has 21 categories of depression types identified with each depression scale in a range of 0 to 4 numerical values.

Table 1 Depression data

INPUTS

Depressed mood	Feeling of Guilt	Suicide	Insomnia Early	Insomnia middle	Insomnia late	Work and activities	Retardation psychomotor	Agitation	Anxiety	Anxiety somatic	Somatic symptoms	Somatic general	Genital symptoms	Hypochondriasis	Loss of weight	Insight	Diurinal variation	Depersonalization and devariation	Paranoid symptoms	Obsessional and compulsive symptoms
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	1
0	0	0	0	2	0	0	0	0	2	0	0	0	0	2	0	0	0	0	0	2
0	0	0	1	0	0	0	0	0	3	0	0	0	0	3	0	0	0	0	1	0
0	0	0	1	1	0	0	0	0	4	0	0	0	0	4	0	0	0	0	1	1
0	0	0	1	2	0	0	0	1	0	0	0	0	1	0	0	0	0	0	1	2
0	0	0	2	0	0	0	0	1	1	0	0	0	1	1	0	0	0	0	2	0
0	0	0	2	1	0	0	0	1	2	0	0	0	1	2	0	0	0	0	2	1
0	0	0	2	2	0	0	0	1	3	0	0	0	1	3	0	0	0	0	2	2
0	0	1	0	0	0	0	0	1	4	0	0	0	1	4	0	0	0	0	3	0
0	0	1	0	1	0	0	0	2	0	0	0	0	2	0	0	0	0	0	3	1
0	0	1	0	2	0	0	0	2	1	0	0	0	2	1	0	0	0	0	3	2
0	0	1	1	0	0	0	0	2	2	0	0	0	2	2	0	0	0	1	0	0
0	0	1	1	1	0	0	0	2	3	0	0	0	2	3	0	0	0	1	0	1
0	0	1	1	2	0	0	0	2	4	0	0	0	2	4	0	0	0	1	0	2
0	0	1	2	0	0	0	0	3	0	0	0	1	0	0	0	0	0	1	1	0
0	0	1	2	1	0	0	0	3	1	0	0	1	0	1	0	0	0	1	1	1
0	0	1	2	2	0	0	0	3	2	0	0	1	0	2	0	0	0	1	1	2
0	0	2	0	0	0	0	0	3	3	0	0	1	0	3	0	0	0	1	2	0
0	0	2	0	1	0	0	0	3	4	0	0	1	0	4	0	0	0	1	2	1

DISTRIBUTION OF PATTERNS

Total Range Categorized in the target values	Target Value	No. Of Patterns In Each Category	No. of patterns for training
<=15	1	49	9
>15 AND <=30	2	786	18
>30 AND <=45	3	903	18
>45	4	62	10
	Total	1800	55

A Radial Basis Function Network

A Radial Basis Function (RBF) [Schwenker, 2001] network (Park, 1991) is a two-layer network whose output nodes form a linear combination of the basis (or kernel) functions computed by the hidden layer nodes. The basis functions in the hidden layer produce a localized response to input stimulus. They produce a significant nonzero response only when the input falls within a small localized region of the input space. The most common basis is a Gaussian kernel function of the form:

$$u_{1,j} = \exp\left[-\frac{(\boldsymbol{x} - \boldsymbol{w}_{1,j})^T (\boldsymbol{x} - \boldsymbol{w}_{1,j})}{2\sigma_j^2} \right] \quad j = 1, 2, \ldots, N_1$$

where

u_{ij} is the output of the j^{th} node in the first layer,

x is the input pattern,

w_{ij} is the weight vector for the j^{th} node in the first layer, i.e., the center of the Gaussian for node j,

σ_j^2 is the normalization parameter for the j^{th} node, and

N_1 is the number of nodes in the first layer.

The node outputs are in the range from zero to one so that the closer the input is to the center of the Gaussian, the larger the response of the node. Gaussian kernels are radially symmetric; each node produces an identical output for inputs that lie a fixed radial distance from the center of the kernel w_{ij}. The output layer node equations are given by:

$$y_j = w^T_{2j}\, u_i \quad j=1,2,\ldots., N_2$$

where

y_j is the output of the j^{th} node,

w_{2j} is the weight vector for this node, and

u_i is the vector of outputs from the first layer.

N_2 is the number of nodes in the output layer.

The output layer nodes form a weighted linear combination of the outputs from the first layer. Thus, the overall network performs a nonlinear transformation forming a linear combination of the nonlinear basis functions.

Training RBF is done as follows,

Step 1: Finding distance between patterns and centers.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

Step 2: Creating an RBF matrix whose size will be (np X cp),

where np= number of patterns (patterns X number of features) used for training and

cp is number of centers which is equal to 10.

The number of centers chosen should make the RBF network learn the maximum number of training patterns under consideration.

Step 3: Final weights which are inverse of RBF matrix multiplied with Target values are calculated.

Step 4: During testing the performance of the RBF network, RBF values are formed from the features obtained are processed with the final weights obtained during training. Based on the result obtained, classification is done.

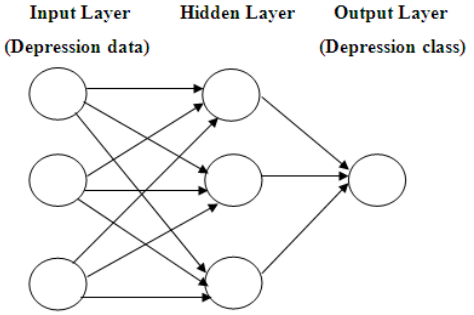


Fig.1 Training RBF

Training RBF
Step 1: Apply RBF. No. of Input = 21 No. of Patterns = 100 No. of Centre = 10 Step 2: Calculate RBF= exp (-x) Step 3: Calculate Matrix as G = RBF A = GT * G Step 4: Calculate, B = A-1 Step 5: Calculate, E = B * GT Step 6: Calculate the Final Weight, F = E * D Step 7: Store the Final Weights in a File.

Testing RBF
Step 1:.Input depression pattern. **Step 2:** Trained weights are read **Step 3:** Calculation of the output is done Output= F * E **Step 4:** The output is compared with the templates.

B Echo State Neural Network

In the ANN architecture, dynamic computational models require the ability to store and access the time history of their inputs and outputs. The most common dynamic neural architecture is the Time-Delay Neural Network (TDNN) [Weishui, 2000] that couples delay lines with a nonlinear static architecture where all the parameters (weights) are adapted with the BPA. Recurrent Neural Networks (RNNs) [Atiya, 2000] implement a different type of embedding. Back propagation through time and real-time recurrent learning, have been proposed to train RNNs. The problem of decaying gradients has been addressed with special processing elements (PEs).

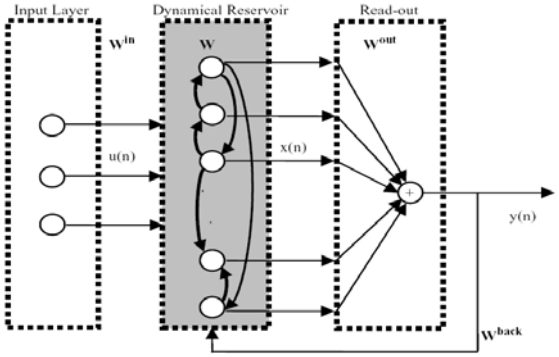


Fig. 1 An Echo State Neural Network (ESNN)

The ESNN shown in Figure 1, has been developed by [Jaeger, 2001]. The topology possesses highly interconnected and recurrent connections of nonlinear PEs or echo states. The PE is called reservoir [Lukoševičius, 2009]. The reservoirs contain rich dynamics and information about the history of input and output patterns. The outputs of these internal PEs (echo states) are fed to a memory less but adaptive readout network (generally linear) that produces the network output. In the topology, only the memory less readout is trained. In the recurrent topology there are fixed connection weights. This reduces the complexity of RNN training to simple linear regression while preserving a recurrent topology.

The echo state condition is defined in terms of the spectral radius (the largest among the absolute values of the eigenvalues of a matrix, denoted by (|| ||) of the reservoir's weight matrix (|| W ||<1). This condition states that the dynamics of the ESNN is uniquely controlled by the input, and the effect of the initial states vanishes. The current design of ESNN parameters relies on the selection of spectral radius.

The recurrent discrete-time neural network has 'M' input units, 'N' internal PEs, and 'L' output units. The value of the input unit at time 'n' is

$$u(n) = [u_1(n), u_2(n), \ldots, u_M(n)]^T,$$

The internal units are represented by

$$x(n) = [x_1(n), x_2(n), \ldots, x_N(n)]^T$$

The output units are represented by

$$y(n) = [y_1(n), y_2(n), \ldots, y_L(n)]^T$$

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

The connection weights are given as follows:

a) In an (N x M) weight matrix $W^{back} = W_{ij}^{back}$ for connections between the input and the internal PEs,

b) In an N × N matrix $W^{in} = W_{ij}^{in}$ for connections between the internal PEs

c) In an L × N matrix $W^{out} = W_{ij}^{out}$ for connections from PEs to the output units and,

d) In an N × L matrix $W^{back} = W_{ij}^{back}$ for the connections that project back from the output to the internal PEs.

The activation of the internal PEs are updated as follows:

$x_{(n+1)} = f(W^{in} u_{(n+1)} + Wx(n) + W^{back} y(n))$,

where $f = (f_1, f_2, \ldots, f_N)$ are the internal PEs' activation functions.

All f_i's are hyperbolic tangent functions $\frac{e^x - e^{-x}}{e^x + e^{-x}}$.

The output from the readout network is computed as follows:

$y(n + 1) = f^{out}(W^{out}x(n + 1))$,

where

$f^{out} = (f_1^{out}, f_2^{out}, \ldots, f_L^{out})$ are the output unit's nonlinear functions.

Training ESNN Algorithm

The algorithm for training the ESNN is as follows:

Step 1: Read a depression pattern. The number of nodes in the input layer is equal to number of features of the depression pattern.

Step 2: Decide the number of reservoirs.

Step 3: The number of nodes in the output layer is equal to 1.

Step 4: Random weights are initialized between input and hidden layer (Ih) hidden and output (ho).

Step 6: Calculate F=Ih*I.

Step 7: Calculate TH = Ho * T.

Step 8: Calculate TT = R*S.

Step 9: Calculate S = tan h(F+TT+TH).

Step 10: Calculate a = Pseudo inverse (S).

Step 11: Calculate W_{out} = a * T and store W_{out} for testing

The algorithm for testing the ESNN is as follows:

Step 1: Read a depression pattern. The number of nodes in the input layer is equal to number of features of the depression pattern**.**

Step 2: Calculate F=Ih*I.

Step 3: TH = ho * T.

Step 4: TT = R*S.

Step 5: S = tan h(F+TT+TH).

Step 6: a = Pseudo inverse (S).

Step 7: Estimated = a * W_{out}.

Step 8: Compare the output with template to decide the category of the depression data.

[IV]. RESULT AND DISCUSSION

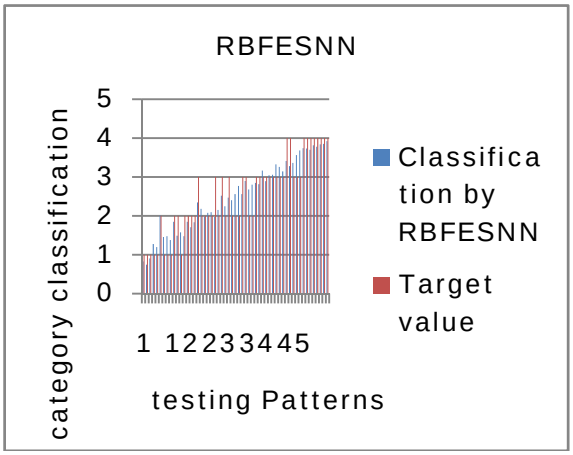


Fig. 3 Classification output of RBFESNN network

The figure 3 shows the target output and RBFESNN classification. The input layer has 21 nodes, hidden layer has 10 nodes and one node in the output layer. The x-axis shows the test patterns and the y-axis shows the classification by RBFESNN. The concept of distance measure is used to associate the input and output pattern values. RBF is capable of producing approximations to an unknown function 'f' from a set of input-output data set. The approximation is produced by passing an input point through a set of basis functions each of which contains one of the RBF centers multiplying the result of each function by a coefficient and then summing them linearly. The data were trained using RBF. The singe output of RBF is used as input for the ESNN. The number of reservoirs is 21. The number of nodes in the output layer of ESNN is 1.

[V]. CONCLUSION

This paper discusses the implementation of RBF ESNN neural network for classifying the depression data. The RBF network learns the depression data in one iteration. The amount of classification performance depends on the number of centers used for training RBF. In the RBF network, the number of nodes/centers has to be decided, based on the number of centers, the classification performance changes. In the ESNN network, the number of reservoirs has to be decided, the classification performance changes accordingly. This paper has carried out the analysis of neural networks algorithms and proposed their implementation for depression data mining. Hence, the research work is a good contribution in the field

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

of psychological depression data mining for diagnosis.

+REFERENCES

[1] Ahlberg A.C., Ljung T., Rosmond R., McEwen B., Holm G., Akersson H.O., and Bjorntorp P., 2002, Depression and anxiety symptoms in relation to anthropometry and metabolism in men, Psychiatry Research, Vol.112, No.2, pp.101-110.

[2] Danuta M. Lisiecka, Angella Carballedo, Andrew J. Fagan, Yolande Ferguson, James Meaney, Thomas Frodl, 2012, Recruitment of the left hemispheric emotional attention neural network in risk for and protection from depression, Journal of Psychiatry Neuroscience.

[3] Hedlund J.L., and Viewig B.W., The Hamilton rating scale for depression: a comprehensive review, J.Operat. Psychiat, 1979, Vol.10, pp.149-65.

[4] Jyoti Joshi, Abhinav Dhall, Roland Goecke. Michael Breakspear, and Gordon Parker, Neural-Net Classification For Spatio-Temporal Descriptor Based Depression Analysis, 21st International Conference on Pattern Recognition, November 2012, Tsukuba International Congress center, Japan.

[5] Lukoševičius M., and Jaeger H., 2009, Reservoir Computing Approaches to Recurrent Neural Network Training, Computer Science Review, Vol.3, No.3, pp.127-149.

[6] Park J., and Sandberg I.W., 1991, Universal Approximation Using Radial-Basis-Function Networks, Neural Computation, Vol.3, No.2, pp.246–257.

[7] Ramaraj E., and Radha P., Depression Data Mining using Back-Propagation Algorithm, IEEE International Advanced Computing Conference, Thapur University, Patiala, The Proceedings published in IEEE Explorer, March 6-7, 2009.

[8] Schmidhuber J., Gomez F., Wierstra D., and Gagliolo M., 2007, Training recurrent networks by evolino. Neural Computation, Vol.19, No.3, pp.757-779.

[9] Schwenker Friedhelm, Kestler Hans A., Palm Günther, Three learning phases for radial-basis-function networks, Neural Networks, 2001, Vol.14, pp.439–458.

[10] Yildiz I.B., Jaeger H., and Kiebel S.J., 2012, Re-visiting the echo state property, Neural Networks,Vol.35, pp.1-20.

R. Bhuvana, doing her Ph.D. at Vels University Chennai. She received M.S (IT), M.Phil. Degree from Bharathidasan University Trichy. She has 7 years of teaching experience in various arts & science colleges. She is presently working as Asst. Professor Dept. of BCA, Veils University Chennai. She has published a number of papers in Conferences, and attends the workshops. Her research interest includes text mining, data mining & data warehousing.

Dr.S.Purushothaman completed his PhD from Indian Institute of Technology Madras, India in 1995. He has 129 publications to his credit. He has 19 years of teaching experience. Presently he is working as Professor in PET Engineering College, India

R.Rajeswari completed M.Sc., Information Technology from Bharathidasan university, Tiruchirappalli and M.Phil Computer Science from Alagappa University, Karaikudi, Tamilnadu, India. She is currently pursuing PhD in Mother Teresa Women's University. Her area of interest is Intelligent Computing, image processing.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

A model for control the traffic pollution using signals by Optimization method

D. Nagarajan,
Department of Information Technology.
Math Section,
Salalah college of Technology
Salalah, Sultanate of Oman.
.

M.Raji,
Department of Mathematics,
Asan Memorial College of Engineering and Technology,
Chengalpattu, India.
.

Abstract—Traffic signals are very important vital factor for reduce the traffic pollution in our world. The past three decades researches much attention about the traffic pollution. There are many opportunities to use clever traffic engineering to reduce the impacts of traffic on public transportation. Often these combine traffic signals with short sections of exclusive public transport lanes. The aim of the paper is to reduce the traffic pollution using traffic signal by Markov chain and genetic algorithm.

Keywords- traffic system; continuous time markov chain; genetic algorithm.

I. INTRODUCTION

Traffic congestion is no fun for anyone, but it's deadly for public transport. When buses and trams are stuck in traffic jams they fall behind schedule and, because this means that more people will be waiting at the next stops, they fall even further behind schedule leading to bunching and compounding delays. Bunched buses and delays make public transport unattractive for customers and increase operational costs, so congestion impacts on public transport must be eliminated whenever possible. Traffic light provides an accurate way to reach the places for the people by showing the lights in signal by means of Red, Yellow, Green .Green color light indicates the signs to go to the same direction. Yellow color light shows that to get ready to stop/start the vehicle. Traffic is automatically maintained by computer activated guidance system that to find out the traffic volume on the roadways. Global positioning satellite systems are fixed in many vehicles. This system shows the correct ways to find out the route the destination. This system utilizes to avoid the traffic congestion and give the paper route to the destination.

Markov processes with a discrete state space are called Markov chains. If the parameter t is discrete, the process is a discrete-time Markov chain. If the parameter t is Continuous, the process is a Continuous -time Markov chain [4].

Genetic algorithms are an optimization pattern for finding good solution to adapt the surrounding. In genetic algorithms, encoding the solution to the chromosomes and equate the corresponding fitness of the solutions. This evaluation is used to obtain the best adapted solution.

Genetic Algorithms are evolutionary techniques to search an optimal solution of the problem. It is inferred from Darwin's Evolution Theory. It was developed by John Holland. In existence, individuals reproduce offspring with different characteristics by occurring crossover between chromosomes and some causes genes are permuted by environment. This is known as a mutation. Parents are assessed by fitness function. New offsprings are produced by parents. The algorithm ends when a standard is attained [2].

II. DESCRIBTION OF THE MODEL

Consider a Traffic signal with three color lights (red,green,yellow). We can describe three states in our system, (i) Red, (ii) Green, (iii)Yellow. We assume that

- The time period during which the light is Red are exponentially distributed with parameter β,
- The time periods during which the light is Green are exponentially distributed with parameter α,
- The time period which the yellow light is turned on is exponentially distributed with parameter μ,
- The yellow light off time is exponentially distributed with parameter λ.

The transition matrix is [1]

$$Q=\begin{bmatrix} -\beta & \beta & 0 \\ \alpha & -(\alpha+\mu) & \mu \\ \lambda & 0 & -\lambda \end{bmatrix}$$

The Continuous time Markov chain is Ergodic and the steady state distribution is worked out by Solving the system of linear equations:

$\beta\,\pi_1=\alpha\pi_2+\lambda\pi_3$

$(\alpha+\mu)\pi_2=\beta\pi_1$

$\lambda\pi_3=\mu\pi_2$

$\pi_1+\pi_2+\pi_3=1$

Solving, $\pi=\frac{\lambda}{\lambda\beta(\alpha+\mu)+\mu\beta(\lambda+\beta)}(\lambda(\alpha+\mu), \lambda\beta, \beta\mu)$

From π, it is possible to compute several steady-state performance indices:

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

- π_2 is the fraction of time in which the light is green,
- π_3 is the fraction of time in which the light is yellow,
- $[\lambda\pi_3]^{-1} = [\mu\pi_2]^{-1}$ is the average time between two consecutive yellow lights,
- $[(\alpha+\mu)\ \pi_2]^{-1} = [\beta\pi_1]^{-1}$ is the average time between two consecutive.

Consider the following [3]
Minimum green light time =3s
Maximum green light time =30s
Extension gap time =3s
Yellow light time =3s
Red light time =0

A.*One way traffic*

The matrix is constructed from the above data

$$\begin{array}{c} \\ R \\ G \\ Y \end{array}\begin{array}{c} \begin{array}{ccc} R & G & Y \end{array} \\ \begin{bmatrix} -3 & 3 & 0 \\ 30 & -(30+3) & 3 \\ 3 & 0 & -3 \end{bmatrix} \end{array}$$

π_1=0.84615
π_2=0.07692
π_3=0.07692
Flow out =$\beta\pi_1$=3(0.84615) =2.53845
Flow in = $\alpha\pi_2+\lambda\pi_3$
=30(0.07692) +3(0.07692)
=2.53836
$\pi_2$is the fraction of time in which the light is green.
π_3 is the fraction of time in which the yellow light is turned on.

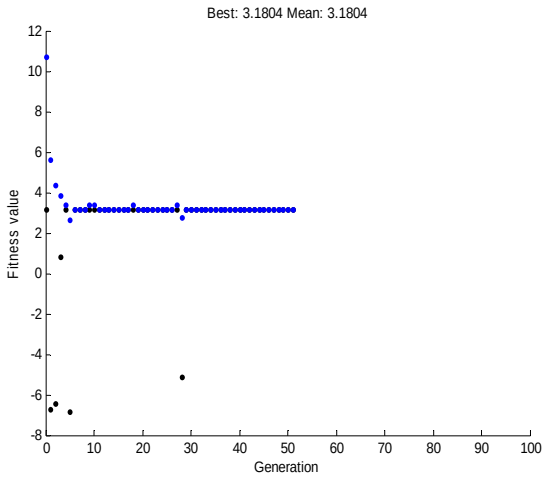


Fig.1.Fitness value between the generations for one way traffic

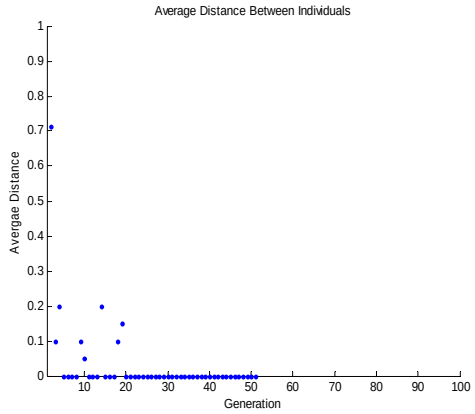


Fig.2.Average distance between individuals for one way traffic

From the above graphs, it seems that the average distance between individuals are substantial little. It shows that 3.1804 is the best fitness value for the given data.

B. *Two way Traffic*

G_1R_2 and R_1G_2matrices which are taken out from the above data.

G_1R_2:

$$\text{Data matrix is } \begin{bmatrix} 3 & 0 & 30 & 27 \\ 3 & 6 & 9 & 12 \\ 3 & 0 & 30 & 27 \\ 3 & 0 & 30 & 27 \end{bmatrix}$$

The TPM of the Markov chain is [5]and [6]

$$A = \begin{bmatrix} 0.05 & 0 & 0.5 & 0.45 \\ 0.1 & 0.2 & 0.3 & 0.4 \\ 0.05 & 0 & 0.5 & 0.45 \\ 0.05 & 0 & 0.5 & 0.45 \end{bmatrix}$$

The steady state distribution of the chain is
π = (0.05 0 0.5 0.45)

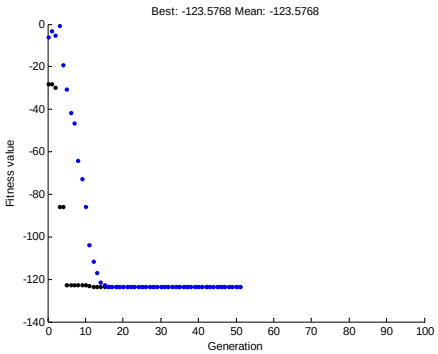


Fig.3.Fitness value between the generations for two way traffic (***G_1R_2***)

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

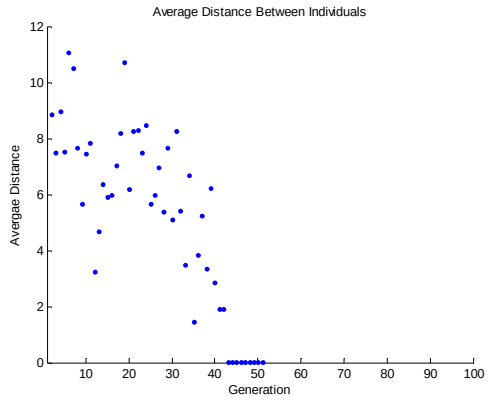


Fig.4. Average distance between individuals for two way traffic(G_1R_2)

From the above graphs, it reveals that the average distance between individuals are substantial little. It shows that -123.5768 is the best fitness value for the given data.

R_1G_2:

Data matrix is $\begin{bmatrix} 3 & 6 & 3 & 30 \\ 3 & 6 & 3 & 6 \\ 3 & 6 & 9 & 12 \\ 3 & 6 & 3 & 6 \end{bmatrix}$

The TPM of the Markov chain is

B= $\begin{bmatrix} 0.07 & 0.15 & 0.07 & 0.71 \\ 0.17 & 0.33 & 0.17 & 0.33 \\ 0.1 & 0.2 & 0.3 & 0.4 \\ 0.17 & 0.33 & 0.17 & 0.33 \end{bmatrix}$

The steady state distribution of the chain is

π = (0.1412 0.2810 0.1789 0.3969)

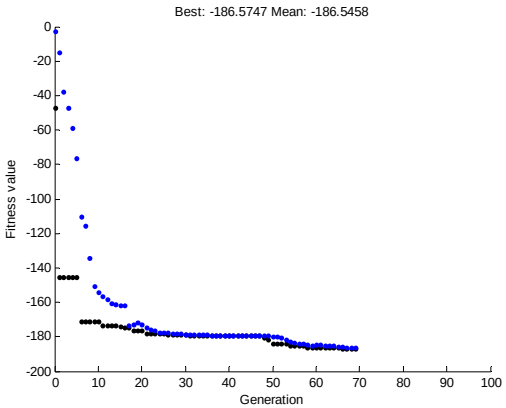


Fig.5.Fitness value between the generations for two way traffic (R_1G_2)

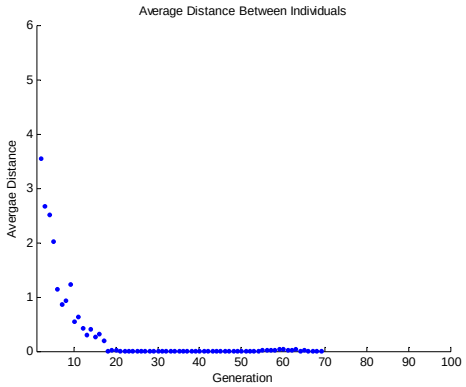


Fig.6.Average distance between individuals for two way traffic(R_1G_2)

From the above graphs, it looks that the average distance between individuals are substantial little. It shows that -186.5747 is the best fitness value for the given data.

III.CONCLUSION

It seems that the traffic pollution depends upon the traffic signal. If we control and regularize the signal may be avoid the traffic pollution. From this paper, it shows that the average distance between individuals are very little. In genetic algorithms, encoding the solution to the chromosomes and equate the corresponding fitness of the solutions is the fitness value for the given data. The probability value for the above is waiting time for the each signal light in two way direction. Two way traffic is the best solution to reduce the traffic pollution in our country according to the time set in the automatic signals.

References

[1] Ajmone Marsan.M,"Stochastic Petri nets:An Elementary Introduction", Advances in Petri nets 1989, Lecture Notes on Computer science, pp 1-29, Springer Verlag,1989.

[2] David E.Goldberg:Genetic Algorithms in search,optimization and machine learning.Pearson Education Inc.(2002)

[3] Yu,X-H.and W.W.Recker,"Stochastic adaptive control model for traffic signal systems",Transportation Research,Part C:Emerging Technologies,14,pp.263-282.

[4] Recker, W.W., B.V. Ramanathan, X-H Yu, and M.G. McNally . Markovian Real-Time Adaptive Control of Signal Systems. Mathematical and Computer Modelling Journal, (1995),22, 4-7, pp. 355-375.

[5] Nagarajan.D,Nagarajan.V,Sunitha.p and seethaleskmi.v :Analysis of Birth weight using Singular Value Decomposition, International Journal of Computer Science and Information Security (2010), Vol. 7, No. 1.

[6] Nagarajan.D., Sunitha.P., Ranking for reordering documents using latent semantic indexing Internationalconference on Advances in modeling,Optimizationand computing.IIT Roorkee 2011.PP 1074-1077.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

An Evaluative Model of Organizational Architecture by the use of Colored Petri Networks

1- Ali akbar tabibi 2-Seyed javadmirabedini

1- Research and Science University of Bushehr, Student Masters , Department of Computer Engineering, Soft ware ,7514963161 Bushehr Iran ,

2- Islamic Azad University, Central Tehran Branch, Tehran Iran,

Abstract

Organizational architecture is composed under a process called *organizational architecture process*. This process is complicated and architecture can use its framework as a modulator of structure to control complicacy and apply the method as a behavior director. In architecture, behavior is prior to structure, and a structure may have different behaviors. But which behavior (method) best suite architecture and thus meet the concerned needs? Evaluation of architecture is needed to answer this question.

As an instance, this article aims to demonstrate validity of architecture behavior on intelligent fuel card using colored Petri networks. As result, it revealed up that the given solution led to identify traffic points and thus helped the architecture designers in choosing the right method.

Keywords: Organizational Architecture, Evaluation of Architecture, Colored Petri

1- Introduction

Development of IT, rapid environmental changes and the need of organizations to harmonize with the competitive market have made organizations to constantly implement techniques to accommodate to new conditions. Many procedures are introduced in the realm of IT to overcome this problem; among which, Organizational architecture is one of the newest and the most applicable ones. In fact, organizational architecture _or IT architecture of the organization_ is derived from information architecture. Organizational architecture is an extensive attitude towards assignment and organizational tasks, work processes, informational entities, communicational networks and organizational hierarchies which are the purposes of competent integrated informational systems. In fact, organizational architecture is the same as informational systems architecture. The only difference is that organizational architecture also concerns other aspects of informational systems, such as: users, location and geographic distribution of systems, work processes, tasks scheduling, tasks motivation, organizational assignments and strategies, etc. The main purpose of organizational architecture is to switch IT as a tool into an organizational resource beside other resources _such as human resources, financial resources, knowledge resources, experiences, etc._in order to serve in organizational assignments and return its costs as well.

2- Problem Declaration

There are various methods, each affecting qualitative and quantitative variables as well as altering architecture behavior. Now, which method should the architect implement during architecture designing?! And how

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

much would implementation of a specific method on architecture fulfill the concerned various needs?! In the second phase of organizational architecture process, architect implements a method on architecture; and the final results of this method implementation would show up in the third phase. If the architect had made a mistake in previous phase or had not meet the concerned needs, then the organization have expended a lot of time and money and would cost a lot for it to change the architecture. Thus, evaluation of behavior and competency of method taken by architect is required here.

3- Literature

UML charts are turned into Petri Networks in Saldhana&Elkoutbi. In Levis, C4ISR architecture products that are generated through an object-oriented approach are changed into colored Petri Networks in order to make an executable model. In Kamandi, turning UML charts into object-oriented random activity networks is examined. Evaluation of software architectures based on colored Petri networks is reviewed in Fukuzawa. Also, using Archimate model, Lacob has studied evaluation of architecture with regard to competency.

4- Conversion Algorithm

1. Making a public notification group using all pages in classes of class chart
2. Building a hierarchical Petri network

i. Making a substitution transition for any interacting class in class graph

ii. Creating a place for any associative class and for any aggregation class and relating an appropriate color set to that place.

iii. Creating the arcs between any substitution transition and places by using activity graph. There should be a one-by-one relationship between associative relations and substitution transitions in practicable graph.

iv. Creating a substitution transition "reaching" for simulating "reaching" of any factor, applicant of services in system.

v. Creating a place with kind of data list for providing entry line of request, proportionate to any "reaching" substitution transition.

vi. Creating arc of "reaching" transition to place (line), equivalent to that and vice versa.

vii. Creating arc of place (line) to the equivalent factor of its applicant and vice versa.

viii. Adding writing of arc on arcs ended or resulted from places (line) for the purpose of adding to the end of line and taking from the beginning of the line.

ix. Creating a subpage for any substitution transition (except for "reaching" substitution transition)

a. Creating a transition for any operation.

b. Attributing entrance, exit, entrance/ exit for ports of places.

c. Creating arcs based on activity graph.

d. Attributing kardinality degree according to the estimation for creating or utilizing number ofbeads.

e. Adding writings of arcs, guard functions of code parts and the rules related to any operation.

f. Attributing estimated time for any transition (operation)

x. Creating a subpage for any "reaching" substitution transition.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

a. Creating a transition (reaching) for creating beads with corresponding element characteristics.

b. Creating a place for maintaining beads with bead-time for specifying activation timeof next transition.

c. Creating a line-place for storing the beads with corresponding element characteristics.

d. Creating arcs in a manner of going and coming with two possible places.

e. Attributing writings of arcs between "reaching" transition and "next" place with rate of reaching factor.

f. Attributing arc writing on arcs ended or resulted from place of line for the purpose of adding to lineend.

xi. Specifying Initial markings for any place, specifying aggregation classes.

xii. Specifying initial marking for any place of line as an empty list.

3. End.

5- Case Study (intelligent fuel card)

Problem Description: We assume that Petroleum Products Distribution Company (PPDC) has decided to register intelligent fuel cards in order to speedup fuel purchase in fuel stations and to improve services given to the costumers. To do this, a financial institute recommended by PPDC must give service to support credit for fuel purchase. This way, any driver would request a credit card number (i.e. account number) from the financial institute to purchase fuel; and the institute would allocate a number to the driver after the legal process has passed.After taking and submission of credit card number to the company, a code is allocated to the driver. The code would be electronically burned onto an intelligent card which is then delivered to the driver to purchase fuel via that.

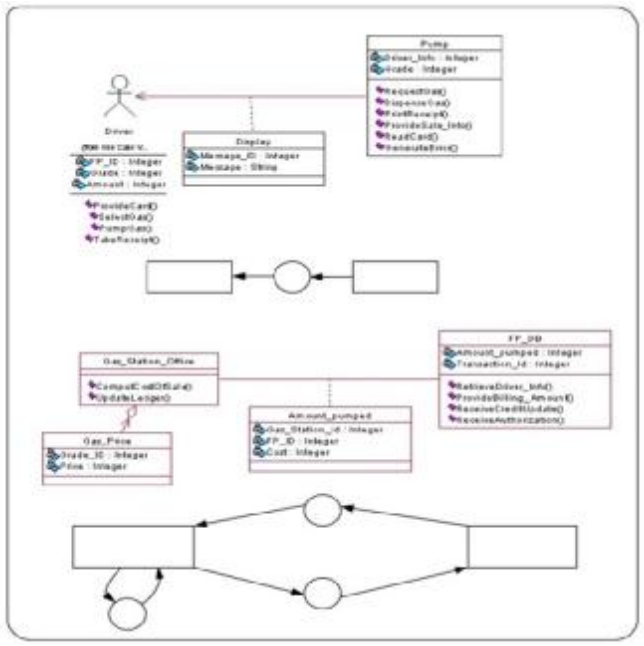


Figure 1:Transfers , Places , Arcs

After generation of arcs, places replacing collective classes can be initially valued and variables can be put into them to implement the model.

To perform this model, we must act according to active agents in the system (those that practice workload into the system); e.g. the driver in the case of intelligent fuel card. Thus, we introduce factors matching the agent (based on workload determination) and put them in the queue to be performed in the model and stimulate the system. After generating high-ranked plane in performable model, it is time to generate sub-planes for each replacing transition. For each replacing transition, there would be a sub-plane with the same title; and as far as each transition contains a specific operation, there would be a separate sub-plane generated for each operation.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

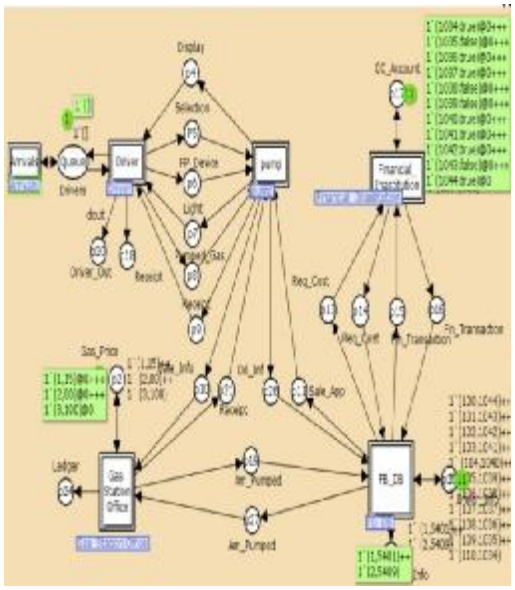


Figure 2 : General view in intelligent fuel card by using colored petri net

We arc from input and output to each transition in sub-planes. Also a specific feature will be generated for each class based on its color and properties. For example, because the pump class has six operations, its sub-plane includes six operations. After generation of locations, color codes are allocated to them and then we arc between transitions and locations. The combination of activity diagram and each location rules is used to draw arcs in sub-planes. For example, activity diagram of the pump class in "Card Read" operation receives an input message as "Drive_Info". Then R-D compares D-I information with database and allows the changes to be saved if the information were confirmed. Thus, we arc from location FP-Card to P26 _which is the output gate, and another arc to P28 _which determines pump feature (driver information). Another task of activity diagram is testing and evaluation of Approval feature, which determines if the operation is done right. For example, we arc from "Sales Approval" to "Request Selection", and another arc to "Generate Error". The information received from R-S is compared in S-E; if the information is right, the operation is run. (Fire) and [#2(app)=true] value sits into R-S; but if incorrect, [#2(app)=false] sits into G-E. After all arcs are generated, it is time to label the arcs; these labels are the defined variables form color groups of collective classes. Also, as far as number of pumps is assumed to be 10, thus FP-DB=10; and each pump time is calculated separately as shown in figure 3.

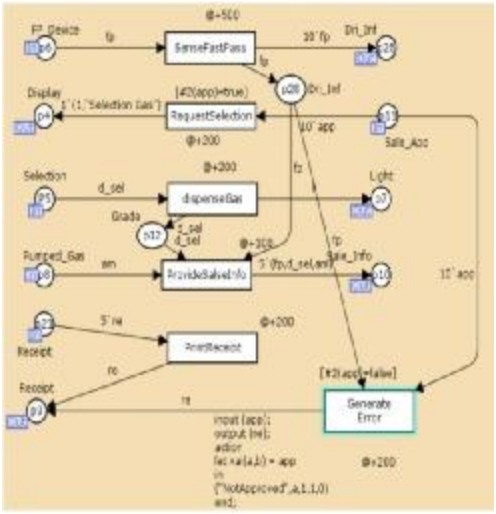


Figure 3 : Under pumps layer

5-1- The Time Spent in Queue

As indicated in figure 4, the time spent in queue is graphed with regard to entry time. As you see, the time spent in queue is almost constant and doesn't vary with the time increased. The average time spent in queue is equal to W_q=32.8

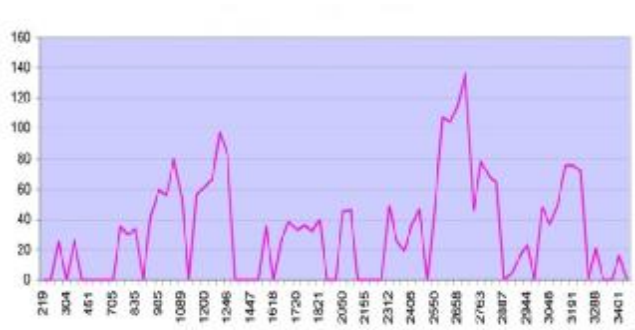


Figure 4 : Chart of wasted time in queue example by intelligent fuel card by twice pumps

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

5-2- Response Time

The response time of the system is equal to the exit time (form system) minus the entry time (to the queue). In fact, this is the total time spent in the system. As illustrated in figure 5, the response time of the system doesn't increase by the lapse of time; that means that no traffic ban would occur in the system and that the drivers could take fueling services without spending a long time in the queue. Using two pumps, the average response time equals to W=99.

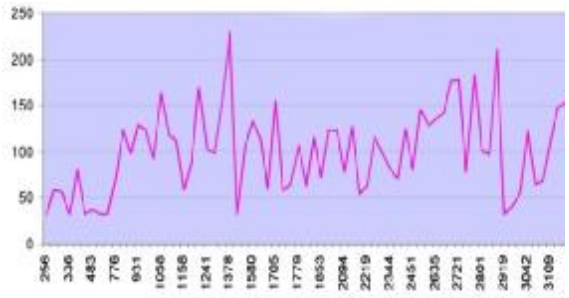


Figure 5 : Chart of answer time example by intelligent fuel card by twice pumps

5-3- Servicing Time

The time of fueling is variable. It depends on whether or not the card has enough credit; and also on the amount of fuel the driver bids. The minimum possible time (i.e. when the driver doesn't have enough credit for fueling) is assumed 3485 ms in this example. The average servicing time is about 60 seconds. Considering the point that there are two service providers, the average servicing time would be half the above mentioned; thus, it varies from about 27 to 36 seconds. This variety is because of different fueling times for different vehicles. But, whereas this time is below 40 second, there would be no traffic ban in fuel station entrance.

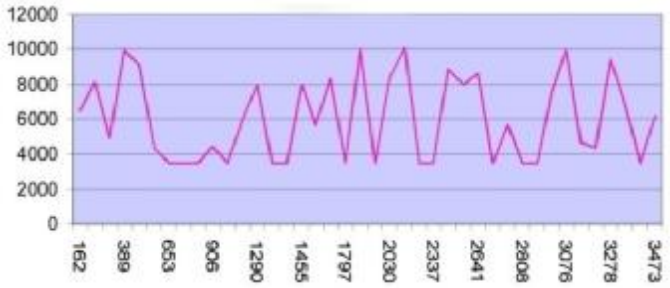


Figure 6 : Chart of service time example by intelligent fuel card by twice pumps

5-4- Queue Length

As indicated in figure 7, using two filling pumps, traffic ban in fuel station entrance may be avoided and thus queue length can be controlled in order that it won't be too long. Average queue length using two pumps is equal to 1.09 (L_q=1.09). This means a new driver in the queue would meet an average queue of 1 in length.

Figure 7 : Chart of queue lengths example by intelligent fuel card by twice pumps

6- Precise Results

6-1- Proposing an algorithm to switch architecture outcomes into a proper conductible model to evaluate architecture competency and behavior

Applying the algorithm given in this research, a conductible model using colored Petri networks may be generated that can help the architect in evaluation of concerned architecture competency and behavior.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

6-2- Helping the designers in choosing the proper method in architecture

The conductible model given in this research can help in choosing or modifying architecture methods. Thus, using the framework and method, the designer provides and composes the architecture. As far as framework defines the structure and method of architecture, and also that behavior follows the structure, several various behaviors can be taken by a single structure; and the architect may doubt about the choice of method. Therefore, the architect may be unable to answer the question of how much the architecture behavior would meet the concerned requirements.

6-2-Determination of points or resources capable of traffic-banning

Implementing the mentioned conductible model, traffic-banning points in the system (organization) may be determined. The points where servicing load rate is grater than servicing rate are more capable of traffic-banning. If this difference is too great or persists, traffic-banning emerges and may be considered a problem that the architect must providea solution for.

7- References

[1]Juric, M., and Pant. K. Business Process Driven SOA Using BPMN and BPEL, Packt Publishing 2008

[2] Khoshafian, S. service Oriented Enterprises, Auerbach 2008

[3] Erl, T.SOA. Principles of Service Design. Prentice Hall 2008

[4] Service Oriented Architecture (SOA) in the Real World. Microsoft Publication 2007

[5] Eziran , Implementation of the Architecture c4isr,May 2005.

[6] Akhavanniaki ,anoshirvan, Monthly development and use of information and communication technology (TAKFA),July 2005.

[7] Shams , Fereidon . Basic concepts of Enterprise ArchitectureMonthly,development and use of information and communication technology (TAKFA) Second year (3), July 2004

[8] M. P. Bienvenu, I. Shin, and A. H. Levis, .C4ISR Architectures III: An Object-Oriented Approach for Architecture Design,. Systems Engineering, Vol. 3, No. 4, Fall 2000

[9]M.Elkoutbi and R.F.Keller. .Modeling Interactive Systems with Hierarchical Colored Petri Nets..Proc. of the Conference on High Performance Computing, 1998.April 6-9, Boston.

[10]Fatolahi, A., and Jalalinia, S. 2004, .Building Transition Architectures Using Zachman Framework., Proc. of the 1st Int'l Conf. of Industrial Engineering, Tehran, Iran.

[11]Fukuzawa, K. AND Saeki, M. 2002. Evaluating Software Architectures by Coloured Petri Nets. ACM SEKE .2002, July 15-19

[9]Korotikov, V. 2003. Using of Petri Net Techniques in Information Systems Design. IEEE 4th siberianrussian workshop and tutorials EDM.2003

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

Security Issues Analysis for cloud computing

Bashir Alam[1], M.N Doja[1], Mansaf Alam[2] and Shweta Malhotra[1]
[1]Department of Computer Engineering Jamia Millia Islamia , New Delhi, India
[2]Department of Computer Science Jamia Millia Islamia , New Delhi, India

***Abstract*— Cloud computing is a new emerging concept recently introduced in the world. Cloud services on the first hand provides many advantages like pay-as-u-go nature, faster deployment of IT resources and the way of future but on the other hand challenges/ issues of cloud overweight the advantages of cloud. Among all the challenges of cloud, the upmost challenge that the world is facing with cloud is "Security" as clients outsource their personal, sensitive data to the cloud over the internet which can be very dangerous if not secured properly. In this paper we have analyzed security issues of cloud from different aspects along with some implemented solutions. Security of cloud can be categorized by service models provided by service providers, data life cycle security issues and it can be categorized by data security, virtualization security and software/application security. We have also analyzed some implemented solution model based on cryptography and shamir's secret sharing algorithm to some of the security issues.**

Keywords- Software as a service (SAAS) Platform as a service (PAAS); Infrastructure as a service (IAAS); Service level agreement (SLA), Multi cloud Database model (MCDB), NetDB2-Multi Share(NetDB2-MS).

I. INTRODUCTION

A. *Cloud Computing*

The most widely used definition made by National Institute of Standard and Technology (NIST) for cloud computing says that Cloud computing is a model that enable convenient network access on demand to a shared pool of configurable computing resources (e.g. networks, servers, storage, applications and services) that can be rapidly provisioned and released with minimal management effort or service provider's interaction[1]. It has five essential characteristics, three service model and four deployment model. Essential Cloud Characteristics includes (i)on demand self-service (ii) elasticity (iii) scalability (iv) measured service and (v) multitenancy . Its three service models are (i) Software as service (ii) Platform as a service and (iii)Infrastructure as a service. Its deployment models are (i) public (ii) private (iii)hybrid (iv) community. As per IDC, the total cloud services market will grow from $884.4 million in 2012 to $2,671.9 million in 2017 at a compound annual growth rate of 24.7 percent [12].

B. *Cloud Challenges*

There are many challenges of the cloud computing like Availability of a service, Security, Shared Nothing Architecture, application parallelism, Interoperability etc. [4]. Security is one of the important issues that should be considered primarily and being taken care of.

As per IDC survey on "IDC cloud services survey: top benefits and challenges" in November 2009, conducted by IDC Enterprise panel by IT executives and their Line of Business Colleagues, it has been concluded that 87 percent point out the security concern, 83 percent point out availability concern, 82 percent point out performance concern, 80 percent point out lack of interoperability concern. As it can be seen in Fig. 1, among all the challenges like availability, performance, lack of interoperability, security is of major concern [13].

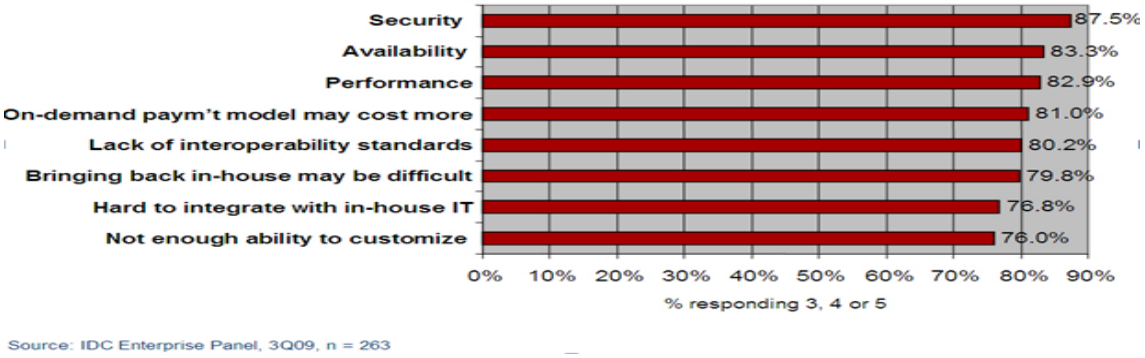


Figure 1: IDC Cloud Services Survey [13].

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

The paper is summarized as follows: Section II describes various cloud security issues from different aspects and also implemented solution models to some of the security concerns. Section III summarizes these issues and the implemented solutions in tabular form. Section IV outlines the conclusion.

II. CLOUD SECURITY ISSUES FROM DIFFERENT ASPECTS

Cloud security issues are categorized by many ways. In this section we have analyzed various security issues from different aspects. Formerly security issues like Privileged user access, Regulatory compliance, location of Data, segregation of Data, Recovery, Investigative support, Long term viability were enlisted for the SLA [2]. Cloud service provider provides different kind of services that includes software as a service (SAAS), Infrastructure as a service (IAAS), Platform as a service (PAAS), according to S. Subashini, V. Kavitha [3] the security issues related to these service models were analyzed. Pengfei you et al. [8] described the cloud security issues related to Data security, virtualization security and application security. Data is particularly important to the cloud, Deyan Chen and Hong Zhao [9] described the cloud issues with respect to Data cycle related issues in [9].

In this paper analysis of some of the implemented solution models provided for some of the security issues has also been done. Cryptography keys and poilicy related solution model has been described by Yang Tang, john C.S. Lui [11], the solution model based on shamir's secret sharing algorithms and multi clouds are described in [5][6][7][10]. Different security issues from different aspects are described as follows:

A. SLA Cloud Security Issues

B. Reddy Kandukuri et al [2] described the meaning of SLA (Service Level Agreement). SLA is the legal agreement between the two communicating parties mainly Client and service provider. Firstly they explained about the contents of typical SLA like it must include what will be the service measuring parameters, what will be done in case of any disaster or problem occurs within the system, it includes what are the customer duties and responsibilities and it must also include how the termination of services takes place. Authors suggested some security issues that should also be included in typical SLA. According to them a Standardized SLA must include Privileged user access, Regulatory compliance, location of Data, segregation of Data, Recovery, Investigative support, Long term viability as shown in Fig 2.

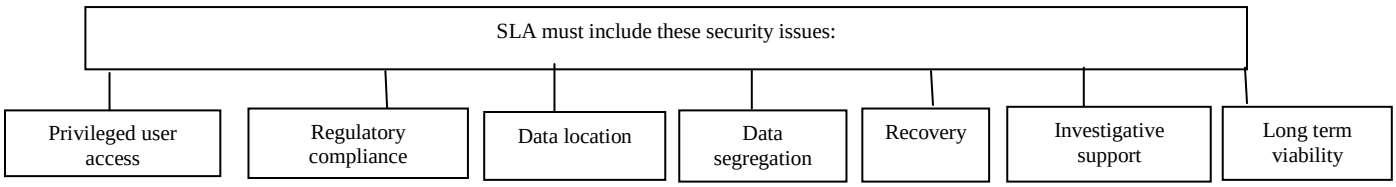


Figure 2: SLA includes these cloud security issues: Privileged user access, Regulatory compliance, Data location, Data segregation, Recovery, Investigative support, Long term viability.

Cloud provider must go through security certifications i.e. regulatory compliance. According to them data location is where the data is stored as customers do not know where there data is getting stored and how the data is being processed in cloud, cloud provider must obey some legal privacy laws requirement of customer's data. They explained data segregation as: cloud consists of many customers, it contains data of many clients at the same storage place so it must include some boundaries they suggested encryption as one of the solution for data segregation. From their point of view, recovery is a method by which in case of disaster clients data can be recovered it includes replication of data, data backup of data. At last they accounted that the cloud services are mandatory for long term viability.

B. Cloud Service Model related Security issues

S. Subashini, V. Kavitha [3] analyzed different security issues related to nature of service delivery models of cloud computing as shown in the figure 3. According to them, the main security issues related to the software as a service models are as follows: Security of Data, Security of Network, Locality of Data, Integrity of Data, Data segregation, Data Access, Authentication and authorization, Data Confidentiality, Data breaches, Virtualization vulnerability, backup of data. They explained data security as Security of client's personnel sensitive data on the cloud, so cloud provider must provide additional security features apart from default one that is used in traditional systems; it involves strong encryption techniques for data security.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

They described network security as securing communication over the internet as all the communication between cloud service provider and customer is done through internet, it involves the use of strong network traffic encryption techniques such as Transport layer security, secure socket layer. Data locality, Data segregation and data recovery security issues are already expressed by B. Reddy Kandukuri et al [2].

According to them, Data integrity means that protecting data from any unauthorized deletion, modification and prevarication, maintaining data integrity become very difficult because of the fact that most of the web applications do not support transaction management and transactions should ensure that data should follow ACID property. They illustrate Data Access as only authorized parties can access the outsourced data on the cloud, depending upon cloud deployment and service models, specified users must firstly be established and predefined access properties and permissions should be granted accordingly. They deliberated Data Confidentiality as to ensure that user data which resides on the cloud cannot be accessed by unauthorized party, Confidentiality can be achieved through proper encryption techniques along with the proper key management. Another solution for confidentiality is to split up attributes between several data servers using customized threshold secret sharing scheme. Data breaches, according to them are possible due to two types of attacks: insider attack or some outsider external factor like hackers can cause the attack. Lastly they accounted the most important issue of cloud computing i.e. Virtualization vulnerability, the main problem occurs in virtualization is to isolate the different instances of VMs from each other.

They summarized the security issues of platform as a service (PAAS) as Host and network intrusion prevention: which means data should remain inaccessible between applications as, in PAAS, customers build their own applications on cloud platform provided by cloud provider. Here in PAAS hackers can attack the visible code of application, they can attack infrastructure. In PAAS Vulnerabilities are not limited to the web applications but also with machine to machine service oriented architecture.

Lastly, they discussed about the Security issues related to Infrastructure as a service (IAAS) model, in IAAS the provider providers the artificial developing environment like virtual machine, storage, network bandwidth etc. According to them the main security issue is related to virtualization and there are many security related problems like reliability of data that is stored within the provider's hardware. They considered Security is the responsibility of both client and provider which differs like security up to hypervisor like physical security, environmental security, virtualization security comes under the control of provider and security of application OS that comes under the control of customers.

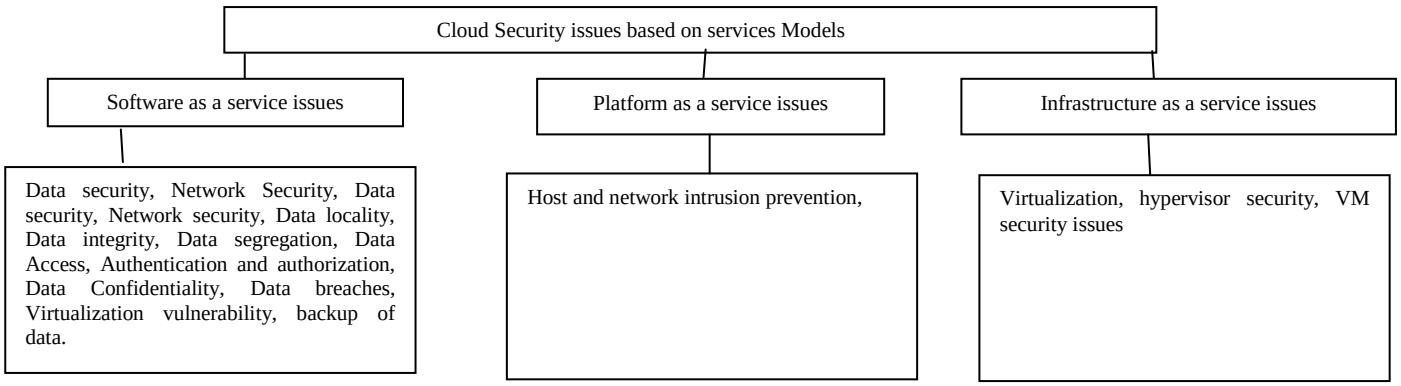


Figure 3: Model of Cloud Security based on services models

C. Cloud data, virtualization, and application related security issues

Above paper discussed the security related issues of service models provided by cloud computing. Pengfei you et al. [8], analyzed the cloud computing security issues with respect to three aspects i.e. Data Security, Application security and virtualization security as can be seen in the Fig. 4 and they also gave some current solutions for these issues. Firstly, they explained the following Data Security related issues as Data Breach, Data Lockin, Data Remanence, Data recovery, Data locality. According to them Data Breach concerned with two security aspects of data: data integrity and data confidentiality the solution for both issues are to use the strong encryption mechanism like AES & DES under the proper management of keys. Data Lockin is another issue of migrating data from one SAAS or IAAS vendor to another vendor but while migrating, data may get lost. Solution for this issue is to have the standardized cloud Application Programming Interface (API). Data recovery and Data locality security issues are already expressed by B. Reddy Kandukuri et al [2]. Lastly Data Remanence issue explained as Data is not permanently erased after deletion so malicious hacker can extract the sensitive data which could be very dangerous. Possible solution for this issue is to encrypt the data along with the proper key management.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

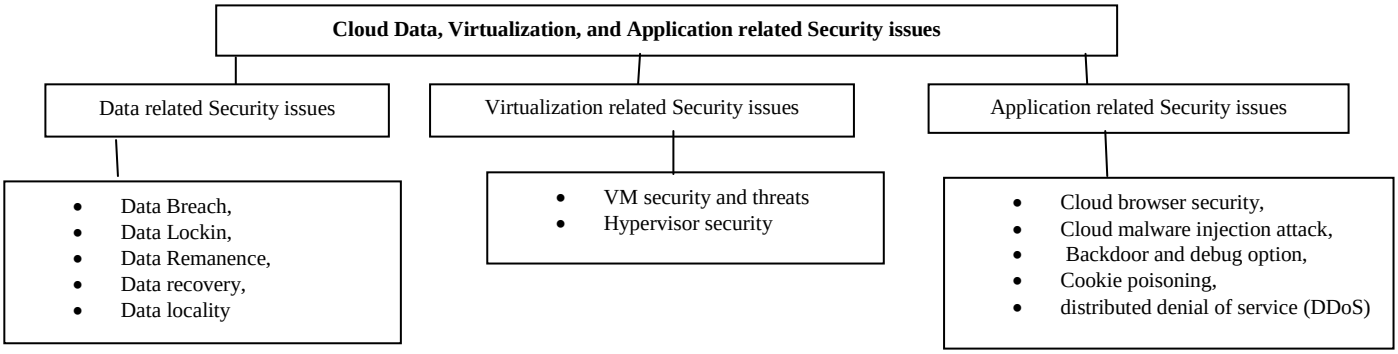


Figure 4: Cloud Security issues related to Data, Virtualization and Application

Then they discussed very important aspect of cloud i.e. Virtualization and virtualization related security issues. Virtualization means on a single physical machine, number of virtual machines can be possible with the help of abstraction of computing resources. A physical machine is composed of operating system and hardware similarly Virtual machine is also composed of operating system but here OS is known as guest operating system. A hypervisor or Virtual machine monitor which is additional layer between hardware and operating system is required to coordinate the multiple VMs on the single physical machine. The main security issue here is VM security and threats which mean there must be a clear cut boundary between VMs. Users are responsible for their VMs, updating, patching the operating systems and all the software's. So it's very easy for hacker to attack guest OS and take control on others VM and then from that VM to the others. The solution for this problem is that users should update and patch guest OS and software. Other security issue related to virtualization is Hypervisor security which means if intrusion takes control on hypervisor it can do anything to any VM, because hypervisor is the system that coordinates all the functionalities of VM. The whole system can go down. Solution is that always update hypervisor product and other virtualization products.

Then lastly they analyzed Application related issues as Cloud browser security, Cloud malware injection attack, Backdoor and debug option, Cookie poisoning, distributed denial of service (DDoS). The main reason for the loopholes in application security is the network security. Any unauthorized user can impersonate as authorized user and access the assigned IP address. They explained Cloud browser security securing browser, Clients whole data is processed on the cloud servers. Clients communicate to the cloud with the help of browser so browser security is must. Traditional solutions are not very secure like Transport layer security which provides host authentication and data encryption. One proposed solution is to have TLS and at the same time XML based cryptography for the browser core. Then they discussed about Cloud malware injection attack i.e. Injecting Malicious VM or service implementation module for the purpose of blocking or modification of data to change the entire system functionality. The solution is to check the integrity of every service instance request with the image hash value of original service instance. Backdoor and debug option security issue was defined as sometimes developers write some backdoor code and debug options for their convenience like revising the website again. But this can lead to some entry point for hackers and then accessing the useful information. There should be a proper care at the development time. They illustrated Cookie poisoning attack and explained cookies maintain the data, which allow any unauthorized user to impersonate an authorized user. Solution to this problem is to delete or encrypt cookies so that no one except authorized users if needed can see the cookies. Lastly in the list of application attack is Distributed denial of service (DDoS) i.e. Attacker flood the entire network with packets that cannot be processed by the system.

On the victim machine the cloud computing operating system gives more resources like more VM's to tackle this situation but when the workload become more the services that the system can provide becomes unavailable, Which is very dangerous for the reputation of cloud computing as availability is the main feature of cloud. The solution they gave for this problem is to have the intrusion detection system which checks all the incoming and outgoing data, also IDS can also be applied on physical machines which having VM concept.

D. Cloud Data Cycle Issues Related Security

Above papers [2][3][8] explained about the various types of security issues related to cloud computing. Deyan Chen and Hong Zhao [9] analyzed the data security and privacy issues associated with cloud computing across all the stages of data life cycle as shown in the Fig. 5, also discussed about some current solutions to this life cycle. Firstly they described the seven phases of data

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

life cycle then they discussed about the security issues with respect to phases of data life cycle. According to them the first phase of data life cycle is Data generation which is explained as the owner generates the data. Generally, Organizations manage their data but in cloud it should be considered how to maintain the data ownership, so what type of information is being collected is totally under control of data owner. Data owner have full control on their personal information. Then second phase they explained as Data transfer which is during data transmission between different platforms (among different enterprises, customers) data confidentiality and integrity should be ensured for that various encryption techniques should be applied along with different transport protocols for providing network security. Third phase is Data use which is illustrated as, Indexing and query problems are related to encryption, so cloud based applications generally are not encrypted. But this causes a serious problem because cloud is multitenant in nature. Many users store their data on the same computing resource so any intruder can easily see the others data with bad intensions. Then they defined forth phase as Data share which means while sharing the data with the others its original protection measure and usage restrictions should be maintained. During sharing of personal information with the third party its authorization, granularity and data transformation need to be considered. Here data transformation means that sensitive data need more cure than the other data. Fifth and important phase they expressed as Data storage means storing the data. The data that is stored on the cloud should meet the following three security aspects: Data Confidentiality, Data integrity and availability. For data confidentiality proper encryption techniques along with the proper key management should be used but the problem with the encryption techniques are it is a very time consuming process and key management is also a big problem. Somewhat similar to data storage they expressed sixth phase as Data Archival which mainly deals with the storage media of data, a place where the data is stored. It should provide easily availability and should not leak data. Last phase of data life cycle they defined as Data Destruction the issue related to this phase is explained as when the data is deleted it is not properly eliminated from the storage devices because of the characteristics of storage media. It could lead to damage the sensitive information.

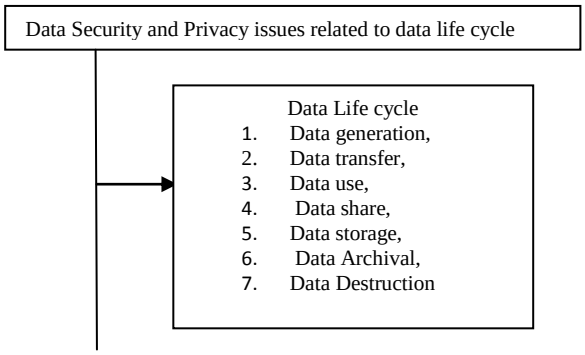


Figure 5: Data Security and Privacy issues related to Data Life Cycle

E. FADE Implemented Solution for Proper Access and Assured Deletion Issues

Papers in [2][3][8][9], discussed about the security issues related to cloud and suggested some of the general solutions to those issues. One security solution is implemented in[11] for two main security problems. First is Access Control which means that only authorized parties can access the outsourced data on the cloud and the second is assured deletion which means once the data is deleted it cannot be accessible to anybody including data owners also.

They designed and implemented system named FADE which is an overlay over the cloud storage to provide the policy based access control and assured deletion. Definition of policy varies for applications. They defined policies with the help of examples and associate this policy with all the files. According to them, FADE system is composed of two main entities as shown in the figure 6, first is FADE Client which applies encryption and decryption to the data files upload to (download from) cloud and second is Key mangers which are the quorum of key managers based on shamir's threshold secret sharing method, they maintain policy based keys for access control and deletion.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

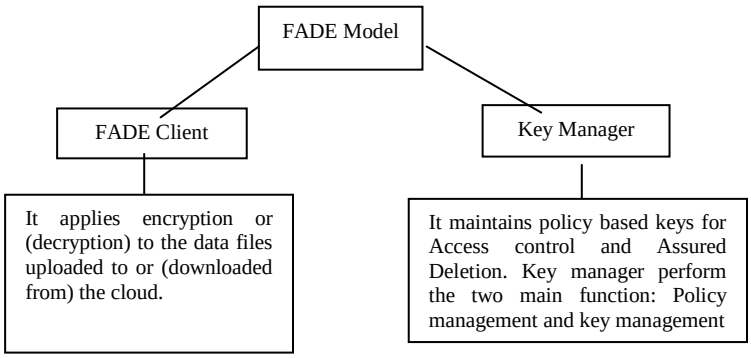


Figure 6: FADE Two Systems

With the help of three cryptographic keys Data key, control key and access key: FADE provide the basic operations of file uploading/ downloading. Figure 7 shows the file uploading operation and Figure 8 shows the file downloading operation for FADE. Authors explained file uploading operation as, if the client wants to upload file to the cloud, the client first request to the key manager about the public control key (n_i and e_i) for the specific policy p_i and client will retain this value for subsequent uses for the policy p_i . Then the client will produce two random keys K, S_i, and on the basis of these keys client encrypt his data file. Then the client sends encrypted file data along with metadata $\{K\}_{Si}$, S_i^{ei}, $\{F\}_K$, p_i to the cloud. To protect the integrity of data the client computes the HMAC signature on every encrypted file and stores this HMAC signature together with the encrypted file in the cloud. The author explained the file downloading operation as follows: the client obtain $[\{K\}_{Si}, S_i^{ei}, \{F\}_K]$ file data and metadata from the cloud. Firstly client calculate the HMAC signature of the data before decrypting the file to see whether this is the original data or not. Then client generate a secret random number R. With the help of public key e_i, client encrypt R and then transfer $S_i^{ei}.R^{ei}$ to the key manager to request for decryption because the key manager is having the private control key for e_i i.e. d_i. Key manager will return the product (S_i.R) to the client. Now client remove the S_i from R and decrypt $\{K\}_{Si}$ and hence $\{F\}_K$

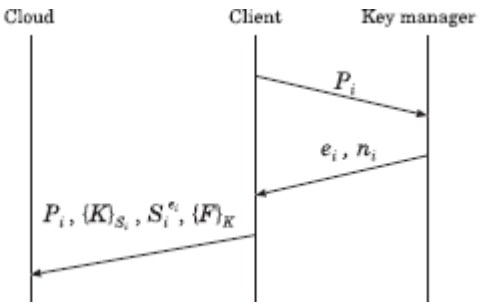


Figure 7: File uploading in FADE [11].

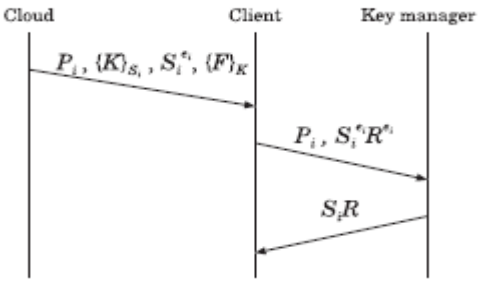


Figure 8: File downloading in FADE [11].

Authors also described the policy revocation and policy renewal operations and explained the file uploading/downloading operations with multiple policies. Then authors also discussed the two extensions to the basic FADE deign to provide more security. First is the use of "attribute based encryption" in order to authenticate clients through policy based access control and second is "threshold secret sharing method" in order to achieve better reliability for key management.

Authors implemented FADE system in C++ on LINUX and they used Amazon S3 as the cloud storage backend. They evaluated the performance of FADE in terms of running time and monetary cost. They figured out the time performance of basic design for file uploading /downloading, policy renewal operations for single policy as well as multiple policies. Authors also evaluated the performance of extended version of FADE, which involves attribute based encryption and threshold secret sharing method for the file upload/download operation. Then authors calculated the monetary overhead using simple pricing model the simple scheme of Amazon S3 for three conjunctive policies and the three key managers. Finally authors concluded that FADE system provides additional level of security for cloud.

F. MCDB Implemented Solution for Data Confidentiality, Data Integrity, Data Availability Issue

Yang Tang, john C.S. Lui [11] implemented one model "FADE" based on policies and cryptographic keys for two security issues access control and assured deletion. Authors in [5][6][7][10] proposed models for data security and privacy issues like data

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

integrity, data intrusion, data confidentiality, data availability. Authors in [5][6][7][10], first expressed the disadvantage of using encryption technique to provide data security in cloud as encrypting data involves overhead of cost in terms of processing and storage. They proposed models like MCDB (multi cloud database model), NetDB2-MS based on Shamir's threshold secret sharing techniques.

According to them shamir's secret sharing is a method of dividing data D into (n) pieces($D_1 \ldots D_n$) in such a way that knowledge of any k or more D_i pieces make the value of D known. Therefore a complete knowledge of (k-1) pieces reveals no information about D, k should be less than n to keep the value of shares un-constructible and ensure the adversary cannot access k pieces. This method is a solution for privacy issues. Authors in [7] proposed MCDB model which is based on multi clouds service providers instead of single service provider. The main purpose of moving to multi cloud is to improve what was offered in single cloud by distributing the reliability, trust and the security among multi cloud providers. They defined the data flow model of multi clouds with the help of two procedures as can be seen in Fig. 9. Firstly they described the sending data procedure in which a user sends its query from web browser or user interface to HTTP server through HTTP request, then it goes to servlet engine through application request, after that the communication between servlet engine and DBMS is done by a JDBC protocol. Secondly they described procedure between DBMS and Cloud service providers (CSP) after receiving the data DBMS divides into n shares and stores each share into different CSPs.

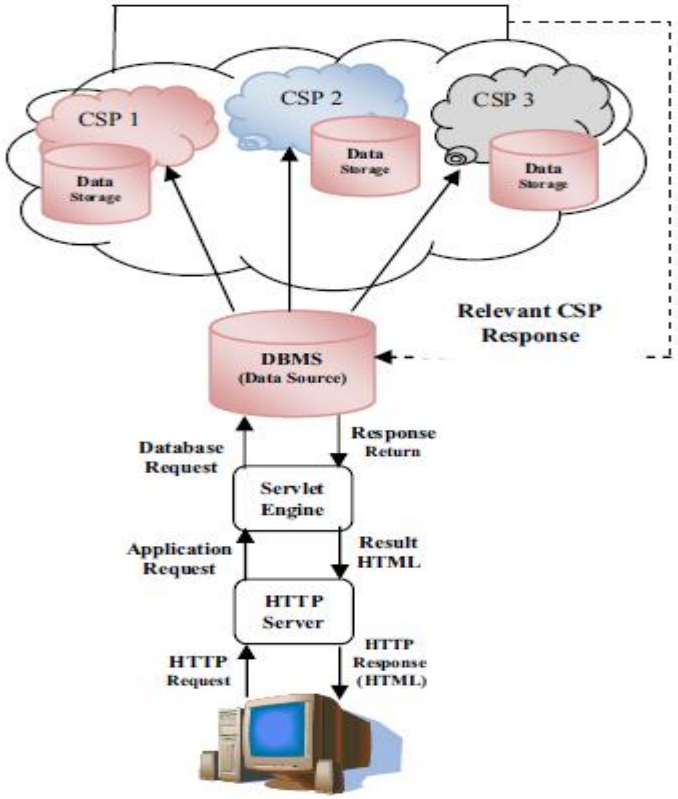


Figure 9. Overview of Multi-clouds [7].

After that DBMS generates a random polynomial function in the same degree for each value of the valuable attribute that the client wants to hide from untrusted cloud provider. These polynomials are not stored at the data source but are generated at the front and at the end of query processing like when any query comes to DBMS. DBMS rewrite the query for each CSP and it will generate the polynomial and then send shares to CSP_i. After that CSPs return the relevant share to the data source. They also defined the three layers of MCDB model. First layer, they named as Presentation layer which includes the components user and HTTP server. Application layer, the second layer which includes servlet engine and lastly, Management layer includes components DBMS, Cloud service providers. Authors compared their model with single cloud model (Amazon) for data retrieval and data storage operations and concluded that the multi cloud model is superior then single cloud model in addressing the security issues like data integrity, data intrusion, data availability.

Similarly, Authors in [5] proposed model NetDB2-MS (NetDB2-Multi share) based on Shamir's secret sharing method. Author compared the secret sharing method which they used in NetDB2-MS with the Blowfish encryption technique that they were using earlier in NetDB2 for data storing and data retrieval operations and also for three queries exact matching, range and aggregate

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

query. They proved that the secret sharing method shows a significant performance improvement for data storage, data retrieval for various query types.

III. SUMMARY

Table1, summarized different security issues and implemented solution models for some particular security issues. NetDB2-MS [5] model based on shamir's threshold secret sharing algorithm is implemented for data privacy issues it shows significant improvement for data retrieval, data storage operations and three queries exact matching, range and aggregate query when compared with the earlier model NetDB2 which was based on blowfish encryption technique. For data integrity, data intrusion and data availability, MCDB [7] model (Multi Cloud Database model) based Shamir's secret sharing algorithm and TMR (triple modular redundancy) with sequential method was proposed. MCDB model also shows improvement when compared with single cloud model for the data retrieval and data storage operations. For proper access and assured deletion one FADE [11] model is implemented which is a cloud overlay over the clouds. FADE model is based on encryption technique, cryptographic keys, and policy based methods. Some authors have given general solutions for security issues such as to use encryption, transport layer security to provide the security solutions.

Table 1 Security issues & implemented solutions for security issues for cloud

Year	Security issues	Models developed	Technique used for the solution for security issues
2009	Includes some security issues that should be included into typical Service Level Agreement. Issues like: User access in privileged mode , Regulatory compliance, location of Data , segregation of Data, Recovery, Investigative support, Long term viability	No	
2010	Security issues related to service models provided by cloud providers. Service models are Software as a service (SAAS), platform as a service(PAAS), infrastructure as a service(IAAS).	No	General solutions like encryption, transport layer security issue
2011	Data privacy issue	NetDB2-Multishare	Based on shamir's secret sharing algorithm. Prosposed model shows significant improvement in performance for data retrieval and storage.
2012	Data confidentiality and correctness of query that includes integrity, completeness and freshness.	Proposed approach	Based on shamir's secret sharing algorithm and redundant shares.
2011	Data integrity, Data intrusion and Data availability	MCDB model based on multi clouds	Based on shamir's secret sharing algorithm, TMR (triple modular redundancy) with sequential method.
2012	Data security issues- Data Breach, Data Lockin, Data Remanence, Data recovery, Data locality, Application security issues- Cloud browser security, Cloud malware injection attack, Backdoor and debug option, Cookie poisoning, distributed denial of service (DDoS), Virtualization security issues- VM security and threats, Hypervisor related issues.	No	General talk on solutions like use of encryption techniques etc.
2012	Data life cycle:Data generation, Data transfer, Data use, Data share, Data storage, Data Archival, Data Destruction related security issues	No	General talk on solutions like use of encryption techniques etc.
2012	Proper Access and Assured Deletion	FADE model	Based on encryption technique, cryptographic keys, policy based methods for assured deletion.

IV. CONCLUSION

We have analyzed several security issues from different aspects. First categorization is based on service models provided by service provider in which we have analyzed security issues related to Software as a service model, platform as a service model, and infrastructure as a service. Second categorization is based on the data security, application security, and virtualization security. As data is stored on cloud so the third categorization is based on data life cycle security issues. Then we have analyzed some implemented solutions for security problems. In this paper we have discussed about two implemented solution models. "FADE" model is based on cryptography and NetDB2 multishare, MCDB-multi clouds is based on shamir's shared secret sharing algorithm.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

REFERENCES

[1] Peter Mell and Tim Grance, "The NIST Cloud Computing Defn.– V. 15,10-7-09"
[2] B. Reddy Kandukuri, V. Ramakrishna, A. Rakshit, "Cloud Security Issues", IEEE International Conference on service computing(SSC 2009), 2009, e. pp. 517-520
[3] S. Subashini S, V. Kavitha, "A survey on security issues in service delivery models of cloud computing", Journal of Network and Computer Applications 34 (1), 2011, pp.1-11.
[4] Sherif Sakr, Anna Liu, Daniel M. Batista and Mohammad Alomari, "A survey of large scale data management approaches in cloud environments", IEEE communications surveys and tutorials, 13(3), 2011, pp. 311-336.
[5] M.A. Alzain and E. Paradede, "Using Multishares for ensuring privacy in Database as a service", IEEE 44th Hawaii international conference on system sciences (HICSS), 2011, pp.1-9.
[6] M.A. Alzain and E. Paradede, "A new Approach using redundancy technique to improve security in cloud computing", IEEE international conference on cyber security, ISMN 978-1-4673-1425-1, 2012 pp. 230-235.
[7] M.A. Alzain and E. Paradede, "MCDB: using Multi clouds to ensure security in cloud computing", IEEE Ninth International Conference on Dependable, Autonomic and secure computing, 2011, pp 784-791.
[8] Pengfei You, Yuxing Peng, Weidong Lui, Shoufu Xui, "Security issues and solutions in cloud computing", 32nd international conference on distributed computing systems workshops, 2012, pp 573-577.
[9] Deyan Chen, Hong Zhao, "Data security and privacy protection issues in cloud computing", International Conference on Computer science and Electronics Engineering, ICCSEE, 2012, pp. 647-651.
[10] M. A. Hadavi, M. Noferesti, R. Jalili, E. Damiani, "Database as a service: toward a unified solution for security requirement", IEEE 36th international conference on computer software and applications workshops, 2012, pp. 415-420.
[11] Yang Tang, Patrick P. C. Lee, John C. S. Lui, Radia Perlman, "Secure Overlay Cloud Storage with Access Control and Assured Deletion", IEEE Transaction on Dependable and Secure Computing, 9(6), 2012
[12] Hamish Barwick (CIO), "Cloud computing adoption increases in Australia IDC", http://www.cio.com.au/article/520964/cloud_computing_adoption_increases_australia_idc_/
[13] Frank Gens, "New IDC IT cloud services survey: Top Benefits and Challenges", http://blogs.idc.com/ie/?p=730

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

Software Engineering Framework using Agile Dynamic System Development Method for Efficient Mobile Application Development

Mr. N. Balasupramanian, Mr. RamKumar Lakshminarayanan , Dr. RD.Balaji,
Lecturer, Department of IT, Higher College of Technology, Muscat, Sultanate of Oman.
.

ABSTRACT
The mobile industry is changing the technologies very often to attract the customer to a greater extent; whether it is application platforms, devices, technology, features, network models or exploration of application use cases, the speed of change for any one of these technologies means that businesses or opportunities have to think carefully before investing in creating their own applications. Now-a-days, the mobile application development is targeted of introducing many new tools, techniques and methodologies for the application development. This paper provides the development team members a right direction to apply appropriate software engineering framework implementing agile method for the development of mobile application and this paper also gives a comparative study between the XP and DSDM agile methods.

Key Words- Going Mobile, Application Development, Software Engineering, Agile, Framework, XP-Extreme Programming, DSDM-Dynamic System Development Method

1. INTRODUCTION

Over the years, we have seen people using Mobile devices and applications almost everywhere for their daily activities to be carried out. "*Going mobile*" is another channel for business whether a person or organization will move and apply mobile technologies for their day-to-day activities. Mobile businesses or opportunities should have a clear idea about their brand and should focus on how to achieve it. "*Going Mobile*" is a strategic plan where more conceptual that are operational. Hence, this strategic plan provides businesses with significant plans for the upcoming cannot afford to ignore mobile, or become satisfied about it and take short cuts to mobile commitment.

Many companies like Face book, Twitter and others do research and development in the mobile application based on the strategic plan "*Going Mobile*" [3]. Basically these companies provide the multiplatform application. Besides web application being developed, Facebook supports four mobile applications in each platform. In the recent years, the usage of mobile phones, apps and application development platform has become more dominant and can expand fast. [1]

The mobile application platforms such as Windows phone, IOS, Android and Symbian which still exist and grow based on the arrival of new smartphones in the general market.
Based on the previous research, the mobile application development can be classified as mobile web application, native application and hybrid application [10]. Mobile web application is an internet / web enabled applications, which are accessed through the

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

mobile device's, obviously need not downloaded and installed on the mobile device. Native application is an application for a specific type of devices such as Smartphones /Tablets etc, which is to be directly installed on the device itself. Access to these type of applications, can be downloaded and installed from an online store or marketplace (The AppStore or Android Apps or Google Play).

Hybrid application is an application that runs on different platform or on different devices. The process of hybrid application is a combination of both nature and mobile web application execution. This type of application actually hosted or runs inside a native container on the mobile device. When developing a mobile application, the team must first have to choose any one classification from the existing applications classification. When the team members are new for this type of development, there arises many types of difficulties during the development of multi-channel application. Some difficulties are addressed below:

- Selecting the right application classification.
- Rather than focusing on the business statement, the team may sometime get agitated by the new technology.
- The application is created successfully by the development team, but the customer finds some complexity in using the application itself.
- Supporting multiple platforms requires maintaining multiple code bases and can result in higher costs in development, maintenance, pushing out updates, etc.

This research provides the following contributions such as recommending a quick direction for the mobile application development team to select and develop a mobile application by proposing a software engineering framework using DSDM for mobile application development.

2. BACKGROUND ON PREVIOUS RESEARCH

2.1 Mobile Application Development

The previous researches have categorized three different mobile application developments such as native application, mobile we application, and hybrid application [10]. Each category has its own merits and demerits that will be discussed here below:

Native apps are normally constructed on a specific platform by using SDK, tools and languages, which is specifically provided by the vendor(for example; iOS uses the programming language Xcode/Objective-c, Android uses the programming language Eclipse/Java, WindowsPhone uses the programming language Visual Studio/C#). Some examples of native application are **Camera+ for iOS devices and KeePassDroid** for Android devices [7]. Each mobile application development platforms namely iOS, Android are unique in the development practice, under native apps. The significant aspect of native apps is that it can interface only with the device's native features, information and hardware. The main issue here in this native application is that they are typically more expensive to develop, especially if they are supporting multiple mobile devices. For instance, developing Android app needs a Java technology and at the same time Symbian

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

also use Java, hence here both the cases are unique [7]. The problem arises only when transferring from one platform to another, even though the hardware seems to be similar, the development technology is quite unique.

Mobile application is efficient when the solution is addressing more than one platform. The web standard languages like HTML5 and CSS3 available to create a web application. [1] There are two different types of approaches applied to trial the development of mobile web application [15]. First type is based on a normal web browser running in the mobile device. Second type, is actually based on the device capability which delivers the intended mobile web application. Comparing the above two types it is clear that the first type requires more attempt than the second one. Also the first type will give good outputs than the second type particularly in terms of dissimilar mobile browser.

The hybrid application is the latest one that is opted to construct mobile applications. The main concept behind hybrid application is that it supports multiple devices and platforms, which runs inside a local container and influences the device's browser engine to provide the HTML and processes the java scripts locally. Hybrid apps can be simply stated, as web-to-native abstraction layer which enables access to device capabilities that are not accessible in the mobile web application. The application normally communicates with the backend like web services, cloud service, or any other middleware. [1]

A comparison between the hybrid, mobile web and native apps is given below: [1]

- Most of the mobile gaming application use native apps to offer more elasticity access to the hardware resources.
- Internal corporate business application uses mobile or hybrid apps which offer access elasticity between devices.
- Consumer application that applies native apps to attract many customers for the application.

The advantages and challenges [4] of the three categories of mobile application are given in the below table 1:

	Native Apps	**Hybrid Apps**	**Mobile web Apps**
Access to device capabilities	1) Single platform affinity 2) Written with multiple platform SDKs 3) Must be written in each platform 4) Access to alternative APIs 5) Faster graphics performance 6) AppStore distribution	1) Cross platform affinity 2)Written with web technologies (HTML5, CSS3, Javascript) 3) Runs locally on the device and supports offline. 4) Access to native apps 5)Appstore distribution	1) Cross platform affinity 2)Written with web technologies (HTML, CSS, Javascript or serverside (PHP, ASP.Net .,) 3) Runs on the web server, viewable on multiple devices 5) Centralized updates.
	Platform affinity		

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

According to the previous research [8], 80% of mobile application developments are done by adopting hybrid and mobile web apps and the remaining 20% of mobile application development are done by adopting mobile enterprise application platform and virtualized platform-dependent, which is basically determined by the business needs and targeted devices. Now identifying the right apps suitable for the mobile application development becomes a big challenge, when selecting a hybrid app compared with native and mobile web apps. Many previous researches on mobile application development [9],[16],[12],[14] depicts that the result is about what type of business problem they want to be solved by the mobile application.

In a technical visualization, findings from the previous researches helped the development team to choose the appropriate type of application for the mobile application development. Mobile application development is basically a multi-channel application which is not discussed much specifically on the aspect of software engineering. On the basis of this assumption, the research discusses the software engineering framework in mobile application development.

3. SOFTWARE ENGINEERING AGILE METHODS

Software Engineering Agile development methods have been very much successfully in software projects. Any application development is a part of software engineering interest [13]. The target platform is not a big deal in software engineering. Scrum, XP, or RUP software engineering methods do not discuss about the precise implementation of mobile application. As per the previous research [2], the software engineering processes needs some adjustments. The research also shows many issues and challenges in the development processes, user interface design, tools, application portability, quality and security [2].

Many software organizations such as Sabre, Sprint, Nortel, Symantec, Fidelity, Borland, Qwest and more have already employed agile methods for application development. [11] While developing mobile applications, many companies have faced a lot of complexity and challenges in the development process. Similar to windows-based application and web-based application, the mobile application developers faced many constraints and challenges in the aspects like memory, screen size, input devices, shorter development lifecycles and tremendous usability requirements. Mobile application developers should be really very much careful while addressing the above constraints and challenges during the development and deployment lifecycle. To achieve the above aspects, mobile developers have started using emulators, test automation, automated deployment process and shorter development cycles.

The above mentioned researches depicts mobile software engineering focus is addressing the main problem of the application. Here it is focused that the component based model, state model, and iterative development model will create enough mobile application. The research [5] proposes eight steps to develop mobile application which are as follows:

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

- Decide what your app will do for the end user
- Decide on the platform
- Signup as developer on the platform you have chosen
- Download the API, choose a coder or do it yourself
- Design your app
- Test, test and test again
- Launch and tell
- Support

A complete mobile software engineering approach is discussed in the handbook [6], which depicts the complete references in the design implementation and emergent applications. An important aspect of mobile software engineering is to provide uniform code that envelops mobile application attributes such as reliability, portability, usability, reliability, battery durability, form factor, usage patterns and its architecture.

4. CURRENT RESEARCH SCHEMA ON MOBILE APPLICATION DEVELOPMENT USING AN AGILE METHOD

In the last few years, Agile methods have grained lot of reputation, since many companies faced increasing difficulty to speedup the delivery process with changing requirements and rapidly growing technology as well. The Agile methods have several advantages over the industry momentum because of its simplicity and self-controlled as appropriate to a wide range of today's software projects.

The following objectives provide a good mental picture of this research comparing to the early researches:

- The research [6], gives a full references to mobile software engineering field and at the same time represents the current and future emerging applications.
- The research [2], discusses the significant software engineering research issues in the area of mobile application development.
- The research [5], proposes a sequence of steps to handle technical problems or issues in the areas of mobile application development.
- The research [1], proposes a light weighted framework that uses the extreme programming an agile methodology to handle technical and non-technical problems on small-medium sized mobile application development.
- The current research, proposes a software engineering framework that uses the dynamic system development method an agile methodology to overcome technical and non-technical problems both in small and large medium sized mobile application development.

The previous research had enough addressed about the challenges in developing mobile applications. On the other hand, this research aims on the development of mobile applications by applying an agile method over a software engineering framework by bringing plainness and concentration to the existing mobile technology standards. Here,

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

we mean agile just not referring to the method DSDM-Dynamic System Development Method.

4.1 Comparison between agile methods implemented on the software engineering framework

The main focus of this research is to provide a comparative study about the agile methods such as extreme programming and DSDM-Dynamic Systems Development Methods, implemented on the software engineering framework. This research has basically found out the fundamental difference between the two agile methods is that, the extreme programming is suitable only for the small medium business and DSDM is suitable for large scale business. Agile methods such as Extreme programming (XP) and Rational Unified Process (RUP) etc., represents some resemblance to DSDM-Dynamic Systems Development Method.
In order to provide the specific differences between the agile methods we discuss the disadvantages of extreme prototyping together with the model and advantages of Dynamic Systems Development Method.

4.1.1 Disadvantages of Extreme programming

The disadvantages of extreme programming are discussed as given below:

- Extreme programming is very much complicated at the implementation level.
- Extreme programming life cycle is iterative only but not incremental.
- Extreme programming development team members never trust the concept of fixed price and scope kind of terms with the clients/users.
- Since extreme programming supports pair programming, there is a unnecessary replication of large of quantity of codes that combines with the unit test. This process increases the execution time excess, which results in lot of duplicated data on the database.
- Extreme programming mainly focuses on the coding entity only and not on the design entity. As a result, there is no problem with small scale projects, but remains to be a great flaw in large scale projects.
- Extreme programming does not focus on the quality aspects of application development.
- Extreme programming emphasizes on refactoring during application development process. Actually refactoring leads to a lot of waste in time rather than being productive.

4.1.2 Advantages of Dynamic System Development Method

- Firstly, it has independent framework and unique tools and techniques. DSDM allows users to fill in the definite steps of the process with their unique techniques and software aids of choice.
- Secondly, the variables in the development are not time or resources but the requirements. Hence, this method enables us to maintain the delivery deadline and the project budget.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

- Finally, there is strong focal point on the aspect of stakeholders' communication and involvement in the system development.

5. DSDM APPROACH ON MOBILE APPLICATION DEVELPOMENT

The Dynamic Systems Development Method (DSDM) an Agile method on software engineering framework of controls which enables the development of mobile applications. [17] This agile method is independent of any precise series of techniques and tools. Whether it is a large or small or medium scale application development, this Agile method supports in greater extend and specifically if application development to be done in short deadlines. DSDM applies both iterative and incremental approached for application development.

The proposed software engineering framework using DSDM for mobile application development given below:

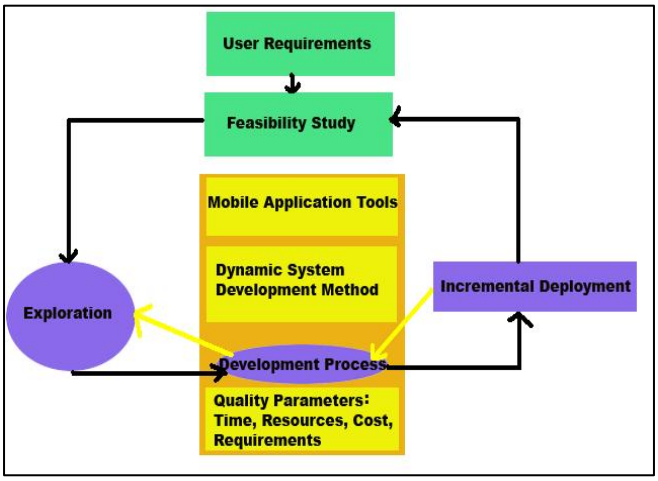


Figure 1: Software Engineering framework using DSDM

The DSDM approach has gained experience over the traditional system development by using the components to make sure that it addresses the right requirements and reduces the development time as well. In this agile method, when the development team involved in a large sized application development, here the development process itself will be broken down into smaller addressable components which is either done for incremental delivery or for application units being developed in parallel by team members. DSDM is able to estimate the cost, quality and time very accurately by applying the principle MOSCOW(musts, should, coulds, and won't) [17] for the application development. DSDM is more suitable for developing mobile application development and non-IT solutions as it forms a part of agile association.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

In this approach, the estimated time and resources are left to be fixed whereas the user requirements can be left changing. Also this approach promise to fulfill at least a minimum compartment of the requirements stated early in the development process.

CONCLUSION

Software Engineering framework using DSDM - Agile mobile application development provides significant opportunities and interest for the mobile application development team members pondering to initiate a light weighted development processes and at the same time addresses the changing user requirements. In the rapidly changing business and technical environment, DSDM -Agile development's mainly focuses on ensuring correct requirements between business and technology is of critical importance. Also not only allowing regular user involvement in the application development by adopting this agile method, the team will able to develop the mobile application in a shorter development cycle and implement more confidently and modify the path in today's changing environment.

REFERENCES

[1] Ridi Ferdiana. 2012. Agile Software Engineering Framework for Evaluation Mobile Application Development. International Journal of Scientific & Engineering Research Volume 3, Issue 12, December-2012.ISSN 2229-5518

[2] Anthony I. Wasserman. 2010. Software engineering issues for mobile application development. In Proceedings of the FSE/SDP workshop on Future of software engineering research (FoSER '10). ACM, New York, NY, USA, 397-400. (Journal Publication)

[3] Dino Esposito. Architecting Mobile Solutions for the Enterprise. Mi- crosoft Press. 2012. (Book)

[4] Doug Seven. 2012. What is a hybrid mobile App? Icenium. Blog pub- lication at http://bit.ly/OMVQVN accessed 23/9/2012. (Unpublished article)

[5] Ivo Salmre. 2005. Writing Mobile Code: Essential Software Engineer- ing for Building Mobile Applications. Addison-Wesley. USA. (Book)

[6] Paulo Alencar and Donald Cowan. 2011. Handbook of Research on Mobile Software Engineering: Design Implementation and Emergent Applications (2 Volumes) (1st Ed.). Engineering Science Reference. (Book)

[7] JT Mudge. 2012. Native App Vs. Mobile Web App: A Quick Comparison at http://sixrevisions.com/mobile/native-app-vs-mobile-web-app-comparison/

[8] Building a Mobile Application Development Framework. Joseph Doolittle Enterprise Architect, Intel IT,Aideen MoohanProject Analyst, Intel IT,John SimpsonEnterprise

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

Architect, Intel IT, Ian SoanesIT Engineering Handheld Product Manager, Intel IT August 2012.

[9] R. Bagrodia, S. Bhattacharyya, F. Cheng, S. Gerding, G. Glazer, R. Guy, Z. Ji, J. Lin, T. Phan, E. Skow, M. Varshney, and G. Zorpas. 2003. iMASH: interactive mobile application session handoff. In Pro- ceedings of the 1st international conference on Mobile systems, ap- plications and services (MobiSys '03). ACM, New York, NY, USA, 259-272. (Journal Publication)

[10] Richard Rodger. Beginning Mobile Application Development in the Cloud. Wrox. 2011. (Book)

[11] Mobile Application Development:A Natural Fit with Agile Methodologies by Robert HollerPresident & CEO, VersionOne.Copyright © 2006 VersionOne, LLC. All rights reserved.

[12] Stefan Saroiu and Alec Wolman. 2009. Enabling new mobile applic a- tions with location proofs. In Proceedings of the 10th workshop on Mobile Computing Systems and Applications (HotMobile '09). ACM, New York, NY, USA, , Article 3 , 6 pages. (Journal Publication)

[13] Steve McConnell. 1996. Rapid Development. Microsoft Press. (Book)

[14] Thomas Olsson, Marika Lehtonen, Dana Pavel, and Kaisa Väänänen - Vainio-Mattila. 2007. User-centered design of a mobile application for sharing life memories. In Proceedings of the 4th international con- ference on mobile technology, applications, and systems and the 1st international symposium on Computer human interaction in mobile technology (Mobility '07). ACM, New York, NY, USA, 524 - 531. (Journal Publication)

[15] Tommi Mikkonen and Antero Taivalsaari. 2008. Towards a Uniform Web Application Platform for Desktop Computers and Mobile D e- vices. Technical Report. Sun Microsystems, Inc., Mountain View, CA, USA. (Journal Publication)

[16] Upkar Varshney and Ron Vetter. 2002. Mobile commerce: frame- work, applications and networking support. Mob. Netw. Appl. 7, 3 (June 2002), 185-198. (Journal Publication)

[17] Pekka Abrahamson, Outi Salo and Jussi Ronkainen, ESPOO2002. Agile Software Development Methods Review and Analysis. VTT Publications 478.

http://sites.google.com/site/ijcsis/
ISSN 1947-5500

IJCSIS AUTHORS' & REVIEWERS' LIST

Assist Prof (Dr.) M. Emre Celebi, Louisiana State University in Shreveport, USA
Dr. Lam Hong Lee, Universiti Tunku Abdul Rahman, Malaysia
Dr. Shimon K. Modi, Director of Research BSPA Labs, Purdue University, USA
Dr. Jianguo Ding, Norwegian University of Science and Technology (NTNU), Norway
Assoc. Prof. N. Jaisankar, VIT University, Vellore,Tamilnadu, India
Dr. Amogh Kavimandan, The Mathworks Inc., USA
Dr. Ramasamy Mariappan, Vinayaka Missions University, India
Dr. Yong Li, School of Electronic and Information Engineering, Beijing Jiaotong University, P.R. China
Assist. Prof. Sugam Sharma, NIET, India / Iowa State University, USA
Dr. Jorge A. Ruiz-Vanoye, Universidad Autónoma del Estado de Morelos, Mexico
Dr. Neeraj Kumar, SMVD University, Katra (J&K), India
Dr Genge Bela, "Petru Maior" University of Targu Mures, Romania
Dr. Junjie Peng, Shanghai University, P. R. China
Dr. Ilhem LENGLIZ, HANA Group - CRISTAL Laboratory, Tunisia
Prof. Dr. Durgesh Kumar Mishra, Acropolis Institute of Technology and Research, Indore, MP, India
Jorge L. Hernández-Ardieta, University Carlos III of Madrid, Spain
Prof. Dr.C.Suresh Gnana Dhas, Anna University, India
Mrs Li Fang, Nanyang Technological University, Singapore
Prof. Pijush Biswas, RCC Institute of Information Technology, India
Dr. Siddhivinayak Kulkarni, University of Ballarat, Ballarat, Victoria, Australia
Dr. A. Arul Lawrence, Royal College of Engineering & Technology, India
Mr. Wongyos Keardsri, Chulalongkorn University, Bangkok, Thailand
Mr. Somesh Kumar Dewangan, CSVTU Bhilai (C.G.)/ Dimat Raipur, India
Mr. Hayder N. Jasem, University Putra Malaysia, Malaysia
Mr. A.V.Senthil Kumar, C. M. S. College of Science and Commerce, India
Mr. R. S. Karthik, C. M. S. College of Science and Commerce, India
Mr. P. Vasant, University Technology Petronas, Malaysia
Mr. Wong Kok Seng, Soongsil University, Seoul, South Korea
Mr. Praveen Ranjan Srivastava, BITS PILANI, India
Mr. Kong Sang Kelvin, Leong, The Hong Kong Polytechnic University, Hong Kong
Mr. Mohd Nazri Ismail, Universiti Kuala Lumpur, Malaysia
Dr. Rami J. Matarneh, Al-isra Private University, Amman, Jordan
Dr Ojesanmi Olusegun Ayodeji, Ajayi Crowther University, Oyo, Nigeria
Dr. Riktesh Srivastava, Skyline University, UAE
Dr. Oras F. Baker, UCSI University - Kuala Lumpur, Malaysia
Dr. Ahmed S. Ghiduk, Faculty of Science, Beni-Suef University, Egypt
and Department of Computer science, Taif University, Saudi Arabia
Mr. Tirthankar Gayen, IIT Kharagpur, India
Ms. Huei-Ru Tseng, National Chiao Tung University, Taiwan

CALL FOR PAPERS
International Journal of Computer Science and Information Security

IJCSIS 2013
ISSN: 1947-5500
http://sites.google.com/site/ijcsis/

International Journal Computer Science and Information Security, IJCSIS, is the premier scholarly venue in the areas of computer science and security issues. IJCSIS 2011 will provide a high profile, leading edge platform for researchers and engineers alike to publish state-of-the-art research in the respective fields of information technology and communication security. The journal will feature a diverse mixture of publication articles including core and applied computer science related topics.

Authors are solicited to contribute to the special issue by submitting articles that illustrate research results, projects, surveying works and industrial experiences that describe significant advances in the following areas, but are not limited to. Submissions may span a broad range of topics, e.g.:

Track A: Security

Access control, Anonymity, Audit and audit reduction & Authentication and authorization, Applied cryptography, Cryptanalysis, Digital Signatures, Biometric security, Boundary control devices, Certification and accreditation, Cross-layer design for security, Security & Network Management, Data and system integrity, Database security, Defensive information warfare, Denial of service protection, Intrusion Detection, Anti-malware, Distributed systems security, Electronic commerce, E-mail security, Spam, Phishing, E-mail fraud, Virus, worms, Trojan Protection, Grid security, Information hiding and watermarking & Information survivability, Insider threat protection, Integrity
Intellectual property protection, Internet/Intranet Security, Key management and key recovery, Language-based security, Mobile and wireless security, Mobile, Ad Hoc and Sensor Network Security, Monitoring and surveillance, Multimedia security ,Operating system security, Peer-to-peer security, Performance Evaluations of Protocols & Security Application, Privacy and data protection, Product evaluation criteria and compliance, Risk evaluation and security certification, Risk/vulnerability assessment, Security & Network Management, Security Models & protocols, Security threats & countermeasures (DDoS, MiM, Session Hijacking, Replay attack etc,), Trusted computing, Ubiquitous Computing Security, Virtualization security, VoIP security, Web 2.0 security, Submission Procedures, Active Defense Systems, Adaptive Defense Systems, Benchmark, Analysis and Evaluation of Security Systems, Distributed Access Control and Trust Management, Distributed Attack Systems and Mechanisms, Distributed Intrusion Detection/Prevention Systems, Denial-of-Service Attacks and Countermeasures, High Performance Security Systems, Identity Management and Authentication, Implementation, Deployment and Management of Security Systems, Intelligent Defense Systems, Internet and Network Forensics, Large-scale Attacks and Defense, RFID Security and Privacy, Security Architectures in Distributed Network Systems, Security for Critical Infrastructures, Security for P2P systems and Grid Systems, Security in E-Commerce, Security and Privacy in Wireless Networks, Secure Mobile Agents and Mobile Code, Security Protocols, Security Simulation and Tools, Security Theory and Tools, Standards and Assurance Methods, Trusted Computing, Viruses, Worms, and Other Malicious Code, World Wide Web Security, Novel and emerging secure architecture, Study of attack strategies, attack modeling, Case studies and analysis of actual attacks, Continuity of Operations during an attack, Key management, Trust management, Intrusion detection techniques, Intrusion response, alarm management, and correlation analysis, Study of tradeoffs between security and system performance, Intrusion tolerance systems, Secure protocols, Security in wireless networks (e.g. mesh networks, sensor networks, etc.), Cryptography and Secure Communications, Computer Forensics, Recovery and Healing, Security Visualization, Formal Methods in Security, Principles for Designing a Secure Computing System, Autonomic Security, Internet Security, Security in Health Care Systems, Security Solutions Using Reconfigurable Computing, Adaptive and Intelligent Defense Systems, Authentication and Access control, Denial of service attacks and countermeasures, Identity, Route and

Location Anonymity schemes, Intrusion detection and prevention techniques, Cryptography, encryption algorithms and Key management schemes, Secure routing schemes, Secure neighbor discovery and localization, Trust establishment and maintenance, Confidentiality and data integrity, Security architectures, deployments and solutions, Emerging threats to cloud-based services, Security model for new services, Cloud-aware web service security, Information hiding in Cloud Computing, Securing distributed data storage in cloud, Security, privacy and trust in mobile computing systems and applications, **Middleware security & Security features:** middleware software is an asset on
its own and has to be protected, interaction between security-specific and other middleware features, e.g., context-awareness, **Middleware-level security monitoring and measurement:** metrics and mechanisms for quantification and evaluation of security enforced by the middleware, **Security co-design:** trade-off and co-design between application-based and middleware-based security, **Policy-based management:** innovative support for policy-based definition and enforcement of security concerns, **Identification and authentication mechanisms:** Means to capture application specific constraints in defining and enforcing access control rules, **Middleware-oriented security patterns:** identification of patterns for sound, reusable security, **Security in aspect-based middleware:** mechanisms for isolating and enforcing security aspects, **Security in agent-based platforms:** protection for mobile code and platforms, Smart Devices: Biometrics, National ID cards, Embedded Systems Security and TPMs, RFID Systems Security, Smart Card Security, Pervasive Systems: Digital Rights Management (DRM) in pervasive environments, Intrusion Detection and Information Filtering, Localization Systems Security (Tracking of People and Goods), Mobile Commerce Security, Privacy Enhancing Technologies, Security Protocols (for Identification and Authentication, Confidentiality and Privacy, and Integrity), Ubiquitous Networks: Ad Hoc Networks Security, Delay-Tolerant Network Security, Domestic Network Security, Peer-to-Peer Networks Security, Security Issues in Mobile and Ubiquitous Networks, Security of GSM/GPRS/UMTS Systems, Sensor Networks Security, Vehicular Network Security, Wireless Communication Security: Bluetooth, NFC, WiFi, WiMAX, WiMedia, others

This Track will emphasize the design, implementation, management and applications of computer communications, networks and services. Topics of mostly theoretical nature are also welcome, provided there is clear practical potential in applying the results of such work.

Track B: Computer Science

Broadband wireless technologies: LTE, WiMAX, WiRAN, HSDPA, HSUPA, Resource allocation and interference management, Quality of service and scheduling methods, Capacity planning and dimensioning, Cross-layer design and Physical layer based issue, Interworking architecture and interoperability, Relay assisted and cooperative communications, Location and provisioning and mobility management, Call admission and flow/congestion control, Performance optimization, Channel capacity modeling and analysis, Middleware Issues: Event-based, publish/subscribe, and message-oriented middleware, Reconfigurable, adaptable, and reflective middleware approaches, Middleware solutions for reliability, fault tolerance, and quality-of-service, Scalability of middleware, Context-aware middleware, Autonomic and self-managing middleware, Evaluation techniques for middleware solutions, Formal methods and tools for designing, verifying, and evaluating, middleware, Software engineering techniques for middleware, Service oriented middleware, Agent-based middleware, Security middleware, Network Applications: Network-based automation, Cloud applications, Ubiquitous and pervasive applications, Collaborative applications, RFID and sensor network applications, Mobile applications, Smart home applications, Infrastructure monitoring and control applications, Remote health monitoring, GPS and location-based applications, Networked vehicles applications, Alert applications, Embeded Computer System, Advanced Control Systems, and Intelligent Control : Advanced control and measurement, computer and microprocessor-based control, signal processing, estimation and identification techniques, application specific IC's, nonlinear and adaptive control, optimal and robot control, intelligent control, evolutionary computing, and intelligent systems, instrumentation subject to critical conditions, automotive, marine and aero-space control and all other control applications, Intelligent Control System, Wiring/Wireless Sensor, Signal Control System. Sensors, Actuators and Systems Integration : Intelligent sensors and actuators, multisensor fusion, sensor array and multi-channel processing, micro/nano technology, microsensors and microactuators, instrumentation electronics, MEMS and system integration, wireless sensor, Network Sensor, Hybrid

Sensor, Distributed Sensor Networks. Signal and Image Processing : Digital signal processing theory, methods, DSP implementation, speech processing, image and multidimensional signal processing, Image analysis and processing, Image and Multimedia applications, Real-time multimedia signal processing, Computer vision, Emerging signal processing areas, Remote Sensing, Signal processing in education. Industrial Informatics: Industrial applications of neural networks, fuzzy algorithms, Neuro-Fuzzy application, bioInformatics, real-time computer control, real-time information systems, human-machine interfaces, CAD/CAM/CAT/CIM, virtual reality, industrial communications, flexible manufacturing systems, industrial automated process, Data Storage Management, Harddisk control, Supply Chain Management, Logistics applications, Power plant automation, Drives automation. Information Technology, Management of Information System : Management information systems, Information Management, Nursing information management, Information System, Information Technology and their application, Data retrieval, Data Base Management, Decision analysis methods, Information processing, Operations research, E-Business, E-Commerce, E-Government, Computer Business, Security and risk management, Medical imaging, Biotechnology, Bio-Medicine, Computer-based information systems in health care, Changing Access to Patient Information, Healthcare Management Information Technology. Communication/Computer Network, Transportation Application : On-board diagnostics, Active safety systems, Communication systems, Wireless technology, Communication application, Navigation and Guidance, Vision-based applications, Speech interface, Sensor fusion, Networking theory and technologies, Transportation information, Autonomous vehicle, Vehicle application of affective computing, Advance Computing technology and their application : Broadband and intelligent networks, Data Mining, Data fusion, Computational intelligence, Information and data security, Information indexing and retrieval, Information processing, Information systems and applications, Internet applications and performances, Knowledge based systems, Knowledge management, Software Engineering, Decision making, Mobile networks and services, Network management and services, Neural Network, Fuzzy logics, Neuro-Fuzzy, Expert approaches, Innovation Technology and Management : Innovation and product development, Emerging advances in business and its applications, Creativity in Internet management and retailing, B2B and B2C management, Electronic transceiver device for Retail Marketing Industries, Facilities planning and management, Innovative pervasive computing applications, Programming paradigms for pervasive systems, Software evolution and maintenance in pervasive systems, Middleware services and agent technologies, Adaptive, autonomic and context-aware computing, Mobile/Wireless computing systems and services in pervasive computing, Energy-efficient and green pervasive computing, Communication architectures for pervasive computing, Ad hoc networks for pervasive communications, Pervasive opportunistic communications and applications, Enabling technologies for pervasive systems (e.g., wireless BAN, PAN), Positioning and tracking technologies, Sensors and RFID in pervasive systems, Multimodal sensing and context for pervasive applications, Pervasive sensing, perception and semantic interpretation, Smart devices and intelligent environments, Trust, security and privacy issues in pervasive systems, User interfaces and interaction models, Virtual immersive communications, Wearable computers, Standards and interfaces for pervasive computing environments, Social and economic models for pervasive systems, Active and Programmable Networks, Ad Hoc & Sensor Network, Congestion and/or Flow Control, Content Distribution, Grid Networking, High-speed Network Architectures, Internet Services and Applications, Optical Networks, Mobile and Wireless Networks, Network Modeling and Simulation, Multicast, Multimedia Communications, Network Control and Management, Network Protocols, Network Performance, Network Measurement, Peer to Peer and Overlay Networks, Quality of Service and Quality of Experience, Ubiquitous Networks, Crosscutting Themes – Internet Technologies, Infrastructure, Services and Applications; Open Source Tools, Open Models and Architectures; Security, Privacy and Trust; Navigation Systems, Location Based Services; Social Networks and Online Communities; ICT Convergence, Digital Economy and Digital Divide, Neural Networks, Pattern Recognition, Computer Vision, Advanced Computing Architectures and New Programming Models, Visualization and Virtual Reality as Applied to Computational Science, Computer Architecture and Embedded Systems, Technology in Education, Theoretical Computer Science, Computing Ethics, Computing Practices & Applications

Authors are invited to submit papers through e-mail ijcsiseditor@gmail.com. Submissions must be original and should not have been published previously or be under consideration for publication while being evaluated by IJCSIS. Before submission authors should carefully read over the journal's Author Guidelines, which are located at http://sites.google.com/site/ijcsis/authors-notes .

ISSN 1947 5500
http://sites.google.com/site/ijcsis/