Srinivas Aluru   Sanghamitra Bandyopadhyay
Umit V. Catalyurek   Devdatt P. Dubhashi
Phillip H. Jones   Manish Parashar
Bertil Schmidt (Eds.)

Communications in Computer and Information Science   168

Contemporary Computing

4th International Conference, IC3 2011
Noida, India, August 2011
Proceedings

Springer

Communications
in Computer and Information Science 168

Srinivas Aluru Sanghamitra Bandyopadhyay
Umit V. Catalyurek Devdatt P. Dubhashi
Phillip H. Jones Manish Parashar
Bertil Schmidt (Eds.)

Contemporary Computing

4th International Conference, IC3 2011
Noida, India, August 8-10, 2011
Proceedings

Springer

Volume Editors

Srinivas Aluru
Iowa State University, Ames, IA, USA and IIT Bombay, India
E-mail: aluru@iastate.edu

Sanghamitra Bandyopadhyay
Indian Statistical Institute, Kolkata, India
E-mail: sanghami@isical.ac.in

Umit V. Catalyurek
The Ohio State University, Columbus, OH, USA
E-mail: catalyurek.1@osu.edu

Devdatt P. Dubhashi
Chalmers University, Göteborg, Sweden
E-mail: dubhashi@cs.chalmers.se

Phillip H. Jones
Iowa State University, Ames, IA, USA
E-mail: phjones@iastate.edu

Manish Parashar
Rutgers, The State University of New Jersey, Piscataway, NJ, USA
E-mail: parashar@rutgers.edu

Bertil Schmidt
Nanyang Technological University, Singapore
E-mail: asbschmidt@ntu.edu.sg

ISSN 1865-0929 e-ISSN 1865-0937
ISBN 978-3-642-22605-2 e-ISBN 978-3-642-22606-9
DOI 10.1007/978-3-642-22606-9
Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2011931779

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

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

Printed on acid-free paper

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

Preface

Welcome to the proceedings of the 2011 International Conference on Contemporary Computing. This was the fourth conference in the series, held annually at the Jaypee Institute of Information Technology, and organized jointly by the hosts and the University of Florida, Gainesville, USA. The conference focuses on issues of contemporary interest in computing, spanning systems, algorithms and applications.

This year's program consisted of 58 contributed papers chosen from among 175 submissions. The submissions were subjected to a rigorous peer-review process by an international team of 130 experts serving as Technical Program Committee members. The Program Committee was led by six Program Track Chairs, themselves representing an eclectic mix of distinguished international experts – Devdatt Dubhashi and Ümit V. Çatalyürek for Algorithms, Bertil Schmidt and Sanghamitra Bandhyopadhyay for Applications, and Phillip Jones and Manish Parashar for Systems. In addition to the contributed papers, the technical program was flanked by seven distinguished keynote speakers:

- Paul Mc Kevitt, University of Ulster, UK
- Jitendra Malik, University of California at Berkeley, USA
- Arun K. Pujari, Sambalpur University, India
- Sanguthevar Rajasekaran, University of Connecticut, USA
- M.P. Ranjan, National Institute of Design, India
- Jose Rolim, University of Geneva, Switzerland
- Guna Seetharaman, Airforce Research Laboratory, Rome, NY, USA

I am grateful to the authors of all the submitted papers for choosing this conference to present their work. The conference would not be possible without your efforts and valuable contributions. I am grateful to the Program Committee for providing thoughtful and timely reviews. Their collective expertise allowed the Track Chairs and myself to conduct a fair evaluation of contributions spanning diverse areas of contemporary computer science research. I would like to thank the General Chairs Sartaj Sahni and Sanjay Goel for giving me the opportunity to lead this year's technical program. They kept tabs on the timelines, and provided guidance and invaluable assistance throughout the process.

Srinivas Aluru

Organization

Chief Patron

Shri Jaiprakash Gaur

Patron

Shri Manoj Gaur

Advisory Committee

S.K. Khanna	Jaypee Institute of Information Technology, India
M.N. Farruqui	Jaypee Institute of Information Technology, India
Y. Medury	Jaypee Institute of Information Technology, India
J.P. Gupta	Jaypee Institute of Information Technology, India
T.R.Kakkar	Jaypee Institute of Information Technology, India
S.L. Maskara	Jaypee Institute of Information Technology, India

Steering Committee

Yaj Medury	Jaypee Institute of Information Technology, India
Sartaj Sahni	University of Florida, USA
Sanjay Ranka	University of Florida,USA
Sanjay Goel	Jaypee Institute of Information Technology, India
Srinivas Aluru	Iowa State University, USA

General Co-chairs

Sartaj Sahni	University of Florida, USA
Sanjay Goel	Jaypee Institute of Information Technology, India

Program Chair

Srinivas Aluru	Iowa State University, USA and IIT Bombay, India

Track Co-chairs

Algorithms

Devdatt Dubhashi	Chalmers University of Technology and University of Gothenburg, Sweden
Umit V. Catalyurek	Ohio State University, USA

Applications

Sanghamitra Bandyopadhyay	Indian Statistical Institute, India
Bertil Schmidt	Nanyang Technological University, Singapore

Systems (Hardware and Software)

Phillip H. Jones	Iowa State University, USA
Manish Parashar	National Science Foundation and Rutgers University, USA

Technical Program Committee

Algorithms

Cevdet Aykanat	Bilkent University, Ankara, Turkey
Amitabha Bagchi	Indian Institute of Technology, Delhi, India
Olivier Beaumont	INRIA, France
Costas Bekas	IBM Zurich Research Laboratory, Switzerland
Sanjukta Bhowmick	University of Nebraska at Omaha, USA
Erik Boman	Sandia National Laboratories, USA
Dany Breslauer	University of Haifa, Israel
Aydin Buluc	Lawrence Berkeley National Laboratory, USA
Peter Damaschke	Chalmers University of Technology, Sweden
Eylem Ekici	The Ohio State University, USA
Renato Ferreira	Universidade Federal de Minas Gerais, Brazil
Irene Finocchi	University of Rome “La Sapienza”, Italy
Assefaw Gebremedhin	Purdue University, USA
Mahantesh Halappanavar	Pacific Northwest National Laboratory, USA
Giuseppe Italiano	University of Rome “Tor Vergata”, Italy
Rohit Khandekar	IBM Research, USA
Michael Langston	University of Tennessee, USA
Alexey Lastovetsky	University College Dublin, Ireland
Fredrik Manne	University of Bergen, Norway

Madhav Marathe	Virginia Tech, USA
Marco Pellegrini	IIT-CNR, Italy
Andrea Pietracaprina	University of Padova, Italy
Geppino Pucci	University of Padova, Italy
Erik Saule	The Ohio State University, USA
Anand Srivastava	Kiel University, Germany
KV Subrahmanyam	Chennai Mathematical Institute, India
Kavitha Telikepalli	Tata Institute of Fundamental Research, India
Denis Trystram	University of Grenoble, France
Bora Ucar	CNRS, France
Anil Kumar Vullikanti	Virginia Tech, USA
Osamu Watanabe	Tokyo Institute of Technology, Japan
Jan-Jan Wu	Academia Sinica, Taiwan

Applications

Ishfaq Ahmad	University of Texas at Arlington, USA
Travis Atkison	Louisiana Tech University, USA
Ansuman Banerjee	Indian Statistical Institute, India
Roberto Baragona	University of Rome, Italy
V. Bhatnagar	Delhi University, India
K.K. Biswas	IIT Delhi, India
Leonard Brown	The University of Texas at Tyler, USA
Sung-Bae Cho	Yonsei University, Korea
Pradeep Chowriappa	Louisiana Tech University, USA
Chris Clarke	University of Bath, USA
Carlos Coello Coello	Cinvestav, Mexico
Chris Cornelis	Ghent University, USA
Rajat De	Indian Statistical Institute, India
Oliver Diessel	University of New South Wales, Australia
Christian Duncan	Louisiana Tech University, Australia
Sumantra Dutta Roy	Indian Institute of Technology Delhi, India
Magdalini Eirinaki	San Jose State University, USA
Scott Emrich	University of Notre Dame, USA
Rui Fan	Nanyang Technological University, USA
Utpal Garain	Indian Statistical Institute, Kolkata, India
Kuntal Ghosh	Indian Statistical Institute, India
Rick Goh	A*STAR Institute of High-Performance Computing, Singapore
Nigel Gwee	Southern University, USA
Sridhar Hariharaputran	Bielefeld University, Germany
Bingsheng He	Nanyang Technological University, Singapore
Alexandru Iosup	Delft University of Technology, The Netherlands
Arpith Jacob	Washington University in St. Louis, USA

Lars Kaderali	University of Heidelberg, Germany
Ananth Kalyanaraman	Washington State University, USA
Naveen Kumar	Delhi University, India
Dominique Lavenier	CNRS-IRISA, France
G.S. Lehal	Punjabi University, India
Mo Li	Nanyang Technological University, Singapore
Weiguo Liu	Nanyang Technological University, Singapore
Subhhamoy Maitra	Indian Statistical Institute, India
Pradipta Maji	Indian Statistical Institute, India
Francesco Masulli	University of Genoa, Italy
Martin Middendorf	University of Leipzig, Germany
Sonajharia Minz	Jawaharlal Nehru University, India
Pabitra Mitra	Indian Institute of Technology, Kharagpur, India
Suman Mitra	DAIICT, India
Teng Moh	San Jose State University, USA
Anirban Mondal	IIIT Delhi, India
Ponnuthurai Nagaratnam Suganthan	Nanyang Technological University, Singapore
Vinayak Naik	IIIT Delhi, India
Mita Nasipuri	Jadavpur University, India
Nicolas Pasquier	University of Nice, France
Partha Roop	University of Auckland, New Zealand
Witold Rudnicki	University of Warsaw, Poland
Punam Saha	University of Iowa, USA
Sheetal Saini	Louisiana Tech University, USA
Stan Scott	Queen's University Belfast, UK
Pushpendra Singh	IIIT Delhi, India
Alexandros Stamatakis	Heidelberg Institute for Theoretical Studies, Germany
Peter Strazdins	Australian National University, Australia
Ashish Sureka	IIIT Delhi, India
Jean-Stéphane Varré	Université Lille 1, LIFL, CNRS, INRIA Lille, France
Gerrit Voss	Nanyang Technological University, Singapore
Malcolm Yoke Hean Low	Nanyang Technological University, Singapore
Jaroslaw Zola	Iowa State University, USA

Systems (Hardware and Software)

Gagan Agrawal	The Ohio State University, USA
Zachary Baker	Los Alamos National Laboratory, USA
Pavan Balaji	Argonne National Laboratory, USA
Viraj Bhat	Yahoo!, UK
Roger Chamberlain	Washington University in St. Louis, USA

Sanjay Chaudhary	Dhirubhai Ambani Institute of Information and Communication Technology, India
Young Cho	University of Southern California, USA
Yeh-Ching Chung	National Tsing Hua University, China
Ewa Deelman	ISI, USC, UK
Mark Gardner	Virginia Tech, USA
Nathan Gnanasambandam	Xerox Research, USA
Madhusudhan Govindaraju	SUNY-Binghamton, USA
Jie Hu	New Jersey Institute of Technology, USA
Adriana Iamnitchi	University of South Florida, USA
Zbigniew Kalbarczyk	University of Illinois at Urbana Champaign, USA
Scott Klasky	Oak Ridge National Laboratory, USA
Tevfik Kosar	State University of New York at Buffalo, USA
Xiaolin (Andy) Li	University of Florida, USA
Shih-Lien Lu	Intel Corporation, USA
Bharat Madan	Applied Research Laboratory - Penn State University, USA
Pramod Meher	Nanyang Technological University, Singapore
Philippe O.A. Navaux	Universidade Federal do Rio Grande do Sul, Brazil
Sushil Prasad	Georgia State University, USA
Viktor Prasanna	University of Southern California, USA
Rajeev Raje	IUPUI, USA
Ivan Rodero	Rutgers University, USA
Ian Rogers	Azul Systems, USA
Arrvindh Shriraman	Simon Fraser University, Canada
Ashok Srinivasan	Florida State University, USA
Parimala Thulasiraman	University of Manitoba, Canada
Ramachandran Vaidyanathan	Louisiana State University, USA
Joseph Zambreno	Iowa State University, USA
Wei Zhang	Virginia Commonwealth University, USA
Albert Zomaya	The University of Sydney, Australia

Publicity Co-chairs

Divakar Yadav	JIIT, Noida, India
Rajkumar Buyya	University of Melbourne, AUS
Paolo Bellavista	University of Bologna, Italy
Koji Nakano	Hiroshima University, Japan
Masoud Sadjadi	Florida International University, USA
Bhardwaj Veeravalli	University of Singapore, Singapore

Publications Committee

Vikas Saxena	JIIT, Noida, India (Publication Chair)
Alok Aggarwal	JIIT, Noida, India
Abhishek Swaroop	JIIT, Noida, India
Mukta Goel	JIIT, Noida, India
Pawan Kumar Upadhyay	JIIT, Noida, India
Rakhi Hemani	JIIT, Noida, India
Chetna Dabas	JIIT, Noida, India

Web Administration

Sandeep K. Singh	JIIT, Noida, India
Shikha Mehta	JIIT, Noida, India

Graphic Design

Sangeeta Malik	JIIT, Noida, India

Registration and Local Arrangements Co-chairs

Krishna Asawa	JIIT, Noida, India
Prakash Kumar	JIIT, Noida, India
M. Hima Bindu	JIIT, Noida, India
Manish K. Thakur	JIIT, Noida, India

Local Arrangements Committee

Manoj Bharadwaj	JIIT, Noida, India
O.N. Singh	JIIT, Noida, India
S.J.S Soni	JIIT, Noida, India
Akhilesh Sachan	JIIT Noida, India
Sanjay Kataria	JIIT Noida, India
S. Bhaseen	JIIT Noida, India
Adarsh Kumar	JIIT, Noida, India
Anshul Gakhar	JIIT, Noida, India
Anuj Gupta	JIIT, Noida, India
Anuja Arora	JIIT, Noida, India
Archana Purwar	JIIT, Noida, India
Arti Gupta	JIIT, Noida, India
Chetna Gupta	JIIT, Noida, India
Gagandeep Kaur	JIIT, Noida, India
Hema N.	JIIT, Noida, India
Indu Chawla	JIIT, Noida, India

Jolly Shah	JIIT, Noida, India
K. Rajalakshmi	JIIT, Noida, India
Kavita Pandey	JIIT, Noida, India
Megha Rathi	JIIT, Noida, India
Maneesha Srivastava	JIIT, Noida, India
Manisha Rathi	JIIT, Noida, India
Minakshi Gujral	JIIT, Noida, India
Parmeet Kaur	JIIT, Noida, India
Pawan Kumar Upadhyay	JIIT, Noida, India
Prashant Kaushik	JIIT, Noida, India
Pritee Parwekar	JIIT, Noida, India
Purtee Kohli	JIIT, Noida, India
Sangeeta Mittal	JIIT, Noida, India
Shikha Jain	JIIT, Noida, India
Saurabh Kumar Raina	JIIT, Noida, India
Suma Dawn	JIIT, Noida, India
Tribhuvan Kumar Tewari	JIIT, Noida, India
Vimal Kumar K.	JIIT, Noida, India
Vivek Mishra	JIIT, Noida, India

Table of Contents

Regular Paper

Application

System (Hardware and Software)

Poster Paper

Erratum

Energy Balance Mechanisms and Lifetime Optimization of Wireless Networks

Jose D.P. Rolim

Director, Center Universitaire d'Informatique of the University of Geneva
Jose.Rolim@unige.ch

Abstract. In this talk, we consider the problem of data propagation in wireless sensor networks and revisit the family of mixed strategy routing schemes. We will argue that maximizing the lifespan, balancing the energy among individual sensors and maximizing the message flow in the network are equivalent. We note that energy balance, although implying global optimality, is a local property that can be computed efficiently and in a distributed manner. We will then review some distributed, adaptative and on-line algorithms for balancing the energy among sensors.

By considering a simple model of the network and using a linear programming description of the message flow, we will show the strong result that an energy-balanced mixed strategy beats every other possible routing strategy in terms of lifespan maximization. We finalize by remarking that although the results discussed in this talk have a direct consequence in energy saving for wireless networks they do not limit themselves to this type of networks neither to energy as a resource. As a matter of fact, the results are much more general and can be used for any type of network and different type of resources.

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, p. 1, 2011.

Intelligent Autonomous Systems: A Layered Architecture in the Age of Multicore Processing

Guna Seetharaman

Information Directorate, Air Force Research Laboratory, Rome, NY, USA
guna@cacs.louisiana.edu

Abstract. Research in intelligent autonomous systems has historically been constrained by access to scalable high-performance computing. The challenges associated with scalability and tractability have significantly influenced analytical approaches and may have led to analysis in the lower dimensional spaces. Such reductions may overly simplify approaches to model and exploit redundancy in manners that human perception is able to in a bidirectional inferencing process. Evidence indicates that human perception involves different methods of classifying, indexing and associating information according to different spatio-temporal and saliency metrics. Recent surge in high performance computing with multicore processors and wide spread access to large disk-space at finer granularity, have triggered a renewed assessment of existing approaches. A survey of the techniques lead to a compelling need for efficient methods for capturing and exploiting “context” and “context-specific information” to swiftly change the behaviors of intelligent systems. The talk will highlight the evolution of signal processing, linguistic hierarchy, computing models where information fusion is considered, and draw some parallels, to make a case for a layered architecture for context-adaptive autonomous systems. Interactive access to petascale supercomputing has given a newer framework to incorporate context in such studies at scale that is required to perform complex intelligence tasks. The talk will highlight our current research in this direction including our vision of the future of intelligent autonomous systems with embedded high-performance computing with reach back compute power.

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, p. 2, 2011.

Intelligent Multimedia at the Imagineering Quarter

Paul Mc Kevitt

Chair in Intelligent MultiMedia, University of Ulster, Northern Ireland
p.mckevitt@ulster.ac.uk

Abstract. Here we focus on research in Intelligent MultiMedia or MultiModal computing concerning the computer processing and understanding of perceptual signal and symbol input from at least speech, text and visual images, and then reacting to it, involving signal and symbol processing techniques from engineering, computer science, artificial intelligence and cognitive science. With IntelliMedia systems, people can interact in spoken dialogues with machines, querying about what is being presented and even their gestures and body language can be interpreted. Of particular interest is the mapping of inputs into, and outputs out of, semantic representations and this is what distinguishes Intelligent MultiMedia from traditional MultiMedia. We will demonstrate here software prototypes such as `PlayPhysics', which uses computer games to teach physics to first year university students. PlayPhysics is a virtual learning environment for teaching physics which integrates research in Intelligent Tutoring Systems (ITSs), where students learn about concepts such as momentum by trying to get an astronaut back to his craft in time by determining optimal mass and velocity. PlayPhysics also gives detailed feedback online. Another prototype we have developed is `MemoryLane', a mobile digital storytelling companion for older people. Reminders of the person's past, such as photos, video, favourite songs or poems (provided by the individual or their family) are input as text, image, moving image and sound, creating the material from which multimodal stories can be generated. Each story is different and MemoryLane can factor in any problems with the person's eyesight, hearing or dexterity and adapt the presentation accordingly (like making the text larger or reducing the amount of sound or images). Based also on preferences it allows the holder to select aspects they like and reject anything they wish to forget. All of this work falls within `The Imagineering Quarter' within the City of Derry/Londonderry, Northern Ireland, comprising 5 neighbouring buildings of the North West Regional College (NWRC) (`Foyle', `Strand', `Lawrence') & University of Ulster (`Foyle Arts', `Computing') focussed on teaching, research & technology transfer with software demonstrators in Digital Creativity (digital storytelling, music, film, theatre, dance, art, design; games, virtual worlds) linking to The Nerve Centre, Verbal Arts Centre community centres, cross-border Letterkenny Institute of Technology (LYIT), local software industry & access to Project Kelvin -- a secure high capacity dedicated broadband link (10 G. LanPhy) direct to Canada, USA, Europe & rest of the island with a delay of only 2 ms. Derry/Londoderry is First UK City of Culture, 2013 and a key contributor is The Imagineering Quarter.

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, p. 3, 2011.

Computational Techniques for Motif Search

Sanguthevar Rajasekaran

UTC Chair Professor, Dept. of CSE & Director, Booth Engineering Center for Advanced Technologies, University of Connecticut
rajasek@engr.uconn.edu

Abstract. The problem of identifying meaningful patterns (i.e., motifs) from biological data has been studied extensively due to its paramount importance. Motifs are fundamental functional elements in proteins vital for understanding gene function, human disease, and identifying potential therapeutic drug targets. Several versions of the motif search problem have been identified in the literature. Numerous algorithms have been proposed for motif search as well. In this talk we survey some of these algorithms. We also summarize our contributions to motif search and related problems. In addition, we will summarize MnM, a web based system built by us for motif search that is used by biologists widely.

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, p. 4, 2011.

Bi-Objective Community Detection (BOCD) in Networks Using Genetic Algorithm

Rohan Agrawal

Jaypee Institute of Information Technology, Computer Science Department,
Noida - 201307, Uttar Pradesh, India
rohan.agrawal.89@jiitu.org

Abstract. A lot of research effort has been put into community detection from all corners of academic interest such as physics, mathematics and computer science. In this paper I have proposed a Bi-Objective Genetic Algorithm for community detection which maximizes modularity and community score. Then the results obtained for both benchmark and real life data sets are compared with other algorithms using the modularity and MNI performance metrics. The results show that the BOCD algorithm is capable of successfully detecting community structure in both real life and synthetic datasets, as well as improving upon the performance of previous techniques.

Keywords: Community Structure, Community detection, Genetic Algorithm, Multi-objective Genetic Algorithm, Multi-objective optimization, modularity, Normalized Mutual Information, Bi-objective Genetic Algorithm.

1 Introduction

In the context of networks, community structure refers to the occurrence of groups of nodes in a network that are more densely connected than with the rest of the nodes in the network. The inhomogeneous connections suggest that the network has certain natural division within it.

The occurrence of community structure is quite common in real networks. An example of the occurrence of community structure in real networks is the appearance of groups in social networks. Let's take the example of a social networking site. Let a node represent an individual and let the edge represent friendship relation between two individuals. If many students in a particular class or school are friends among themselves, then the network graph will have many connections between them. Thus one community could be identified as a school community. Other communities could be related to work, family, colleges or common interests.

Other examples are citation networks which form communities by research topics. Sport teams form communities on the basis of the division in which they play, as they will play more often with teams that are in the same division/community as them.

Now let us consider the potential applications of the detection of communities in networks. Communities in a social network might help us find real social groupings, perhaps by interest or background. Communities can have concrete applications. Clustering Web clients who have similar interests and are geographically near to each

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 5–15, 2011.

other may improve the performance of services provided on the World Wide Web, in that each cluster of clients could be served by a dedicated mirror server [1]. Identifying clusters of customers with similar interests in the network of purchase relationships between customers and products of online retailers enables to set up efficient recommendation systems [2], that better guide customers through the list of items of the retailer and enhance the business opportunities. Clusters of large graphs can be used to create data structures in order to efficiently store the graph data and to handle navigational queries, like path searches [3][4]. Ad hoc networks [5], i.e. self-configuring networks formed by communication nodes acting in the same region and rapidly changing (because the devices move, for instance), usually have no centrally maintained routing tables that specify how nodes have to communicate to other nodes. Grouping the nodes into clusters enables one to generate compact routing tables while the choice of the communication paths is still efficient [6].

The aim of community detection in graphs is to identify the modules by using the information encoded in the network topology. Weiss and Jacobson [7] were among the first to analyze community structure. They searched for work groups within a government agency. Already in 1927, Stuart Rice looked for clusters of people in small political bodies based on the similarity of their voting patterns [8].

In a paper appearing in 2002, Girvan and Newman proposed a new algorithm, aiming at the identification of edges lying between communities and their successive removal. After a few iterations, this process led to the isolation of communities [9]. The paper triggered inertest in this field, and many new methods have been proposed in previous years.

In particular, physicists entered the game, bringing in their tools and techniques: spin models, optimization, percolation, random walks, synchronization, etc., became ingredients of new original algorithms. The field has also taken advantage of concepts and methods from computer science, nonlinear dynamics, sociology, discrete mathematics.

Genetic algorithms [10] have also been used to optimize modularity. In a standard genetic algorithm one has a set of candidate solutions to a problem, which are numerically encoded as chromosomes, and an objective function to be optimized on the space of solutions. The objective function plays the role of biological fitness for the chromosomes. One usually starts from a random set of candidate solutions, which are progressively changed through manipulations inspired by biological processes regarding real chromosomes, like point mutation (random variations of some parts of the chromosome) and crossing over (generating new chromosomes by merging parts of existing chromosomes). Then, the fitness of the new pool of candidates is computed and the chromosomes with the highest fitness have the greatest chances to survive in the next generation. After several iterations only solutions with large fitness survive. In a work by Tasgin et al. [11], partitions are the chromosomes and modularity is the fitness function.

Genetic algorithms were also adopted by Liu et al. [12]. Here the maximum modularity partition is obtained via successive bipartitions of the graph, where each bipartition is determined by applying a genetic algorithm to each sub graph (starting from the original graph itself), which is considered isolated from the rest of the graph. A bipartition is accepted only if it increases the total modularity of the graph.

In 2009, Pizutti [13] proposed a multi-objective genetic algorithm for the detection of communities in a network. The two fitness functions used were community score and community fitness. The algorithm had the advantage that it provided a set of solutions based on the maximization of both the evaluation functions.

In section 2, the problem of Community Detection will be formulated mathematically with the introduction of two functions. In section 3, all the stages of the Genetic Algorithm such as Initialization, Fitness Functions, Mutation and Crossover will be elaborated upon. In section 4, the experimental results of BOCD will be presented and compared with existing Community Detection techniques. The Conclusion will be presented in section 5.

2 Problem Definition

A network N_w can be modeled as a graph G = *(V,E)* where *V* is a set of objects, called nodes or vertices, and *E* is a set of links, called edges, that connect two elements of *V*. A community (or cluster) in a network is a group of vertices having a high density of edges within them, and a lower density of edges between groups. The problem of detecting k communities in a network, where the number k is unknown, can be formulated as finding a partitioning of the nodes in k subsets that are highly intra-connected and sparsely inter-connected. To deal with graphs, often the adjacency matrix is used. If the network is constituted by *N* nodes, the graph can be represented with the *N* × *N* adjacency matrix A, where the entry at position *(i, j)* is 1 if there is an edge from node *i* to node *j*, 0 otherwise.

Let us introduce the concept of Community Score as a defined in [13] and [14]. Let $S \subset G$ be the sub graph where node i belongs to, the degree of i with respect to S can be split as

$$k_i(S) = k_{in}^i(S) + k_{out}^i(S) .$$

Where

$$k_{in}^i(S) = \sum\nolimits_{j\epsilon S} A_{ij} .$$

is the number of edges connecting i to the other nodes in S. Here A is the adjacency matrix of G.

$$k_{out}^i(S) = \sum\nolimits_{j\notin S} A_{ij} .$$

is the number of edges connecting i to the rest of the network. Let μ_i represent the fraction of edges connecting i to the other nodes in S.

$$\mu_i = \frac{1}{|S|} k_i^{in}(S).$$

where $|S|$ is the cardinality of S. The power mean of S of order r, $M(s)$

$$M(s) = \frac{\sum_{i \in S}(\mu_i)^r}{|S|}.$$

In the computation of $\mathrm{M(s)}$, since $0 \leq \mu_i \leq 1$, the exponent r increases the weight of nodes having many connections with other nodes belonging to the same community, and diminishes the weight of those nodes having few connections inside S.

The volume v_s of a community is defined as the number of edges connecting vertices inside S,

$$v_s = \sum_{i,j \epsilon S} A_{ij}\,.$$

The $score$ of S is defined as

$$score(S) = M(S) \times v_s\,.$$

The Community score of a clustering $\{S_1, \ldots\ S_k\}$ of a network is defined as

$$\mathrm{CS} = \textstyle\sum_{i=1}^{k} \mathrm{score}(S_i)\,. \tag{1}$$

The problem of community detection has been formulated in [14] as the problem of maximizing the Community Score. The other objective is to maximize modularity, defined in [15]. Let k be the number of modules found inside a network. The modularity is defined as

$$Q = \textstyle\sum_{s=1}^{k} \left[\frac{l_s}{m} - \left(\frac{d_s}{2m}\right)^2\right]. \tag{2}$$

where l_s is the total number of edges joining vertices inside the module s, and $\frac{l_s}{m}$ represents the fraction of edges in the network that connect the same community. d_s represents the sum of the degrees of the nodes of s. If the number of within-community edges is no more than random, we will get $Q = 0$. The maximum value of Q is 1, which indicates strong community structure.

3 Algorithm Description

The various stages of the genetic algorithm have been described in the following subsections. The framework used was NSGA-II in C described in [24].

3.1 Genetic Representation

The chromosome is represented in the format mentioned in [16]. The representation of an individual consists of N genes, and each gene can take a value in the range {1, …, N}, where N is the number of nodes in the network. If a value j is assigned to the ith gene, this suggests that i and j are in the same cluster. But if i and j are already assigned, then the gene is ignored. Thus later genes will have less bearing on cluster formation. The decoding of this individual to obtain clusters can be done in linear time according to [17].

For example, consider the individual for a network of 34 nodes (N = 34). The number in the curly brackets represents the index of the element in the individual.

{1}2, {2}3, {3}4, {4}14, {5}17, {6}17, {7}6, {8}14, {9}19, {10}19, {11}17, {12}14, {13}2, {14}9, {15}19, {16}9, {17}15, {18}8, {19}21, {20}8, {21}27, {22}1, {23}15, {24}26, {25}26, {26}32, {27}30, {28}26, {29}25, {30}3, {31}19, {32}4, {33}23, {34}9

We are assuming here that if the above individual was stored in an array, the index of the 1st element would be 1 and not 0. In the above chromosome, the element at index 1 of the array is 2. Thus nodes 1 and 2 are in the same cluster. Similarly the element at index position 2 is 3, thus 2 and 3 are put in the same cluster. Since nodes 1 and 2 are already in Cluster 1, we have nodes 1, 2 and 3 put in the same cluster. The element at index position 3 is 4, thus nodes 3 and 4 are also in the same cluster. The element at the 5th index position is 17. Since neither 5, nor 17 have been previously assigned a cluster, they are put together in a new cluster, Cluster 2. This process goes on iteratively till the last element. Finally the clusters are made as follows:

Cluster1:	1, 2, 3, 4, 14, 8, 12, 13, 18, 20, 22
Cluster2:	5, 17, 6, 7, 11
Cluster3:	9, 19, 10, 15, 16, 21, 27, 23, 30, 31, 33, 34
Cluster4:	24, 26, 25, 32, 28, 29

3.2 Initialization

The population is initialized randomly from values between 1 and N, where N is the number of nodes in the network.

3.3 Fitness Functions

The algorithm used here is a bi-objective optimization, where both fitness functions are minimized. The first fitness function is derived from equation (2).

$$\min f_1 = 1 - Q\ .$$

The 2nd fitness function uses both equations (1) and (2).

$$\min f_2 = (1 - Q) + (\frac{10}{1 + CS})$$

Q lies in the range [0,1], therefore the minimization of $(1 - Q)$ helps in finding the maximum value of modularity. In the second fitness function, the weight 10 for the Community Structure term (CS) has been found out empirically. The above pair of fitness functions taken together performs better than the single objective optimization of either of the two taken separately.

3.4 Crossover and Mutation

Simple Uniform crossover is used as the crossover operator. The crossover site is chosen at random. Selection strategy used is tournament selection, with 4 individuals contesting in the tournament.

Take for e.g.

Parent 1:	1, 2, 4, 5, 3, 5, 6, 1, 9, 4
Parent 2:	3, 6, 3, 2, 6, 4, 3, 1, 2, 9

Suppose the crossover site is randomly decided at 5. This means that the first 5 elements of Child 1 will come from Parent 1, i.e. {1, 2, 4, 5, 3}. The other elements for Child 1 will come from Parent 2, i.e. {4, 3, 1, 2, 9}. The beginning elements for Child 2 come from Parent 2 and the latter elements come from Parent 1. Thus the children formed are:

Child 1:	1, 2, 4, 5, 3, 4, 3, 1, 2, 9
Child 2:	3, 6, 3, 2, 6, 5, 6, 1, 9, 4

Mutation operator also performs simple mutation, i.e. a gene is chosen at random and its value is simply changed.

4 Experimental Results

Bi-Objective Community Detection (BOCD) is applied on 3 real world networks, the American College Football [19], Bottlenose Dolphin [26] and the Zachary Karate Club [18] network. The method is also tested on a benchmark generating program proposed in [23] which is an extension of the benchmark proposed by Girvan and Newman in [9].

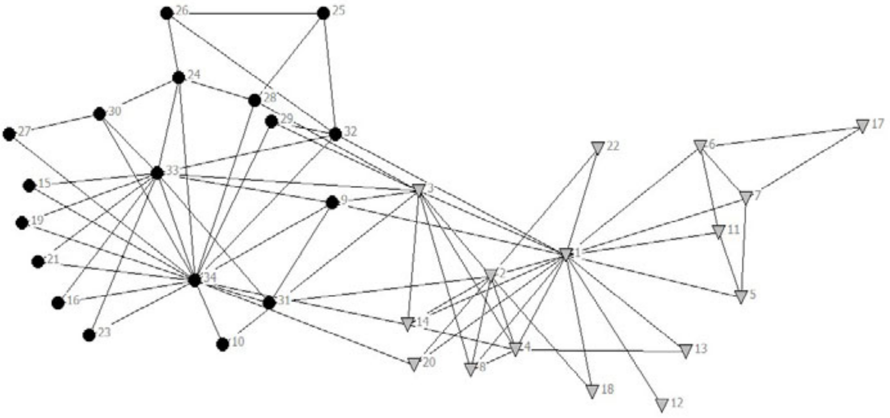


Fig. 1. The 34 node Zachary Karate Club Network divided into 2 communities. This was how the club actually broke into 2 groups. The first group is shown by circular nodes and the second by triangular nodes. The modularity of this division is 0.371.

The experiments were performed on a Core2duo machine, 2.0 Giga Hz with 3 Mb RAM. The framework for Multi-Objective Genetic Algorithm used was NSGA-II written in C described in [24]. The parameters used in compiling the code are as follows:

Population: 200
Generations: 3000
Crossover Probability: 0.7
Mutation Probability: 0.03

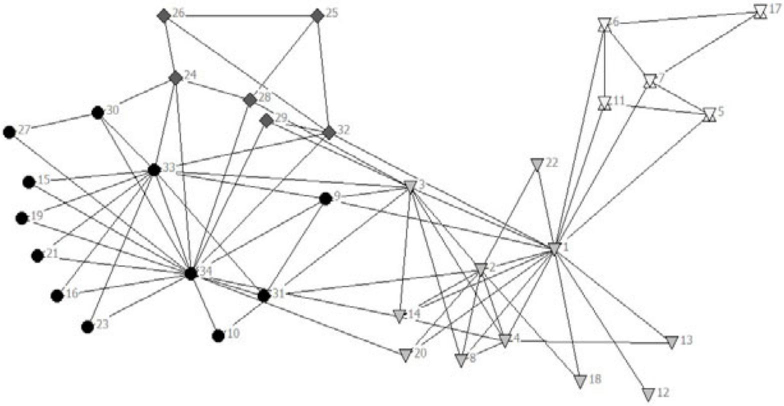


Fig. 2. Division of the Zachary Karate Club Network into 4 communities by BOCD. Each community is shown with a different symbol. The modularity of this division is 0.419.

The evaluation metrics used were Modularity which was described above, as well as Normalized Mutual Information which was described in [22]. The results obtained by BOCD are compared with the fast GN algorithm [20] and MOGA-Net [13] on the basis of Modularity and NMI.

4.1 Zachary Karate Club Network

This network was generated by Zachary [18], who studied the friendship of 34 members of a karate club over a period of two years. During this period, because of disagreements, the club divided in two groups almost of the same size. The original division of the club in 2 communities is shown in Figure 1. The BOCD algorithm divides the nodes into 4 communities, with this separation showing a higher value of modularity then the original solution itself. As can be seen from Table 1, BOCD performs better than both GN and MOGA-Net in terms of modularity. The NMI of the division was found to be 0.695622, which is better than GN algorithm but not MAGA-Net. As MOGA-Net generates a pareto set of results, they have achieved higher NMI values.

4.2 American College Football Network

The American College Football network [9] is a network of 115 teams, where the edges represent the regular season games between the two teams they connect. The teams are divided into conferences and play teams within their own conference more frequently. The network has 12 conferences or communities. The division obtained by BOCD was better than the result of MOGA-Net and was exactly on equal terms with the modularity value of the GN algorithm. The NMI of the division was found to be 0.878178, which is the highest value among the three algorithms.

4.3 Bottlenose Dolphin Network

The network of 62 bottlenose dolphins living in Doubtful Sound, New Zealand, was compiled in [26] by Lusseau from seven years of dolphin behavior. A tie between 2 dolphins was established by their statistically frequent association. The network split naturally into 2 large groups, the number of ties being 159. The performance of BOCD was much better than the GN algorithm and marginally better than that of MOGA-Net. The NMI of the division was found to be 0.615492, which lies in between MOGA-Net and GN performance wise, the best being MOGA-Net.

Table 1. Comparison of Modularity values for the 3 real datasets. The first column gives the value of modularity for NMI = 1. The following columns give modularity results for the fast GN, MOGA-Net and BOCD algorithms.

Dataset	Mod. For NMI=1	GN	MOGA	BOCD
Zachary Karate Club	0.371	0.380	0.415	0.419
College Football	0.518	0.577	0.515	0.577
Bottlenose Dolphins	0.373	0.495	0.505	0.507

Table 2. Comparison of NMI values for the 3 real datasets. The columns give NMI results for the fast GN, MOGA-Net and BOCD algorithms.

Dataset	GN	MOGA	BOCD
Zachary Karate Club	0.692	1.0	0.695
College Football	0.762	0.795	0.878
Bottlenose Dolphins	0.573	1.0	0.615

4.4 Benchmark Test Network

The network consists of 128 nodes divided into four communities of 32 nodes each. The average degree of each node is 16. The fraction of edges shared by each node with nodes in its own community is known as the mixing parameter. If the value of the mixing parameter $\mu > 0.5$, it suggests that a node will have more link to other nodes, outside its community. Thus finding community structure will be difficult for $\mu = 0.5$, as evident from the following graph. According to a graph drawn in [13], MOGA-Net could achieve an NMI of less than 0.1 for $\mu = 0.5$. Thus our algorithm performs better in case of a higher mixing parameter.

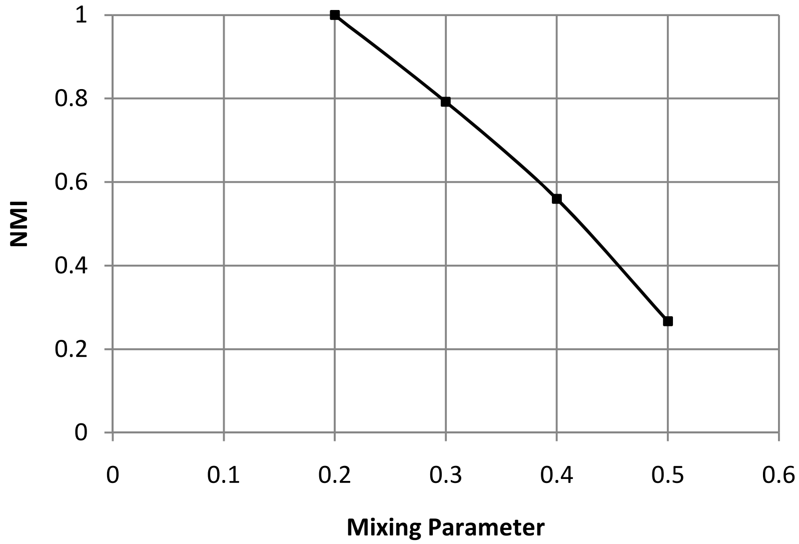


Fig. 3. NMI values obtained by BOCD for different values of mixing parameter. Here r=2.5.

Table 3. Modularity and NMI values for benchmark network with increasing value of Mixing Parameter. The performance for μ = 0.2 is the best, and expectedly deteriorates as μ is increased.

Mixing Parameter(μ)	Modularity	NMI
0.2	0.4511	1.0
0.3	0.347	0.792138
0.4	0.218	0.559844
0.5	0.181	0.266481

5 Conclusions

The paper presented a Bi-Objective Community Detection technique through the use of Genetic Algorithm. By simply combining community score and modularity, the BOCD algorithm improved upon the performance of both the GN algorithm which used Modularity and MOGA-Net which used community score in the community detection problem. Results on real life networks as well as synthetic benchmarks show the capability of this approach in finding out communities within networks. Future research should aim at decreasing computational complexity of Community Detecting algorithms and finding communities in networks with a high mixing parameter.

Acknowledgments. I would like to thank Dr S K. Gupta and Pranava Chaudhary, both of whom have profoundly helped me in understanding Genetic Algorithm and Multi-Objective Optimization. I would also like to thank Andrea Lancichinetti for providing the NMI calculating code. The software used for drawing network diagrams was Netdraw [25].

References

1. Krishnamurthy, B., Wang, J.: On network-aware clustering of web clients. SIGCOMM Comput. Commun. Rev. 30, 97–110 (2000)
2. Reddy, P.K., Kitsuregawa, M., Sreekanth, P., Rao, S.S.: A graph based approach to extract a neighborhood customer community for collaborative filtering. In: Bhalla, S. (ed.) DNIS 2002. LNCS, vol. 2544, pp. 188–200. Springer, Heidelberg (2002)
3. Agrawal, R., Jagadish, H.V.: Algorithms for searching massive graphs. IEEE Trans. on Knowl. and Data Eng. 6, 225–238 (1994)
4. Wu, A.Y., Garland, M., Han, J.: Mining scale-free networks using geodesic clustering. In: Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 719–724. ACM Press, New York (2004)
5. Perkins, C.E.: Ad Hoc Networking. Addison-Wesley Professional, Reading (2000)
6. Steenstrup, M.: Cluster-Based Networks. In: Perkins, C.E. (ed.) Ad Hoc Networking, pp. 75–138. Addison-Wesley, Reading (2001)
7. Weiss, R.S., Jacobson, E.: A method for the analysis of the structure of complex organizations. American Sociological Review 20, 661–668 (1955)
8. Rice, S.A.: The Identification of Blocs in Small Political Bodies. The American Political Science Review 21, 619–627 (1927)
9. Girvan, M., Newman, M.E.J.: Community structure in social and biological networks. Proceedings of the National Academy of Sciences of the United States of America 99, 7821–7826 (2002)
10. Holland, J.H.: Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence. MIT Press, Cambridge (1992)
11. Tasgin, M., Herdagdelen, A., Bingol, H.: Community detection in complex networks using genetic algorithms, `http://arxiv.org/abs/0711.0491`
12. Liu, X., Li, D., Wang, S., Tao, Z.: Effective algorithm for detecting community structure in complex networks based on GA and clustering. In: Shi, Y., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2007. LNCS, vol. 4488, pp. 657–664. Springer, Heidelberg (2007)
13. Pizzuti, C.: A Multi-objective Genetic Algorithm for Community Detection in Networks. In: Proceedings of the 2009 21st IEEE International Conference on Tools with Artificial Intelligence, pp. 379–386. IEEE Computer Society, Washington (2009)
14. Pizzuti, C.: GA-net: A genetic algorithm for community detection in social networks. In: Rudolph, G., Jansen, T., Lucas, S., Poloni, C., Beume, N. (eds.) PPSN 2008. LNCS, vol. 5199, pp. 1081–1090. Springer, Heidelberg (2008)
15. Newman, M.E.J., Girvan, M.: Finding and evaluating community structure in networks. Physical Review E 69, 26113 (2004)
16. Park, Y.J., Song, M.S.: A genetic algorithm for clustering problems. In: Proceedings of the 3rd Annual Conf. Genetic Programming, pp. 568–575. Morgan Kauffman, San Francisco (1998)

17. Handle, J., Knowles, J.: An evolutionary approach to Multiobjective clustering. IEEE Transactions on Evolutionary Computation 11, 56–76 (2007)
18. Zachary, W.W.: An information flow model for conflict and fission in small groups. Journal of Anthropological Research 33, 452–473 (1977)
19. Newman, M.E.J.: Modularity and community structure in networks. Proceedings of the National Academy of Sciences 103, 8577–8582 (2006)
20. Newman, M.E.J.: Fast algorithm for detecting community structure in networks. Physical Review E 69, 66133 (2004)
21. Lancichinetti, A., Fortunato, S., Kertesz, J.: Detecting the overlapping and hierarchical community structure in complex networks. New Journal of Physics 11, 33015 (2009)
22. Danon, L., Duch, J., Diaz-Guilera, A., Arenas, A.: Comparing community structure identification. Journal of Statistical Mechanics: Theory and Experiment, 9008 (2005)
23. Lancichinetti, A., Fortunato, S., Radicchi, F.: Benchmark graphs for testing community detection algorithms. Physical Review E (Statistical, Nonlinear, and Soft Matter Physics) 78, 46110 (2008)
24. Srinivas, N., Deb, K.: Multiobjective optimization using nondominated sorting in genetic algorithms. Evolutionary Computation 2, 221–248 (1994)
25. Netdraw, `http://www.analytictech.com/netdraw/netdraw.htm`
26. Lusseau, D., Schneider, K., Boisseau, O.J., Haase, P., Slooten, E., Dawson, S.M.: The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations. Behavioral Ecology and Sociobiology 54, 396–405 (2003)

A Membrane Algorithm to Stabilize a Distributed Computing System

Susmit Bagchi

Department of Informatics, Gyeongsang National University
College of Natural Sciences, Jinju, Gyeongnam 660-701, South Korea
susmitbagchi@yahoo.co.uk

Abstract. The applications of biologically-inspired computing models into the distributed computing paradigm have potential benefits offering self-detection and self-reconfiguration capabilities of the computing systems. In the large scale distributed systems, the arbitrary failure of nodes and network partitions are the major challenges in terms of failure detection, fault-tolerance and maintainability. This paper proposes a novel distributed algorithm for self-detection and self-reconfiguration of distributed systems on the event of arbitrary node failures resulting in network partitioning. The algorithm is designed based on the hybridization of biological membrane computing model and cell-signaling mechanisms of biological cells. This paper presents the problem definition, design as well as performance evaluation of the proposed algorithm.

Keywords: membrane computing, bio-inspired computing, cell-signaling, fault-tolerance, distributed algorithms.

1 Introduction

The present day distributed systems are characterized by large scale, where the number of nodes present in such a system can be scaled up to thousands. The examples of such massively scaled distributed systems are the grid systems and cloud computing systems. The main characteristics of these distributed systems are dynamicity, unreliability of nodes, possibility of network partitioning and, the unreliability of the entire system [1]. The dynamics of large scale distributed systems are difficult to manage through centralized or distributed control mechanisms [2]. In addition, the traditional assumptions and approaches to design the distributed algorithms are not suitable for direct applications to the large scale distributed systems [1]. In recent time, a growing trend to design the self-configurable systems or self-organizing algorithms for various computing systems is observed [2, 5]. The self-organizing design approach aims at creating autonomous systems capable of managing itself reliably in normal or in adverse operational conditions. It is highly desirable to design a self-managing as well as self-organizing large scale distributed

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 16–28, 2011.

systems offering reliability and transparency. It is important to model the failure of nodes as well as the resulting transformation in the distributed computing structures in order to understand the stability of the entire system. In this paper, a distributed algorithm is proposed based on the cell-signaling mechanism of biological organisms while employing diffusion pattern within membrane computing model to stabilize the distributed system on the event of arbitrary node failures. The algorithm is easily realizable. The experimental results illustrate that, the algorithm successfully stabilizes the system on the event of arbitrary failure (i.e. lysis, in biological terms) of the nodes or on self-detection of network partitions. The distinguishing features of the algorithm are, *(a) self-detection of lysis and transformation* and, *(b) self-stabilization of the system.*

1.1 Motivation

The large scale distributed systems, such as grid and cloud computing systems, are difficult to manage and control [1, 2]. The dynamic behaviour of a large distributed system is nearly impossible to monitor in details while maintaining continuous operability and performance-driven service delivery. On the other hand, failure of nodes or lysis of nodes (in biological terms) and network partitioning are the common phenomena in any distributed computing system. The traditional fault tolerant algorithms are not adequate to manage a large scale distributed system offering a complete transparency of operations as well as reliability [1, 2]. On the other hand, the self-organizing design approach is more suitable to manage and control a large distributed computing system offering reliability and transparency as compared to the traditional distributed algorithms [5]. The self-organizing adaptive systems employ the bio-computing models inspired by the functionalities of the biological organisms inclusive of multi-cellular systems [1, 2, 4]. There exists a set of applications of bio-inspired models into the computing systems. For example, the self-organizing grid computing system is developed and experimented with success [3]. In another case, the ant-colony optimization technique is used in designing the self-organizing grid [4] and peer-to-peer (P2P) systems [2].

Interestingly, any large scale distributed computing systems can be modeled as a biological cellular structure. In multi-cellular model, individual cells can be represented as individual nodes and, in single-cell model, individual nodes can be represented as cell membranes, where a set of nodes behaves as a complete cell. The biological cellular organisms exhibit remarkable properties of detecting and repairing the damage of their cellular architecture. Hence, it is highly interesting to employ the properties of self-organizing biological systems into the large scale distributed systems. To this direction, the membrane computing model [6, 7] can serve as a conceptual framework to design a self-stabilizing distributed system on the face of the lysis of nodes and the network partition. In this paper a novel distributed algorithm is proposed, which is capable of auto-controlling and stabilizing a large-scale distributed system on the event of lysis of nodes. The design approach employs single-cellular biological system model of membrane computing, where each node behaves as a constituting membrane of a cell and its intra-cellular components (i.e. mitochondria, ribosomes etc.). The diffusion pattern of bio-inspired computing model and the cell-signaling mechanism of biological cells are employed in the proposed design architecture. Rest of the paper is organized as follows. Section 2 formulates the problem definition. Section 3 describes the design and functional analysis of the algorithm. Section 4 and section 5 describe the experimental evaluations and background work, respectively. Section 6 concludes the paper.

2 The Problem Definition

The membrane structure of P system [7] is comprised of a set of cell membranes, where the membranes form a living biological cell. The membranes of the intra-cellular structures create interior regions within the cytoplasm of the cell. The basic schematic representation of the cellular membranes is illustrated in Figure 1, where c_m, m = 1,2,…7 represents the cell membranes.

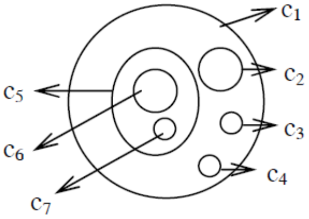


Fig. 1. The computational model of a biological cell

It is important to note that the outermost membrane of a cell (skin c_1) never dissolves. However, the intra-cellular membranes may be dissolved at any time exposing the corresponding intra-structural cellular matrix to the exterior matrix of the cytoplasm of the cell. Translating the cellular model into computing domain, the skin is numerically numbered as one (1) and the other intra-cellular membranes are numerically numbered distinctly as illustrated in Figure 1. There exists a set of evolution rules within each region of the cytoplasm and the rules control the ongoing computation of the corresponding region bounded by the membranes. The numerically numbered membrane structure can be conceptually represented as a tree in the domain of computing systems [6]. The spanning-tree representation of cellular membranes is illustrated in Figure 2.

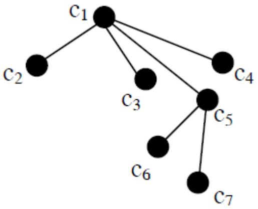


Fig. 2. Spanning tree representation of the membranes of a cell

It is observable that, any cellular membrane structure can be modeled as rooted spanning-tree, where the nodes of the tree execute the distributed program following a local set of rules as applicable to the individual nodes. Hence, each node in the spanning-tree represents the living intra-cellular structures bounded by membranes. The root of the tree is the skin of the biological cell.

2.1 Membrane Lysis and Partition

One of the main research problems in the domain of distributed computing is the unreliability of the nodes. In a highly structured distributed system, the elimination or lysis of a node (i.e. membrane) will result network partition, which may cause global inconsistency and possibly, the failure of the entire system (for example, Byzantine failure). However, according to biological cellular functions, lysis of membranes should not severely affect the overall organism. In order to model the lysis of membranes and its effects, a mathematical formulation of the phenomenon is constructed for better understanding and adapting to the distributed computing paradigm by following finite automata theory, membrane computing model and cell-signaling system.

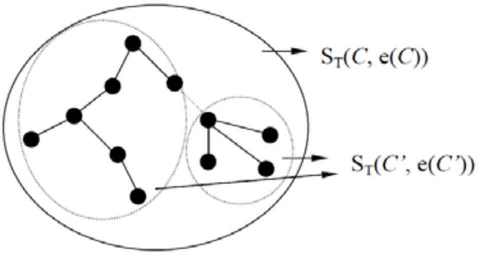


Fig. 3. Transformation of the membranes of a cell

Let, α is a finite and non-empty alphabet of objects and β is the set of possible actions to be executed by evolution rules applicable to a cell cytoplasm, given by, $\beta = \{a_k : k = 1, 2, 3....N\}$. If, $\exists g \in \alpha$ such that, $\delta(g) \rightarrow y$, where $y \in (\alpha \text{ x } \beta)^*$ then, $\delta()$ is called the lysis function and the membrane covering that region containing $\delta()$ gets dissolved after execution of $\delta()$. Suppose, $C = \langle c_m : m = 1, 2, 3, ...n\rangle$ is a cell comprised of n number of cell membranes (c_1 is the skin of the cell) then, the rooted spanning tree covering C is given by $S_T(C, e(C))$, where $e(C)$ is the set of edges in the spanning tree S_T covering nodes in C preserving the order. Let, $\Re_m : C \rightarrow \alpha$ is a relation defined over the cell such that, $\forall\ c_j, c_k \in C$ and, $\exists q, h \in \alpha$, if $u = (c_j \Re_m q)$ and $v = (c_k \Re_m h)$ then, either $u \cap v \neq \phi$ or, $u \cap v = \phi$. The segments u and v are called regions in the cell cytoplasm, where $u \cap v \neq \phi$ represents parent-child relation between u and v, in any order. Hence, $\exists\ c_k \in C$ such that, $k \neq 1$ and, $g = \Re_m(c_k)$ then, the execution of $\delta(g)$ in a cell C will transform $S_T(C, e(C))$ into $S_T(C', e(C'))$, where $C' \subset C$ and, $|e(C)|-|e(C')| \geq 1$. This indicates that, if $\prod$ is a set of partitions (π_i) in cell C' then, $\exists \pi_k \in \prod : c_1 \in \pi_k \Rightarrow c_1 \notin \pi_i, \forall \pi_i \in \prod - \pi_k$, where i, k = 0, 1, 2, …n and, $i \neq k$. The δ(g) induced transformation of a cell into the connected components of a spanning tree is illustrated in Figure 3. It is easy to observe that, δ() can induce d number of connected components in the resulting spanning tree, where $d \geq 2$. However, the transformation of S_T will inject unreliability in the distributed computing architecture. This indicates that, distributed computing based on membrane computing model needs to design a fault-tolerant distributed algorithm in order to stabilize the computation at distributed nodes while allowing arbitrary network partitioning. Further, it is important to note that, in distributed computing model, the transformation of S_T could be transient in nature given by following the timed-transition model $\langle t_q, t_r \rangle : S_T(C, e(C))|t_q \rightarrow S_T(C', e(C'))|t_r$ and $t_q < t_r$. Hence, the distributed algorithm should stabilize such transient behaviour of the transformation of the cell membranes representing the corresponding spanning tree.

3 The Algorithm Design

The distributed algorithm designed in this paper aims to incorporate self-reconfiguration and fault-tolerant capabilities following the membrane computing model. The root of the spanning tree is considered as skin membrane and the membranes of a cell can emit (i.e. output) the computation (i.e. reaction) results into the outer regions. In the reverse pathway, the computation (reaction) results can penetrate the cell membranes in order to enter into the internal regions of a cell from the exterior environments or regions. This bi-directional transport phenomenon is the basis of any working of a healthy cell involving cell-signaling and reaction activation processes [8]. In the analogous system, a node of the spanning tree sends/receives the messages to/from the other nodes connected to it in any direction. The computations going on at a node is totally internal to the node. In addition, following the concept of lysis of a membrane other than skin, any node except the root of the corresponding spanning tree can be considered to be unreliable. This indicates that the overall structure of the spanning tree can be transformed over time. It is thus required that the algorithm should stabilize two phenomena viz., (a) continuation of correct distributed computation and, (b) detecting as well as maintaining the structural integrity of the overall system by recognizing the dynamic transformation of the tree.

3.1 The Algorithm

It is evident that, depending upon the nature of $\langle t_q, t_r \rangle$, any two connected nodes in $S_T(C, e(C))$ and in $S_T(C', e(C'))$ can assume four possible instantaneous states such as, strongly-connected, moderately-connected, weakly-connected and, presently-disconnected. Hence, based upon the instantaneous characteristics of $\langle t_q, t_r \rangle$, the local view of each node will change in time. On the other hand, the components of $S_T(C', e(C'))$ should recognize and freeze computation temporarily if the components are not reachable from the skin $c_1 \in C$. The computation can restart from the checkpoint in any smaller exterior sub-trees (i.e. components not connected to skin), if the nodes of the component rejoin the component containing $c_1 \in C$. The pseudo-code representation of the algorithm is illustrated in Figure 4.

The membranes (nodes) in $S_T(C, e(C))$ and $S_T(C', e(C'))$ coordinate through the transaction of the periodic heart-beat (HB) messages (represented as msg_{HB}) originating from the skin membrane of the cell (i.e. systole cycle having time period of heart_beat_period) and the reverse transaction of instantaneous local views of each membrane to the skin (i.e. diastole cycle). Hence, each node in the spanning-tree performs periodic systole-diastole cycles executed between the skin and the other membranes through bidirectional communication in the spanning tree. Each membrane in the cell maintains a buffer (own_buffer[]) to store HB messages (i.e. msg_{HB}) received by it from the parent. On the other hand, the sets of elements in a $status\_buffer_x$ [] stored at a membrane maintain the status of connectivity of a child membrane (c_x) to the corresponding parent membrane. Upon receiving a HB message, each membrane, except the skin, will emit the local_view (i.e. updated list of connected nodes in the entire sub-tree under it) to the respective parent membrane, which contains the partial (local) view of the membrane at any point of time. Next, the node re-emits the HB messages to all of its children only once in that systole-diastole cycle as illustrated in Figure 5. This indicates that, skin membrane will have the global view of the entire cell at any time where as, the other non-leaf (intermediate) membranes will have the respective local views about the sub-tree(s) under it. The absolutely interior membranes of the cell will have no local as well as global views.

The $status\_buffer_x$ of a child membrane c_x, maintained at the parent membrane, is updated by a token (∞) in one slot of the buffer after transmitting a HB message to the child membrane from the parent in a systole-diastole cycle. On completion of a systole-diastole cycle, each membrane checks the contents of the $status\_buffer_x$ of each of its child membrane as illustrated in Figure 4.

The status of connectivity of a membrane (i.e. strongly, moderately, weakly connected or presently disconnected) is adjusted by the algorithm according to the number of tokens (z) present in the respective $status\_buffer_x$. The lower value of z with respect to capacity indicates the better connectivity at any point of time. In order to correctly construct the instantaneous local view of a membrane, the parent membrane updates the $status\_buffer_x$ of a child membrane by eliminating a token upon receiving the local view of the child membrane from it. Finally, every membrane in the spanning tree consumes one HB message from the own_buffer on generating or handling any distributed event while performing internal computations (for example, to initiate inter-process communication or to handle exceptions). A membrane freezes computing temporarily if its own_buffer contains no HB message indicating that the corresponding membrane could be possibly out of systole-diastole cycles due to $\langle t_q, t_r \rangle$, where $t_r - t_q >> 1$.

```
Cell C = $\langle c_m : m = 1, 2, 3, \ldots .n\rangle$;  // represents a distributed system
Integer :: capacity >> 0, $1< x \le n$, z; // x is the IDs of nodes in sub-tree under a node $c_m \in C$
Messages :: $msg_{HB}$;  //the heart-beat (HB) message
List :: current_children_list$_m$ <strong_connected{$c_j$}, moderate_connected{$c_k$},
        weak_connected{$c_p$}, present_disconnected{$c_s$}: $c_j, c_k, c_p, c_s \in C$ >,
        local_view$_m$<>, temp$_m$<>;
Array :: message[ ], own_buffer[capacity], status_buffer$_x$[capacity];
Constant Integer :: heart_beat_period > 0; //repeating time interval to send $msg_{HB}$
Initially: local_view$_m$ = current_children_list$_m$, temp$_m$ = $\phi$;
$\forall c_m \in C$ do {
 if (m = = 1 $\wedge$ heart_beat_period = = TIME_OUT) send $msg_{HB}$ to all children of $c_m$;
 message[ ] = receive from parent or from children of $c_m$;
 if (message[ ] = = $msg_{HB}$) {                  // a HB message from parent
         store $msg_{HB}$ message in free space in own_buffer[ ];
         send $msg_{HB}$ to each membrane in the list current_children_list$_m$;
         local_view$_m$ = temp$_m$ $\cup$ current_children_list$_m$;
         send the list local_view$_m$ to the parent of $c_m$;
         $\forall c_x \in$ current_children_list$_m$ mark one free space of status_buffer$_x$ [ ] as $\infty$;
          temp$_m$ = $\phi$;
                                  }
 else if (message[ ] = = current_children_list$_x$ from a child $c_x$ of $c_m$) {
          remove one $\infty$ from the status_buffer$_x$[ ];
         $\forall c_x \in$ current_children_list$_m$ do{
                  z = number of $\infty$ in status_buffer$_x$[ ];
                  if ($0 \le z \le$ 0.3capacity) move $c_x$ to strong_connected{$c_j$};
                  if (0.3capacity < z $\le$ 0.6capacity) move $c_x$ to moderate_connected{$c_k$};
                  if (0.6capacity < z $\wedge$ z $\neq$ capacity) move $c_x$ to weak_connected{$c_p$};
                  if (z = = capacity) move $c_x$ to present_disconnected{$c_s$};
                                                }
          temp$_m$ = temp$_m$ $\cup$ current_children_list$_x$ ;
                              }
 else if (handling computing events || sending/receiving IPC messages){
          if (own_buffer[ ] $\neq \phi$) {
                  perform IPC or handle computing events;
                  remove one $msg_{HB}$ from own_buffer[ ];
                                  }
          else stop computing and wait for $msg_{HB}$;
     }
}
```

Fig. 4. Pseudo-code representation of the distributed algorithm for membranes

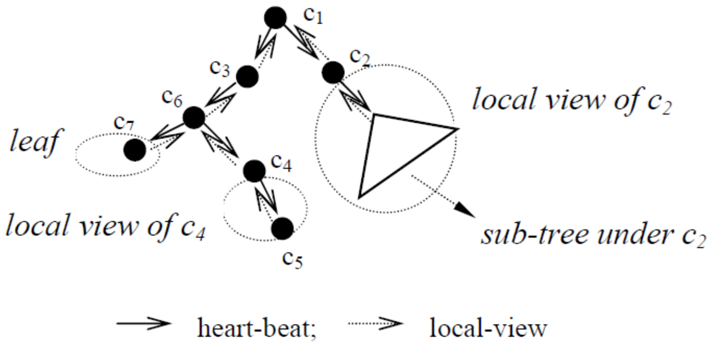


Fig. 5. A systole-diastole cycle in the cell

Hence, if t_r - $t_q \approx 0$ for some $\langle t_q, t_r\rangle$ induced on $S_T(C, e(C))$, then the structure of $S_T(C, e(C))$ will be stabilized for a sufficiently large and finite buffer space of own_buffer. On the other hand, if t_r - t_q >>1 for some $\langle t_q, t_r\rangle$ induced upon $S_T(C, e(C))$, then the resulting $S_T(C', e(C'))$ will also be stable, where distributed computation will be carried out on a component containing c_1. On the other hand, all other components will freeze (by self-recognition of partition) the computation at time t_r. The computation can be restarted later in the disconnected sub-tree, when the sub-tree will regain connection to the systole-diastole cycles at later time.

3.2 Analysis of Transformation

It is evident from the algorithm that, the dynamic variation of the availability of HB messages in own_buffer of any membrane determines the transition of the graph from $S_T(C, e(C))$ to $S_T(C', e(C'))$. The variation of the number of HB messages in the own_buffer depends upon the instantaneous network latency between the two nodes in the spanning tree, heart-beat frequency, the connectivity of two nodes and, the rate (local rate) of computation at any node. In this section, a mathematical model of dynamics of $S_T(C, e(C))$ to $S_T(C', e(C'))$ transformation is constructed based on cell-signaling network model [9]. Let, B(t) is the instantaneous number of available HB messages in the own_buffer of $n_i \in C$ at time t. Assume that, $\lambda_i(t)$ and $\lambda_o(t)$ denote the rate of input and rate of output of HB messages at n_i at any time t, where $d/dt(\lambda_i(t)) \neq 0$ and $d/dt(\lambda_o(t)) \neq 0$. If, p ($dp/dt = 0$) is the time period for estimating the residual amount of HB messages in the buffer then, in any duration between time instants ε and ε+p, $B(t)|_{t=\varepsilon+p} = \int_{\varepsilon,\varepsilon+p} (\lambda_i(t) - \lambda_o(t))\, dt$. Now, $\lim_{p\to 0} [\int_{\varepsilon,\varepsilon+p} (\lambda_i(t) - \lambda_o(t))\, dt] \approx p[\lambda_i(\varepsilon) - \lambda_o(\varepsilon)] \geq 0$. Hence, in general at any time instant t and for sufficiently small p, $d/dt(B(t)) = p[d/dt(\lambda_i(t) - \lambda_o(t))]$. This implies that, in discrete time intervals, $B(t+\Delta t)-B(t) = p[(\lambda_i(t+\Delta t) - \lambda_o(t+\Delta t)) - (\lambda_i(t) - \lambda_o(t))]$. Initially, at t = 0, $B(0) = A \geq 0$, a constant. Hence, $B(\Delta t) = p[\lambda_i(\Delta t) - \lambda_o(\Delta t)]$ for sufficiently small p. This indicates that, in any small time interval between $t_1 = x$ and $t_2 = y$, $B(y) = B(x) + [\{(\lambda_i(y) - \lambda_o(y)) – (\lambda_i(x) - \lambda_o(x))\}/ \{\lambda_i(y-x) - \lambda_o(y-x)\}]$, where y > x. It can be observed that, at some time instant x < w < y, B(w) = 0 will induce the transformation of $S_T(C, e(C))$ into $S_T(C', e(C'))$. In other words, the dynamics of availability of HB messages in the buffer of any node, controlled by the systole-diastole cycles, determines the initiation of the transformation of the spanning tree representing the membranes in a cell.

4 Experimental Evaluations

The algorithm is simulated using distributed event-driven computing model. The membranes are implemented as distributed computing nodes connected by randomly varying network conditions (link-bandwidth and congestion) forming a spanning tree representing the cell containing 200 membranes. The random variation of the network link-bandwidth, having uniform distribution, is controlled within the maxima-minima limits, where minimum is set to 3Kbps and maximum is set to 10Mbps. The congestion of the network is controlled thorough uniform probability distribution ranging from zero (0) transmission to the full-transmission (i.e. without re-transmission). The continuous trains of events are generated at the membranes to carry out distributed computation at varying frequencies (f_e), where event-frequency can vary between three levels such as, low (15 events/sec), moderate (300 events/sec) and high (1000 events/sec). Accordingly, the systole-diastole frequency (f_{HB}) is varied from low to high rate. The average transformation time (t_s) of $S_T(C, e(C))$ is measured in the cellular structure, where value of t_s indicates the time duration for first onset of structural modification in the cell *C*. The value of t_s determines the overall stability of the system. Larger is the value of t_s, more is the stability of the cell. If the transformation time is higher then, the system is more stable in the time domain. The lower value of transformation time indicates rapid transformation of the cell in a short time interval. The variation of t_s with respect to f_{HB} is illustrated in Figure 6 for constant buffer capacity (i.e. memory size).

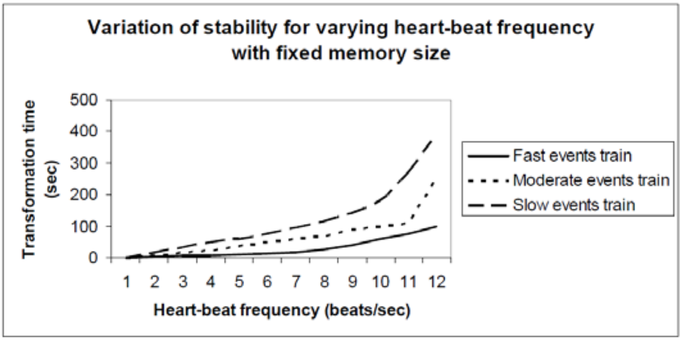


Fig. 6. Variation of stability of cell for varying f_{HB} with constant buffer capacity

It is observable in Figure 6 that, transformation time monotonically increases for any systole-diastole frequency with monotonically decreasing event-frequency. In addition, for any particular event-frequency, the corresponding transformation time monotonically increases with increasing systole-diastole frequency.

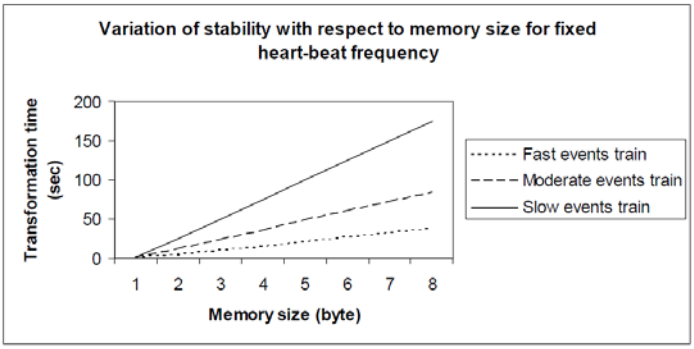


Fig. 7. Variation of stability for varying buffer capacity with constant f_{HB}

However, the rate of growth of transformation time is highest when event-frequency is lowest. In all cases, the transformation time reaches at a point of contra flexure, where the transformation time starts to increase rapidly. These points are the stability points of the cell for the corresponding event-frequency. The cell is highly stable on right side of a stability point and relatively unstable on the left side of a stability point. On the other hand, the variation of transformation time with respect to memory size (i.e. own_buffer capacity) for constant systole-diastole frequency is illustrated in Figure 7, for different event-frequencies. It is observable in Figure 7 that, with increasing memory size of a membrane and decreasing event-frequency, the transformation time tends to increase. Hence, a sufficiently large buffer allocated to a membrane, capable to accommodate a fast event train, can stabilize the system. The variations of the internal memory size, systole-diastole frequency and the transformation time of a cell are illustrated in Figure 8.

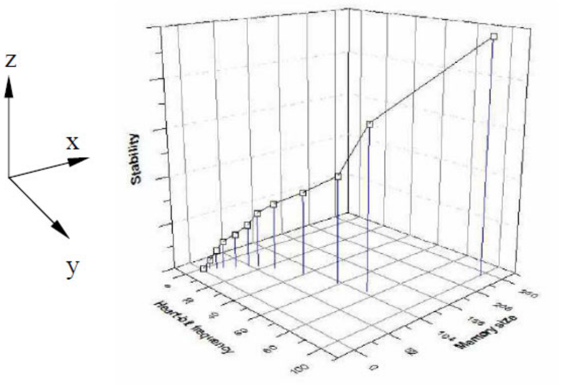


Fig. 8. Interplay of stability (z), f_{HB} (y) and memory size (x)

It is evident from Figure 8 that, the overall stability of the distributed membranes is a function of memory size and the systole-diastole frequency. If the memory size of a membrane is low then, the increasing systole-diastole frequency can stabilize the membrane of the cell to a limited extent. However, if the memory size of the membrane is increased coupled with higher systole-diastole frequency then, the stability of the cell tends to increase rapidly.

5 Background Work

In recent time, the applications of the principles of biological systems into the computing systems have gained enormous attention [1]. The main reasons are the incorporation of the self-reconfiguration capability and fault tolerance into the computing systems. The biological systems are essentially self-regulating and highly reliable in nature. Hence, the applications of the models of the biological systems into the distributed computing systems are promising [1, 2]. Researchers have proposed a set of design patterns of biological systems for the possible applications into the distributed computing systems. One of the major design patterns is the diffusion-based distributed computing model [1]. Self-chord [2] is a bio-inspired algorithm for information servicing in a structured peer-to-peer (P2P) system. In this design approach, a chord like structure is employed in the distributed computing systems (Chord, P2P, Grids) to incorporate autonomy of behaviour and self-organizing capability. On the other hand, the self-organizing grid computing systems are designed based on the ant-colony optimization techniques [3, 4]. In such a system, the individual software objects or agents follow a set of rules based upon the environmental context of the corresponding agents. Hence, there is no need to employ any centralized control mechanism. Another example of the ant-colony optimization technique based P2P system is the bio-inspired query routing mechanism [5]. In this design, the queries between peers are routed based on the ant-colony optimization techniques. The membrane computing system is another bio-inspired computing model, which can be applied to the distributed computing systems [6]. In the membrane computing model, the cellular architecture of biological systems is applied. The membrane computing model can incorporate parallelism at different levels in the large parallel and distributed systems. It is interesting to note that, membrane computing models can be directly applied to the distributed computing systems [7, 11].

However, the distributed systems are generally designed based on the message-passing semantics, where the distributed nodes are connected through a set of communication channels [12]. The message-passing between two nodes can be mapped into the cell-signaling mechanism of the biological systems [8, 9]. Hence, the message-passing semantics of distributed systems can be designed by using the cell-signaling model of the biological systems. The network partitioning is one of the major challenges in realizing distributed systems. The network partitioning can happen in a distributed system at any time, which can lead to the failure of computation. Researchers have proposed the network-partition aware CORBA system, which is fault-tolerant in nature [10]. The design approach incorporates different replication and reconciliation strategies in order to achieve the fault-tolerance. However, this design approach of partition-aware CORBA does not consider any bio-computing model.

6 Conclusion

In recent time, the bio-inspired computing has gained enormous research attention. The bio-inspired computing models can incorporate self-reorganizing and fault-tolerance capabilities into the standard computing systems. The membrane computing system is a bio-inspired computing model, which is promising in designing the distributed computing systems. However, in a distributed computing system, the message-passing model is widely employed and such systems are prone to failure on the event of network partitioning. This paper proposes a novel distributed algorithm to handle network portioning and node failures by employing the self-reconfiguration technique of biological cellular structures. The design approach of the algorithm, presented in this paper, hybridizes two different biological mechanisms. The basic design framework of the proposed algorithm follows the combination of diffusion mechanism of cellular structures and the membrane computing model, which incorporates the possibility of network-partitioning. The problem of network-partitioning is mitigated in the algorithm by using the cell-signaling mechanism of biological cells. The algorithm is capable to identify network partition and can adopt accordingly by employing self-reconfiguration mechanism of biological cellular structures. The algorithm is easy to realize. The simulated results indicate that the algorithm is capable to detect node failures and resulting network partitioning. The algorithm effectively stabilizes the distributed computing structure on self-detection of the arbitrary failure of the nodes.

References

1. Babaoglu, O., et al.: Design Patterns from Biology for Distributed Computing. ACM Transactions on Autonomous and Adaptive Systems 1(1) (2006)
2. Forestiero, A., et al.: Self-Chord: A Bio-Inspired Algorithm for Structured P2P Systems. In: 9th IEEE/ACM International Conference on Cluster Computing and the Grid (CCGRID). IEEE Computer Society Press, Los Alamitos (2009)
3. Erdil, D.C., Lewis, M.J., Ghazaleh, N.A.: An Adaptive Approach to Information Dissemination in Self-organizing Grids. In: Proceeding of the International Conference on Autonomic and Autonomous Systems, ICAS (2006)
4. Ko, S.Y., Gupta, I., Jo, Y.: A New Class of Nature-inspired Algorithms for Self-adaptive Peer-to-Peer Computing. ACM Transactions on Autonomous and Adaptive Systems 3(3), 1–34 (2008)
5. Michlmayr, E.: Self-Organization for Search in Peer-to-Peer Networks: The Exploitation-Exploration Dilemma. In: 1st International Conference on Bio-Inspired Models of Network, Information and Computing Systems. IEEE, Los Alamitos (2006)
6. Ciobanu, G.: A Programming Perspective of the Membrane Systems. International Journal of Computers, Communications & Control 1(3) (2006)
7. Ciobanu, G., Desai, R., Kumar, A.: Membrane systems and distributed computing. In: Păun, G., Rozenberg, G., Salomaa, A., Zandron, C. (eds.) WMC 2002. LNCS, vol. 2597, pp. 187–202. Springer, Heidelberg (2003)
8. Bardwell, L., et al.: Mathematical Models of Specificity in Cell Signaling. Biophysical Journal 92 (2007)

9. Eungdamrong, M., Iyengar, R.: Modeling Cell Signaling Networks. Biology of the Cell (Journal) 96(5) (2004)
10. Beyer, S., et al.: Implementing Network Partition-Aware Fault-Tolerant CORBA Systems. In: 2nd International Conference on Availability, Reliability and Security (ARES), IEEE, Los Alamitos (2007)
11. Ciobanu, G.: Distributed Algorithms over Communicating Membrane Systems. Biosystems (Journal) 70(2) (2003)
12. Pacheco, P.: Parallel Programming with MPI. Morgan Kaufmann Publisher, San Francisco (1997)

ε –Pareto Dominance Based Multi-objective Optimization to Workflow Grid Scheduling

Ritu Garg and Darshan Singh

Computer Engineering Department, National Institute Of Technology,
Kurukshetra, Haryana, India
{ritu.59,er.dssaini}@gmail.com

Abstract. Grid facilitates global computing infrastructure for user to consume the services over the network. To optimize the workflow grid execution, a robust multi-objective scheduling algorithm is needed. In this paper, we considered two conflicting objectives like execution time (makespan) and total cost. We propose a multi-objective scheduling algorithm, using ε –MOEA approach based on evolutionary computing paradigm. Simulation results show that the proposed algorithm generates multiple scheduling solutions near the Pareto optimal front with uniform spacing and better convergence in small computation time.

Keywords: Workflow Grid Scheduling, Multi-objective Optimization, MOEA, Pareto dominance, ε-Pareto dominance.

1 Introduction

With the rapid development of networking technology, grid computing [1] has emerged for satisfying the increasing demand of the computing power of scientific computing community. Geographically distributed computers which are highly dynamic and heterogeneous are used to solve complex problems from e-Science in less time. One of the key challenges of heterogeneous systems is the scheduling problem. Scheduling of computational tasks on the Grid is a complex optimization problem, which may require consideration of different scheduling criteria. Usually, the most important are the task execution time, cost of running a task on a machine, reliability, resource utilization, turnaround time, lateness etc.

The optimization of scheduling problem is NP-complete, so numerous heuristic algorithms have been proposed in literature [2]. Many heuristics have also been proposed for workflow scheduling in order to optimize a single objective [6] [7] [8]. Defining the multiple objectives for the task scheduling problem for generating efficient schedules at reduced computational times are of research interest in the recent days. In a multi-dimensional parameter space, it is in general not possible to find a solution that is best with respect to all the objectives, which makes the problem of requirements specification a real challenge. Thus, when optimizing a multi-objective function, we want to find the Pareto set [12] of solutions, which is the set, composed of all non-dominated solutions. A solution is said non-dominated if there is

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 29–40, 2011.

no other solution which optimizes one criterion without worsening another one. In this paper, we considers the two objectives for task scheduling keeping in view the tradeoff between two conflicting objectives of minimizing the makespan and total cost under the specified deadline and budget constraint.

To generate optimal scheduling solutions for workflow grid, we used NSGA-II [13] and ε –MOEA [16] approaches based on the paradigm of Multi-Objective Evolutionary Algorithms (MOEA) [12]. Rest of the paper is organized as follows. Section 2 specifies some of the related work. In section 3, we introduced the Grid Workflow Scheduling problem. Section 4, describes the technique of multi objective optimization and different multi objective evolutionary algorithms used. Section 5 discusses the simulation analysis. Finally section 6 gives the conclusion.

2 Related Work

For problem of scheduling, Directed Acyclic Graph (DAG) based task graphs in parallel computing are reported already in literature [3]. To schedule scientific workflow applications in Grid Computing environments, Heterogeneous Earliest Finish Time(HEFT) [8] and Genetic Algorithms have been applied with extension by the ASKALON project [4][9]. E. Tsiakkouri et al [7] proposed two scheduling approaches namely LOSS and GAIN to make adjustment in the schedule generated by a time optimized heuristic and a cost optimized heuristic within the users' specified budget constraint respectively. Bi-criteria workflow scheduling algorithm that performs optimization based on a flexible sliding constraint has been proposed in [5], where the concept of dynamic programming is used in order to explore the search space effectively. J. Yu and Buyya [6] developed time optimization and cost optimization algorithms based on the genetic algorithms within the budget and deadline constraints respectively. In the paper [18], effectiveness of Evolutionary Algorithms as compared to Simulated Annealing and Particle Swarm Optimization has been shown for scheduling jobs on Computational Grids. Furthermore, the Multi-Objective Evolutionary Algorithms (MOEAs) for workflow scheduling have been investigated to optimize two conflicting objectives simultaneously [10], [11]. The ε-constraint classic Optimization method [17] has been applied in grid scheduling on independent tasks by considering makespan and flow-time objectives. The work presented in [19] addresses tradeoff between execution time and reliability.

Unlike the aforementioned work, we have proposed the use of ε –MOEA [16] approach in workflow execution for two major objectives makespan and cost. With ε-MOEA approach a set of trade-off schedules have been generated closed to the Pareto-optimal front; uniformly spaced and the algorithm convergence is fast .We have also simulated NSGA II for the comparison with ε – MOEA approach.

3 Scenario of Workflow Grid Scheduling

We define workflow Grid scheduling as the problem of assigning different available grid resources to precedence constraint tasks in the workflow. We model the application as a task graph: Let *G = (V, E)* be a directed acyclic graph with *V* as the set of *m* tasks i.e. each task $t_i \in V$, $1 \le i \le m$ and *E* is the set of *e* edges representing precedence constraint among the tasks i.e. each edge $(t_i, t_j) \in E$, $1 \le i \le m$, $1 \le j \le m$, $i \ne j$. In the task graph, a task without any predecessor is called an *entry task* and a task without any successor is called an *exit task*. Let a set *R* represents the *n* number of resources available in the grid where each resource $r_j \in R$. To process the tasks in a task graph, three matrices are required. First, an *m* × *m Execution Time* matrix in which each entry $w_{i,j}$ gives estimated execution time to complete task t_i on resource r_j. Second, a *m* × *m Data* matrix, where each entry $data_{i,k}$ is the units of data required to be transmitted from task t_i to task t_k. Last an *n* × *n Data Transfer Time* matrix represents the data transfer time (for a data unit) between two resources i.e. each entry $B_{s,t}$ is used to store the time required to transfer a data unit from processor r_s to r_t. Furthermore, a vector *Cost* in which entry $cost_j$ specifies cost of using the resource in per unit of time.

Before discussing the objective functions, it is necessary to define some attributes. Let, a task t_i is to be scheduled on resource r_j then the attributes $ST(t_i)$ and $FT(t_i)$ represent the starting time and finish time of a task t_i on resource r_j respectively. These are defined by:

$$ST(t_i) = \max_{t_p \in pred(t_i)} \{FT(t_p) + CT_{p,i}\} . \tag{1}$$

$$FT(t_i) = ST(t_i) + w_{i,j} . \tag{2}$$

Where $pred(t_i)$ is the set of immediate predecessor tasks of task t_i and $CT_{p,i}$ is the total communication time required to transfer data units from task t_p (scheduled on resource r_s) *to* task t_i(scheduled on resource r_j) ,which is calculated as follows:

$$CT_{p,i} = data_{p,i} \times B_{s,j} . \tag{3}$$

For the entry task t_{entry} , the *ST* is defined by:

$$ST(t_{entry}) = 0 . \tag{4}$$

For the other tasks in the task graph, the ST and FT are computed recursively, starting from the entry task, as shown in Equation (1) and (2), respectively. In order to solve the workflow optimization problem, we define two objectives time (makespan) and cost (total cost) for a schedule S as below:

$$\text{Minimize } Time(\mathrm{S}) = FT(t_{exit}) . \tag{5}$$

$$\text{Minimize } Cost(\mathrm{S}) = \Sigma\, C_{i,j} . \tag{6}$$

$$\text{Subject to } Cost(\mathrm{S}) < \mathrm{B} \text{ and } Time(\mathrm{S}) < \mathrm{D} . \tag{7}$$

Where $FT(t_{exit})$ is the finish time of exit task of a task graph and B is the cost constraint (Budget) and D is the time constraint (Deadline) required by users for workflow execution. And $C_{i,j}$ is the cost of executing a task t_i on resource r_j and is calculated as follows:

$$C_{i,j} = Cost_j \times w_{i,j} . \tag{8}$$

4 Evolutionary Multi-objective Workflow Grid Scheduling

4.1 Multi-objective Optimization

The multi-objective optimization problem [12] can be stated as the problem of simultaneously minimizing or maximizing multiple objectives which are conflicting in nature. In this paper, both makespan and total cost are minimization objective, so we present optimization accordingly. Generally, multi-objective optimization is not restricted to find a single solution, rather than a set of solutions called Pareto or non-dominated solutions. All non-dominated solutions in the objective space collectively called Pareto-optimal front. Finding Pareto optimal front of a problem is the main concern of multi-objective optimization.

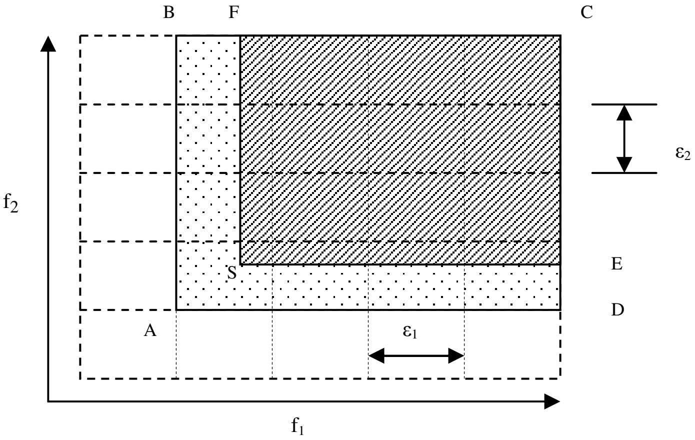


Fig. 1. The concept of usual dominance and ε-dominance

There are two relations in multi-objective optimization problem called usual Pareto Dominance and ε-Pareto Dominance, which are stated as:

Relation1: usual Pareto Dominance
Let f(s) = (f_1(s), f_2(s) …., f_m(s)) consists of m objectives. Consider two vector solutions s_1 and s_2. Then solution s_1 is said to dominate s_2 iff following two conditions hold:

1. $\forall i \in \{1,2,...m\} : f_i(s_1) \le f_i(s_2)$
2. $\exists j \in \{1,2,...m\} : f_j(s_1) < f_j(s_2)$

Relation2: ε-Pareto Dominance
Solution s_1 is said to ε-Dominate s_2 iff following two conditions hold:

1. $\forall i \in \{1,..m\} : \lfloor f_i(s_1)/\varepsilon_i \rfloor \le \lfloor f_i(s_2)/\varepsilon_i \rfloor$
2. $\exists j \in \{1,...m\} : \lfloor f_j(s_2)/\varepsilon_j \rfloor < \lfloor f_j(s_2)/\varepsilon_j \rfloor$

As depicted in Fig. 1, usual domination allows solution S to dominate only the area SFCES, whereas the definition of ε-dominance allows solution S to ε-dominate the entire area ABCDA.

4.2 Evolutionary Algorithms

There are many well known evolutionary algorithms to solve the multi-objective problems effectively. In this paper we presented two multi-objective algorithms NSGA-II and ε-MOEA to solve the grid workflow scheduling problem.

K. Deb and his students proposed an elitist non-dominated sorting genetic algorithm (NSGA-II) [13] which is the improved version of NSGA. NSGA-II evaluates solutions based on the values of each objective by ensuring elitism and diversity among the solutions. The NSGA II procedure is outlined as follow:

```
NSGA-II procedure
 begin
    Generate initial random population P(0) of size N;
     Evaluates members of P(0) according to their
     objective functions;
     t := 0;
     repeat
       Apply selection, Crossover and mutation on P(t) to
       produce offspring Q(t);
       Make combined population R(t) := P(t) U Q(t);
       Apply fast-non-dominated-sorting on R(t)to create
       Pareto fronts and assign a rank to each member of
       R(t) based on their Pareto front;
       Assign crowding distance to each member of R(t);
       Include N members of R(t) in P(t+1) according to
       their rank and crowding distance;
       t := t + 1;
     until t < maximum number of generations;
     Report final members of P as obtained solutions;
 end.
```

In NSGA II, initially a random parent population P1 of size N is generated and each solution or schedule is assigned a fitness value using some fitness function. The offspring population Q1 is then created by applying genetic operations i.e. selection, crossover and mutation on parent population P1. The procedure is different from the first generation onward; first the two populations Pt and Qt are combined to form population Rt of size 2N in the t^{th} generation. Then, a fast-non-dominated-sorting [13] is applied on population Rt. Since all previous and current population members are included in Rt, elitism is ensured. The new parent population Pt+1 is created by adding solutions from the first front F1 (Best front) if size of F1 is smaller than N. The remaining members of population Pt+1 are selected from next non-dominated from i.e. solutions form F2, followed by solutions from F3 and so on. Thereafter, last non-accepted fronts' solutions are sorted based on crowding distance measurement [13] and less crowded solutions of the sorted front are included until size of population Pt+1 exceeds N. Now, selection, crossover and mutation operations are performed on this population Pt+1 to create new child population Qt+1. This algorithm is repeated until maximum number of generations M.

We applied another evolutionary approach called ε-MOEA [16] which is based on the ε-dominance strategy. The ε-dominance does not allow non-dominated solutions having difference less than ε_i in the i^th^ objective, therefore maintaining diversity between solutions of the population as shown in Fig. 1. The procedure of ε-MOEA with one offspring per generation is presented in the following.

```
ε-MOEA procedure
  begin
     Generate initial random EA population P(0)of size N;
     Make archive population E(0) from ε-non-dominated
     solutions of P(0);
     t := 0;
     repeat
       Choose a solution p from two randomly selected
       solutions of P(t) based on usual non-domination;
       Choose a random solution e from E(t);
       Produce offspring c by performing crossover and
       mutation on p and e solutions;
       Accept/reject c in P(t) according to usual
       domination checking with members of P(t);
       Accept/reject c in E(t) according to ε-domination
       checking with members of E(t);
       t := t + 1;
     until t < maximum number of generations;
     Report final archive members as obtained solutions;
  end.
```

In the ε-MOEA, two populations, an EA population P(t) and an archive population E(t) are evolved simultaneously and independently. Here, t is the generation counter. The ε-MOEA starts with an initial population P(0) which is generated randomly and the initial archive population E (t) is created with ε-non-dominated solutions of P(0). Thereafter, two solutions (one each from P(t) and E(t)) are selected for crossover and mutation operations. To select a solution from P(t), two population members of P(t) are picked up randomly and a dominance checking (usual Pareto dominance) is performed between them. The solution which dominates the other is chosen. Otherwise, one solution is chosen randomly if both solutions are non-dominated to each other. Let, this selected solution is called p. To choose another solution for crossover and mutation, a random solution is picked from E(t). Let, this solution is denoted by e. However, other strategies can be applied to select a solution e. After the selection of solutions p and e, m offsprings solutions are generated by applying crossover and mutation operation on them. The offspring solutions are denoted by c_i (where i = 1, 2, 3,…, m). Now, all produced offspring are compared with EA and archive population for their possible inclusion. The above procedure is continued till the number of generations and the final solution in the archive population E(t) are reported as the obtained solutions.

The complete detail that how an offspring c is accepted/ rejected in P(t) and E(t) can be found from the base paper of ε-MOEA [16].

4.3 Methodology

In order to apply evolutionary algorithms to solve the workflow scheduling problem, we have to formulate population, fitness function development, and genetic operators i.e. selection, crossover and mutation. The methods we have used are described in the following sub-sections.

4.3.1 Population Formulation

An individual of the population is formulated with two strings called task matching string (MS) and scheduling order string (SS). The MS is formed by assigning each task randomly to a resource. E.g. MS (t) = r means task t is assigned to resource r. The SS denotes the scheduling sequence of tasks which are matched on the same resource. SS is also randomly generated while preserving the precedence relation between workflow tasks. Two fitness functions $F_{TIME}(S)$ and $F_{COST}(S)$ are formed in order to evaluate individuals according to makespan and total cost respectively. These fitness functions are calculated from Equation (5) and (6) by adding the penalty. On the violation of deadline and budget constraints, penalty is added respective to objective function, otherwise not.

4.3.2 Selection Operator

Individuals are selected according to their fitness value of both objectives. We used binary tournament selection due to it's widely use in the past. In binary tournament selection one of two randomly individuals is selected based on their fitness value. Thus individual having good fitness value get more chance to be survive in the next generation.

4.3.3 Crossover and Mutation Operators

Crossover produces new individuals from the existing ones by interchanging machines (resources) of them. We have used one point crossover, which showed good performance for workflow scheduling problem. Mutation operator is used to explore new things which could not be exploited by crossover operator. In mutation, a task of the individual is reassigned on another resource randomly. Mutation operator used here is replacing mutation. We have applied crossover and mutation only on matching string.

5 Evaluation and Discussion of Results

We used GridSim [20] toolkit to simulate workflow task scheduling for multi-objective optimization. GridSim is a java based toolkit for modeling and simulation of resource and application scheduling in large-scale parallel and high performance distributed computing environment such as Grid.

In our experiments, we simulated complex workflow applications consisting of 20 tasks on 8 resources and these resources are maintained by different organizations in the grid. Two single objective optimization algorithms HEFT [8] and Greedy Cost are used to make deadline and budget effectively. HEFT is a time optimization workflow scheduling in which tasks are matched and scheduled based on minimum execution time of resources irrespective of resource's cost. Greedy Cost is a cost optimization workflow scheduling algorithm in order to match and schedule tasks on cheapest resources. Thereby, HEFT gives minimum makespan M_{min} and maximum total cost TC_{max} of the workflow schedule. And Greedy Cost gives maximum makespan M_{max} and minimum total cost TC_{min} of the workflow schedule. Thus Deadline and Budget are formulated as:

$$Deadline = M_{max} – C\ (M_{max}\ \ - M_{min})\ . \tag{9}$$

$$Budget = TC_{max} – C\ (TC_{max}\ \ - TC_{min})\ . \tag{10}$$

The value of parameter C can lies between 0.1 and 0.7. For both deadline and budget, we used 0.1, 0.4 and 0.7 to make loose, intermediate and stiff constraints respectively. The default parameter setting for our simulated MOEA approaches is mentioned in Table 1. Further, ε-value for ε-MOEA was varied from 0.0145 to 0.02 to control the diversity and extent of obtained solutions.

Table 1. Default Setting for Evolutionary Algorithms

Parameter/Operation	**Value/Type**
Population size	10
Number of generations	200
Population initialization	Random
Crossover rate (NSGA-II)	0.9
Mutation rate (NSGA-II)	0.5
Selection operator	Binary tournament
Crossover operator	One- point crossover
Mutation operator	Replacing mutation
Number of child per generation (ε-MOEA)	10

The Pareto optimal solutions of NSGA-II and ε-MOEA, obtained after 200 generations at loose, intermediate and stiff constraints are shown in Fig. 2, 3 and 4.

The results of Fig. 2 show that most of the solutions obtained with ε-MOEA are lie on the better front as compared to NSGA-II while preserving uniform diversity between solutions. All obtained Solutions at intermediate constraint of ε-MOEA shown in Fig. 3 are far better than NSGA-II in terms of both Pareto front and diversity among solutions. In the Fig. 4, obtained solutions of ε-MOEA are also good than NSGA-II. Here, number of obtained solutions of both approaches is less as compared to obtained solutions in Fig. 2 and Fig. 3 because only few solutions meet stiff deadline and budget constraints.

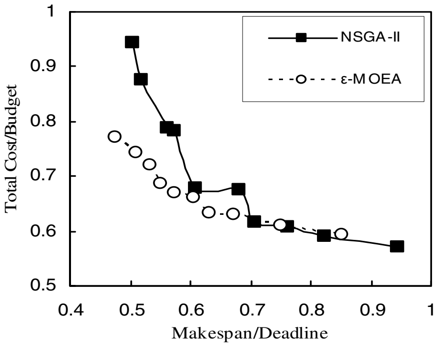


Fig. 2. Obtained Pareto Optimal front on loose constraint

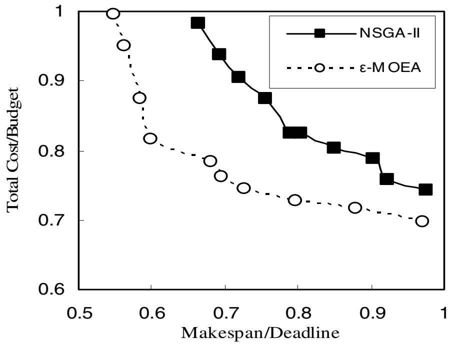


Fig. 3. Obtained Pareto Optimal front on intermediate constraint

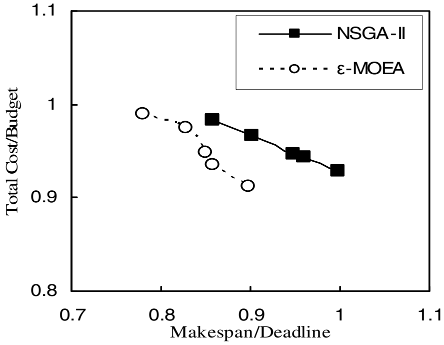


Fig. 4. Obtained Pareto Optimal front on stiff constraint

5.1 Performance Evaluation: GD, Spacing and Computational Time

For the performance comparison between NSGA-II and ε-MOEA, we conducted our experiment over 5 runs and then average of these runs has been taken for evaluation. To measure the quality of evolutionary algorithms, we used two metrics Generational Distance (GD) [21] and Spacing [21]. GD is the well known convergence metric to evaluate the quality of an algorithm against the reference front P*. The reference front P* was obtained by merging solutions of both algorithms over 5 runs. On the other side, Spacing metric was also used to evaluate obtained solutions of NSGA-II and ε-MOEA in order to check uniform diversity among solutions. Mathematically GD and Spacing metric are expressed in equation (11) and (12).

$$GD = \frac{\left(\sum_{i=1}^{|Q|} d_i^2\right)^{1/2}}{|Q|} \quad (11)$$

$$Spacing = \sqrt{\frac{1}{|Q|}\sum_{i=1}^{|Q|}(d_i - \bar{d})^2} \quad (12)$$

In Equation (11), d_i is the Euclidean distance between the solution of Q and the nearest solution of P*. Q is the front obtained using algorithm for which we calculate GD metric. In Equation (12), d_i is the distance between the solution and its nearest solution of Q and it is different from Euclidean distance. And, $\bar{d}$ is the mean value of the distance measures d_i. The small value of both GD and Spacing metric is desirable for an evolutionary algorithm. Further, we normalized Euclidean distance and distance value before using them in Equation (11) and (12) because both objectives in our problem are on different scale.

Table 2. GD, Spacing metric results along with computation time on three constraints

Constraint	Evolutionary Algorithm	GD metric	Spacing metric	Time (nanoseconds)
Loose	NSGA-II	0.071887	0.075533	218570070
	ε -MOEA	0.026443	0.043269	128062142
Intermediate	NSGA-II	0.079683	0.066143	212192220
	ε -MOEA	0.034225	0.052019	128798604
Stiff	NSGA-II	0.143163	0.121573	221491614
	ε -MOEA	0.067345	0.091872	118246202

The results obtained with these metrics along with computational time taken by each algorithm are depicted in Table 2. The average GD metric value of ε-MOEA is less as compared to NSGA-II for all constraints loose, intermediate and stiff. It means ε-MOEA has better convergence power than NSGA-II. On the other side, average value obtained with Spacing metric of ε-MOEA is also less than NSGA-II for all the constraints as presented in Table 2 which clearly state the better diversity among the solutions generated by ε-MOEA. Table 2 also shows the average computation time (over 5 runs) taken by ε-MOEA is much less as compared to NSGA-II.

6 Conclusion and Future Work

Multi-objective optimization is studied and applied in this paper for workflow grid task scheduling. Multiple trade-off Pareto optimal solutions are obtained so that user can make decision according to his/her choice. NSGA-II and ε-MOEA two well known evolutionary algorithms are applied on two important conflicting objectives makespan and total cost of the schedule in this paper. Simulation results obtained with NSGA-II and ε-MOEA are presented and evaluated according to convergence and diversity metrics. Finally we concluded that ε-MOEA is a good compromised evolutionary algorithm for workflow grid task scheduling in terms of convergence towards better Pareto optimal front, uniformly distributed solutions with small computation overhead.

In future, NSGA-II and ε-MOEA approaches can be applied for more than two objectives in grid. Further, power of other multi-objective evolutionary algorithms can be explored in grid workflow task scheduling to produce multiple optimal solutions.

References

1. Foster, I., Kesselman, C.: The Grid: Blueprint for a New Computing Infrastructure. Morgan Kaufmann, San Francisco (1999)
2. Braun, T., Siegal, H., Beck, N.: A comparison of Eleven Static Heuristics for Mapping a Class of Independent Tasks onto Heterogeneous Distributed Computing Systems. Journal of Parallel and Distributed Computing 61, 810–837 (2001)
3. Wang, L., Siegel, H., Roychowdhury, V., Maciejewski, A.: Task Matching and Scheduling in Heterogeneous Computing Environments using a Genetic-Algorithm-Based Approach. Journal of Parallel Distributed Computing 47, 9–22 (1997)
4. Wieczorek, M., Prodan, R., Fahringer, T.: Scheduling of Scientific Workflows in the SKALON Grid Environment. SIGMOD Rec., 34, 56–62 (2005)
5. Wieczorek, M., Podlipning, S., Prodan, R., Fahringer, T.: Bi-criteria Scheduling of Scientific Workflows for the Grid. IEEE, Los Alamitos (2008) 978-0-7675-3156-4/08
6. Yu, J., Buyya, R.: Scheduling Scientific Workflow Applications with Deadline and Budget Constraints using Genetic Algorithms. Scientific Programming 14, 217–230 (2006)
7. Tsiakkouri, E., Sakellariou, R., Zhao, H., Dikaiakos, M.: Scheduling Workflows with Budget Constraints. In: CoreGRID Integration Workshop, Pisa, Italy, pp. 347–357 (2005)

8. Haluk, T., Hariri, S., Wu, M.: Performance-Effective and Low-Complexity Task Scheduling for Heterogeneous Computing. IEEE Transactions on Parallel and Distributed Systems 13, 260–274 (2002)
9. Prodan, R., Fahringer, T.: Dynamic scheduling of Scientific Workflow Applications on the Grid: A case study. In: SAC 2005, pp. 687–694. ACM, New York (2005)
10. Yu, J., Kirley, M., Buyya, R.: Multi-objective Planning for Workflow Execution on Grids. In: Proceedings of the 8th IEEE/ACM International conference on Grid Computing (2007), ISBN:978-1-4244-1559-5, doi:10.1109/GRID.2007.4354110
11. Talukder, A., Kirley, M., Buyya, R.: Multiobjective Differential Evolution for Scheduling Workflow Applications on Global Grids. John Wiley & Sons, Ltd, Chichester (2009), doi:10.1002/cpe.1417
12. Deb, K.: Multi-Objective Optimization using Evolutionary Algorithms. Wiley & Sons, England (2001)
13. Deb, K., Pratap, A., Aggarwal, S., Meyarivan, T.: A Fast Elitist Multi-Objective Genetic Algorithm: NSGA-II. In: Deb, K., Rudolph, G., Lutton, E., Merelo, J.J., Schoenauer, M., Schwefel, H.-P., Yao, X. (eds.) PPSN 2000. LNCS, vol. 1917, pp. 553–562. Springer, Heidelberg (2000)
14. Zitzler, E., Laumanns, M., Thiele, L.: SPEA2: Improving the Strength Pareto Evolutionary Algorithm for Multiobjective Optimization. In: Giannakoglou, K.C., Tshalis, D.T., Periaux, J., Papailion, K.D., Fogarty, T. (eds.) Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems, Athens, Greece, pp. 95–100 (2001)
15. Knowles, J., Corne, D.: The Pareto Archive Evolution Strategy: A New Baseline Algorithm for Multi-Objective Optimization. In: The Congress on Evolutionary Computation, pp. 98- 105 (1999)
16. Deb, K., Mohan, M., Mishra, S.: A Fast Multi-objective Evolutionary Algorithm for Finding Well-Spread Pareto-Optimal Solutions. KanGAL Report Number: 2003002, Indian Institute of Technology, Kanpur, India (2003)
17. Camelo, M., Donoso, Y., Castro, H.: A Multi-Objective Performance Evaluation in Grid Task Scheduling using Evolutionary Algorithms. In: Applied Mathematics and Informatics (2010) ISBN: 978-960-474-260-8
18. Grosan, C., Abraham, A., Helvik, B.: Multiobjective Evolutionary Algorithms for Scheduling Jobs on Computational Grids. In: International Conference on Applied Computing, pp. 459-463, Salamanca, Spain (2007) ISBN 978-972-8924-30-0
19. Dogan, A., Ozguner, F.: Biobjective Scheduling Algorithms for Execution Time-Reliability trade-off in Heterogeneous Computing Systems. Comput. J. 48(3), 300–314 (2005)
20. Buyya, R.: GridSim: A Toolkit for Modeling and Simulation of Grid Resource Management and Scheduling, `http://www.buyya.com/gridsim`
21. Deb, K., Jain, S.: Running Performance Metrics for Evolutionary Multi-objective Optimization. In: Simulated Evolution and Learning (SEAL 2002), pp. 13–20 (2002)

Morphology Based Feature Extraction and Recognition for Enhanced Wheat Quality Evaluation

Manish Chhabra and Parminder Singh Reel

Electronics and Communication Engineering Department
Thapar University, Patiala-147001, India
manish_chh@yahoo.co.in, parminder.reel@gmail.com

Abstract. Wheat grain quality assessment is important in meeting market requirements. The quality of the wheat can be judge by its length, thickness, width, area, etc. In this paper on the basis of simple mathematical calculations different parameters of a number of wheat grains are calculated. The present paper focused on the classification of wheat grains using morphological. The grain types used in this study were Hard Wheat, Tender Wheat. In this paper the application of neural network is used for assessment of wheat grain. The contours of whole and broken grains have been extracted, precisely normalised and then used as input data for the neural network. The network optimisation has been carried out and then the results have been analysed in the context of response values worked –out by the output neurons.

Keywords: Wheat Quality Assessment, Image Recognition, Feature Extraction, Image Segmentation, neural network, MATLAB GUI.

1 Introduction

The past few years was marked by the development of researches that contribute to reach an automatic classification of cereal grains which is perceived as a possible solution to prevent human errors in the quality evaluation process. There are various methods in the quality control which can replace the human operator. One of these methods includes Computer vision system. After hours of working the operator may lose concentration which in turn will affect the evaluation process. So a computer vision system proved to be more efficient at the level of precision and rapidity. But it is a complex work to classify various cereal grains because of their natural diversity in appearance.

Many researches were carried out to classify cereal grains. Characterization models were based on morphological features ([1– 9]), color features ([10–13]) or textural features ([14]). Other researchers ([15–17]) have tried to combine these features for the sake of improving the efficiency of classification. Recently, wavelet technique was integrated in cereal grains characterization ([18, 19]). This technique, developed by Mallat [20], is used in textural image analysis to make object classification more precise. Wheat grain has a complex structure that includes a crease on one side. There are a number of quantitative characteristics of wheat grains, such as length, width,

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 41–50, 2011.

thickness, and the presence or absence of a stain called black point which is useful for predicting grain quality. The definitions of grain length, width, and thickness are illustrated in section 3 under feature analysis and extraction. Ordinary 2D imaging systems can only see the length and width rather than the thickness of grains because they tend to lie either ‘‘crease up’’ or ‘‘crease down’’. The thickness measurement together with the length and width measurements can also be used for calculating grain roundness or plumpness and for parametric modelling of 3D wheat grain morphology. The assessment of wheat grain quality is being done at all the stages of its production, processing and storage. Small grains and impurities are removed using mechanical sieves. Still such a procedure does not guarantee an efficient removal of all the broken grains. Considerable quantity of damaged grains in the wheat affects its quality as a seed, and also as a material for further processing. The quality evaluation of food and agricultural products is done mainly by their visual evaluation, carried out by qualified experts. It is expensive, time-consuming and subjective to evaluate and analyze a huge number of samples. Computer image analysis is one of the fields which enable an objective evaluation of grain materials. It has been applied for quality assessment of edible beans [21], rice [22], pistachio nuts as well as wheat grain.

The classification systems used for qualitative evaluation of the examined samples are mainly based on the analysis of geometrical parameters of the studied objects (length, width, area, circumference, shape coefficients. This paper only presents the techniques to extract the features like length, area, diameter from a wheat image and to recognize that whether a sample is a wheat grain or not.

2 System Design

This system is divided into some section in order to support the feature extraction process. The various processing steps for analysis are mentioned in the following flow chart.

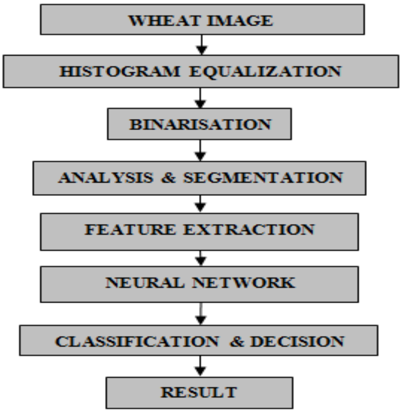


Fig. 1. The proposed approach for Wheat recognition

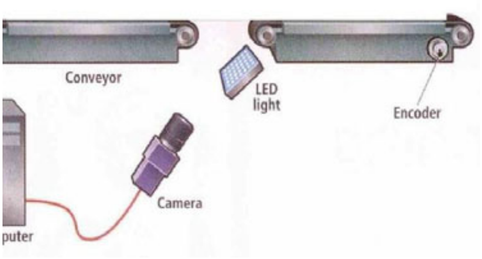


Fig. 2. Complete setup with Conveyor belt

2.1 Converting RGBIimage to Binary Image

Sony TX color digital camera with 3008 X 2000 pixels (with 16 bits for each channel) is used for image acquisition. Only lightness information was used. The lightness was obtained from the color image by averaging the three channels. Full color information was used in other parts of a larger project for black point detection.

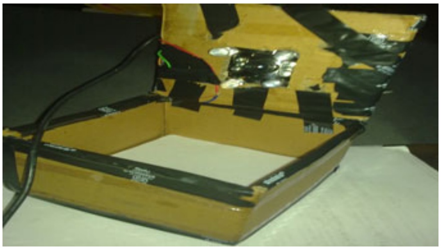


Fig. 3. Setup used for clicking the photos

Fig. 2 shows the setup used for clicking the photos of wheat including the conveyor belt, computer, camera, LEDs. Fig. 3 shows the setup used. This is a box 25*25*25 made up of cardboard and pasted white sheet paper so that the light falling on it is reflected and the absorption should be less. Then the camera was mounted on the top of the box and LEDs were attached around camera for illumination. The main thing here is that the position of camera and the LEDs and the distance between the base of the box and the camera is not changed throughout the experiment. Initially different quantity of wheat was taken. For example photographs of 1, 2, and 3 to 15 wheat's were taken by adding them one by one wheat in the box. An RGB image is firstly converted into a grayscale image. Eq. 1 is the formula used to convert RGB value of a pixel into its grayscale value.

$$\text{gray} = 0.2989*R + 0.5870*G + 0.1140*B \tag{1}$$

Where R, G, B correspond to the colour of the pixel, respectively. An example of a sample of three wheat seeds is taken. Fig4 (b) shows the gray scale image of the input image 4(a).

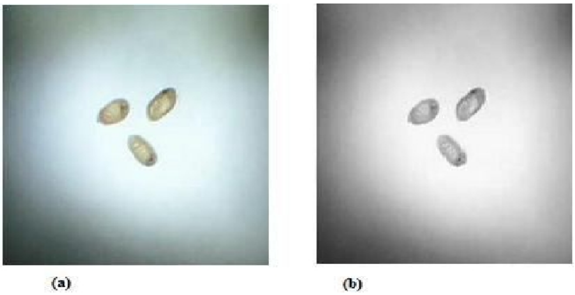
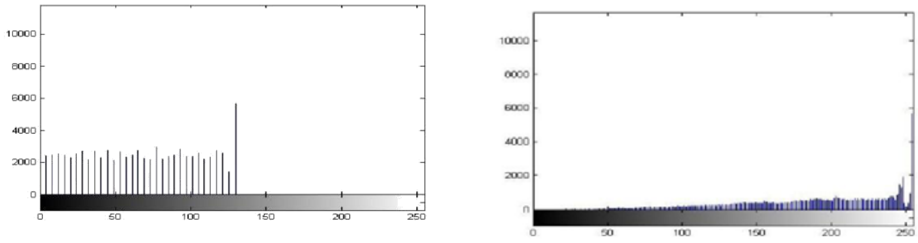


Fig. 4. Images of wheat grains (a) original image, and (b) gray image

Gray scale image is obtained but the image intensity is not proper. So there is a need to enhance the image. Image enhancement is used through histogram equalization. It enhances the contrast of images by transforming the values in an intensity image, or the values in the colour map of an indexed image, so that the histogram of the output image approximately matches a specified histogram. Fig. 5 shows the histograms of the two grayscale images (a) without applying histogram equalization and (b) with applying equalization.



(a) (b)

Fig. 5. Images of histogram (a) without equalization, and (b) with equalization

The level to convert grayscale into binary image is determined according to the RGB histogram. The output image replaces all pixels in the input image with luminance greater than the level by the value 1 and replaces all other pixels by the value 0. Fig. 6(a) shows the binary image of the equalized image. A rectangular averaging filter of size 3 × 3 is applied to filter noises. Then pixel values are rounded to 0 or 1.

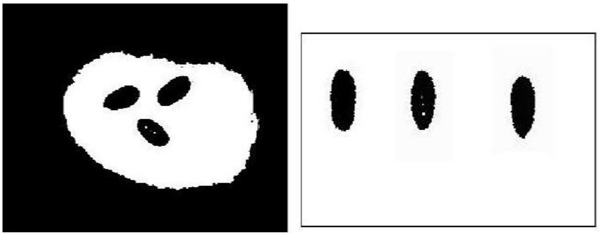


Fig. 6. (a) binary image of the equalized image and (b) the Wheat grain images separated

2.2 Separation of Wheat Grains

Separation of each wheat grain and saving them in some other image is done so that each image has only one wheat grain and the features can be easily extracted. Separation of wheat is done by simple mathematical calculations. The result is calculated by applying two nested loops having initial and final limits as (0,0) and size of the image respectively. In the loop pixel having value 0 is searched and all surrounding pixels should also be 0, and have area between 350 pixels and 1000 pixels. The area of a single wheat grain is estimated to be between these limits. After getting the number and the collection of pixels of each grain they are saved in some other image.

When mentioning the wheat shape, the first thing appears in your mind might be the margin of a wheat. Convolving the image with a Laplacian filter of following 3 × 3 spatial mask:

0 1 0
1 -4 1
0 1 0

The margin of the wheat image.is calculated An example of image pre-processing is illustrated in Fig. 6(b). To make boundary as a black curve on white background, the “0” “1” value of pixels is swapped.

3 Feature Analysis and Extraction

In this paper, 7 commonly used digital morphological features (DMFs), derived from 5 basic features, and are extracted so that a computer can obtain feature values quickly and automatically (only one exception).

3.1 Basic Geometric Features

Firstly, 5 basic geometric features are calculated.

Principle Diameter: The principle diameter is defined as the longest distance between any two points on the margin of the wheat sample. It is denoted as Dp.
Physiological Length: The only human interfered part of our algorithm is that you need to mark the two terminals of the crease via mouse click. The distance between the two terminals is defined as the physiological length. It is denoted as Lp.
Physiological Width: Drawing a line passing through the two terminals of the wheat boundary, one can plot infinite lines orthogonal to that line. The number of intersection pairs between those lines and the wheat margin is also infinite. The longest distance between points of those intersection pairs is defined at the physiological width. It is denoted as Wp. The two lines are orthogonal if their degree is 90◦ ± 0.5◦ since the coordinates of pixels are discrete.
Wheat Area: The value of wheat area is easy to evaluate, just counting the number of pixels of binary value 1 on smoothed wheat image. It is denoted as Aw.
Wheat Perimeter: Denoted as Pw, Wheat perimeter is calculated by counting the number of pixels consisting wheat margin.

3.2 Digital Morphological Features

Based on 5 basic features introduced previously, 12 digital morphological features are defined which are used for Wheat recognition.

Smooth factor: The effect of noises to image area is used to describe the smoothness of wheat image. In this paper, smooth factor is defined as the ratio between area of wheat image smoothed by 5 × 5 rectangular averaging filter and the one smoothed by 2 × 2 rectangular averaging filter.

Aspect ratio: The aspect ratio is defined as the ratio of physiological length Lp to physiological width Wp, thus Lp/Wp.

Form factor: This feature is used to describe the difference between wheat seed and a circle. It is defined as 4πA/P2 where Aw is the wheat area and Pw is the perimeter of the wheat margin.

Rectangularity: Rectangularity describes the similarity between wheat and a rectangle. It is defined as LpWp/Aw, where Lp is the physiological length, Wp is the physiological width and Aw is the wheat area.

Narrow factor: Narrow factor is defined as the ratio of the diameter Dp and physiological length Lp, thus Dp/Lp.

Perimeter ratio of diameter: Ratio of perimeter to diameter, representing the ratio of wheat perimeter Pw and wheat diameter Dp, is calculated by Pw/Dp.

Perimeter ratio of physiological length and physiological width: This feature is defined as the ratio of wheat perimeter Pw and the sum of physiological length Lp and physiological width Wp, thus Pw/(Lp +Wp).

4 Neural Networks Implementation

The Neural Network used in this experiment consisted of 3 layers. The input layer is formed by using 140 neurons, reflecting the number of input vector components. In order to train the neural network, a set of training wheat was required, and the varieties were predefined. During training, the connection weights of the neural network were initialized with some random values.

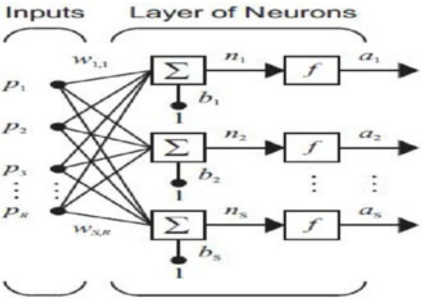
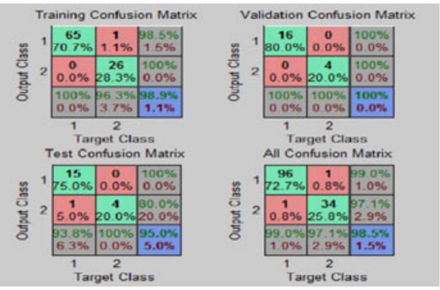


Fig. 7. Basic structure of a Neural Network **Fig. 8.** Samples of wheat taken for testing

Fig. 7 shows the basic structure of a neural network. It consists of no. of inputs denoted by p, R is the number of elements in input vector S number of neurons in layer. P in our case is a vector containing number of wheat samples in our case it is 200 and there are 12 different p i.e. p varies from 1 to 12 depending upon the factors which have been taken into account. The output in our case is 0 or 1 i.e. whether a given sample is wheat or not so there are 2 values of a. In order to train the neural network, a set of training wheat was required, and the varieties were predefined. During training, the connection weights of the neural network were initialized with some random values. The training samples in the training set were input to the neural network classifier in random order and the connection weights were adjusted according to the error back propagation learning rule. A total of 200 sample of wheat were used This process was repeated until the mean squares error (MSE) fell below a predefined tolerance level or the maximum number of iterations is achieved. When the network training was finished, the network was tested with test dataset (60 wheat grains), and the classification accuracies were calculated.

5 Experimental Results

The results have been calculated using 75% wheat samples for training, 15% for validation and 15 % for testing. The result has been plotted using MATLAB GUI in the form of confusion matrix.



Training Confusion Matrix

Output Class	1	2	
1	65 70.7%	1 1.1%	98.5% 1.5%
2	0 0.0%	26 28.3%	100% 0.0%
	100% 0.0%	96.3% 3.7%	98.9% 1.1%
Target Class	1	2	

Validation Confusion Matrix

Output Class	1	2	
1	16 80.0%	0 0.0%	100% 0.0%
2	0 0.0%	4 20.0%	100% 0.0%
	100% 0.0%	100% 0.0%	100% 0.0%
Target Class	1	2	

Test Confusion Matrix

Output Class	1	2	
1	15 75.0%	0 0.0%	100% 0.0%
2	1 5.0%	4 20.0%	80.0% 20.0%
	93.8% 6.3%	100% 0.0%	95.0% 5.0%
Target Class	1	2	

All Confusion Matrix

Output Class	1	2	
1	96 72.7%	1 0.8%	99.0% 1.0%
2	1 0.8%	34 25.8%	97.1% 2.9%
	99.0% 1.0%	97.1% 2.9%	98.5% 1.5%
Target Class	1	2	

Fig. 9. The confusion matrix a) training, b) Validation, c) Testing d) Overall Result

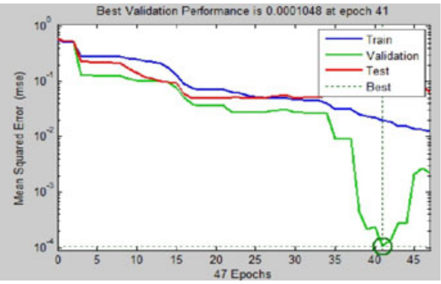


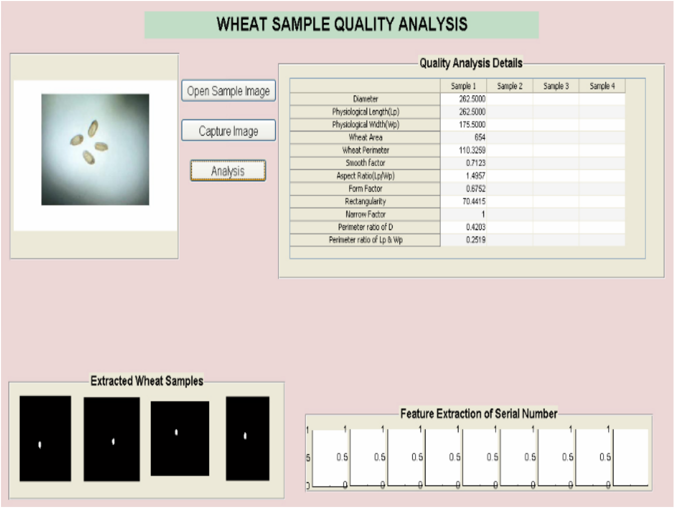
Fig. 10. Graph between the Mean Square Error and Epochs

Fig. 9 shows the confusion matrix. For the training purpose 75% of the data is used and the result in the green boxes shows the accuracy in finding the wheat grain correctly and the boxes in red are for the errors. So the less the error the more is the accuracy. In the training case the error is only 1.1% where as for validation it is 0% and for the testing purpose it is 5% which makes it 0.8% overall. The final accuracy has been shown in the blue matrix. In the first case the accuracy is 98.9%, 100% for the second and 95% for testing which makes it 98.5 % overall. Fig.10 shows the graph between the mean square error and the epochs. The less the error the more is the accuracy. The error is minimum at 42 Epochs and the error is 10^{-4} which is obtained during validation.

Comparison of the values for a fresh wheat and a broken wheat is shown Table 1 which shows that there is quite a large difference between the values.

Table 1. Comparison between broken and unbroken wheat sample

Features	Unbroken Wheat grain	Broken Wheat grain
Diameter	307.5	124.5
Physiological Length(L_p)	307.5	124.5
Physiological Width(W_p)	187.5	109.5
Wheat Area	610	279
Wheat Perimeter	104.669	67.8406
Smooth factor	1.2889	1.44
Aspect Ratio(L_p/W_p)	1.64	1.137
Form Factor	0.6997	0.7618
Rectangularity	94.5184	48.8629
Narrow Factor	1	1
Perimeter ratio of diameter	0.3404	0.5449
Perimeter ratio of physiological length and physiological width	0.2115	0.2899



	Sample 1	Sample 2	Sample 3	Sample 4
Diameter	262.5000			
Physiological Length(Lp)	262.5000			
Physiological Width(Wp)	175.5000			
Wheat Area	654			
Wheat Perimeter	110.3259			
Smooth factor	0.7123			
Aspect Ratio(Lp/Wp)	1.4957			
Form Factor	0.6752			
Rectangularity	70.4415			
Narrow Factor	1			
Perimeter ratio of D	0.4203			
Perimeter ratio of Lp & Wp	0.2519			

Fig. 11. GUI for the wheat sample

Fig. 11 shows snapshot of the MATLAB GUI designed using MATLAB GUIDE tool for advertisement detection and elimination. It consists of three buttons one for taking photos of wheat from the database of an operating system and second button is for taking photos from the camera and the third is for analysis and calculation.

6 Conclusion

The paper presents the technique for grain thickness measurements and grain crease detection for grain quality assessment which is important in meeting market requirements. Grain thickness can be measured using mathematical calculations techniques where a sample of grains can be measured at the same time. By measuring different parameters like area, width, length, etc and forming a matrix of the input data which is given as an input to the neural network for training purposes and the output is obtained. Output data obtained which is greater than 0.5 is termed as wheat grain otherwise as a broken grain. As we can see the case discussed in the paper the accuracy achieved is 98% which is quite high and can be made more accurate if more number of hidden neurons is used.

References

1. Abdellaoui, M., Douik, A., Annabi, M.: Détérmination des critéres de forme et de couleur pour la classificationdes grains de céréales. In: Proc. Nouvelles Tendances Technologiques en Génie Electrique et Informatique, GEI 2006, Hammamet,Tunisia, pp. 393–402 (2006)
2. Barker, D.A., Vouri, T.A., Hegedus, M.R., Myers, D.G.: The use of ray parameters for the discrimination of Australian wheat varieties. Plant Varieties and Seeds 5(1), 35–45 (1992)
3. Barker, D.A., Vouri, T.A., Hegedus, M.R., Myers, D.G.: The use of Chebychev coefficients for the discrimination of Australian wheat varieties. Plant Varieties and Seeds 5(1), 103–111 (1992)
4. Keefe, P.D.: A dedicated wheat grain image analyzer. Plant Varieties and Seeds 5(1), 27–33 (1992)
5. Sapirstein, H.D., Kohler, J.M.: Physical uniformity of graded railcar and vessel shipments of Canada Western Red Spring wheat determined by digital image analysis. Canadian Journal of Plant Science 75(2), 363–369 (1995)
6. Paliwal, J., Shashidhar, N.S., Jayas, D.S.: Grain kernel identification using kernel signature. Transactions of the ASAE 42(6), 1921–1924 (1999)
7. Majumdar, S., Jayas, D.S.: Classification of cereal grains using machine vision. Transactions of the ASAE 43(6), 1669–1675 (2000)
8. Neuman, M., Sapirstein, H.D., Shwedyk, E., Bushuk, W.: Wheat grain colour analysis by digital image processing: I. Methodology. Journal of Cereal Science 10(3), 175–182 (1989)
9. Neuman, M., Sapirstein, H.D., Shwedyk, E., Bushuk, W.: Wheat grain colour analysis by digital image processing: II. Wheat class determination. Journal of Cereal Science 10(3), 182–183 (1989)
10. Luo, X.Y., Jayas, D.S., Symons, S.J.: Identification of damaged kernels in wheat using a colour machine vision system. Journal of Cereal Science 30(1), 49–59 (1999)
11. Majumdar, S., Jayas, D.S.: Classification of cereal grains using machine vision. II. Color models. Transactions of the ASAE 43(6), 1677–1680 (2000)

12. Majumdar, S., Jayas, D.S.: Classification of cereal grains using machine vision. III. Texture models. Transactions of the ASAE 43(6), 1681–1687 (2000)
13. Majumdar, S., Jayas, D.S.: Classification of cereal grains using machine vision. IV. Combined morphology, color, and texture models. Transactions of the ASAE 43(6), 1689–1694 (2000)
14. Paliwal, J., Visen, N.S., Jayas, D.S., White, N.D.G.: Comparison of a neural network and a nonparametric classifier for grain kernel identification. Biosystems Engineering 85(4), 405–413 (2003)
15. Visen, N.S., Jayas, D.S., Paliwal, J., White, N.D.G.: Comparison of two neural network architectures for classification of singulated cereal grains. Canadian Biosystems Engineering 46, 3.7–3.14 (2004)
16. Abdellaoui, M., Douik, A., Annabi, M.: Hybrid method for cereal grain identification using morphological and color features. In: Proc. 13th IEEE International Conference on Electronics, Circuits, and Systems, Nice, France, pp. 870–873 (2006)
17. Choudhary, R., Paliwal, J., Jayas, D.S.: Classification of cereal grains using wavelet, morphological, colour, and textural features of non-touching kernel images. Biosystems Engineering 99, 330–337 (2008)
18. Douik, A., Abdellaoui, M.: Cereal varieties classification using wavelet techniques combined to multi-layer neural networks. In: Proc. 16th Mediterranean Conference on Control and Automation, Ajaccio, France, pp. 1822–1827 (2008)
19. Chtioui, Y., Panigrahi, S., Backer, L.F.: Rough sets theory as a pattern classification tool for quality assessment of edible beans. Trans. of the ASAE 42(4), 1145–1152 (1999)
20. Mallat, S.G.: A theory for multiresolution signal decomposition: the wavelet representation. IEEE Transactions on Pattern Analysis and Machine Intelligence 11(7), 674–693 (1989)
21. Sakai, N., Yonekawa, S., Matsuzaki, A.: Two dimensional image analysis of the shape of rice and its application to separating Varieties. Journal of Food Engineering 27, 397–407 (1996)
22. Ghazanfari, A., Irudayara, J., Kusalik, A.: Grading pistachio nuts using neural network approach. Trans. of the ASAE 39(6), 2319–2324 (1996)

An Optimal Design of FIR Digital Filter Using Genetic Algorithm

Ranjit Singh Chauhan[1] and Sandeep K. Arya[2]

[1] Astt. Prof., Electronics & Communication Engg., JMIT Radaur,
Yamunanagar, Haryana, India
[2] Chairman, Electronics & Communication Engg., GJU Hisar, Haryana, India

Abstract. The Paper presents a simple computer-aided design approach for designing Finite Impulse Response (FIR) digital filters. FIR filter is essentially a digital filter with non Recursive responses. Since the error surface of digital FIR filters is generally nonlinear and multimodal, global optimization techniques are required in order to avoid local minima. There are many ways for the design of FIR Digital filters. This Paper Presents soft computing technique for the design of FIR filters. In this Paper, Genetic Algorithm (GA) base evolutionary method is proposed for design of FIR digital filter. GA is a well-known powerful global optimization algorithm introduced in combinatorial optimizations problems. The Simulation result for the employed example is presented in this paper and can be efficiently used for FIR digital filter design.

Keywords: Digital Filter, Finite-Impulse response (FIR), Genetic Algorithm (GA), Optimization.

1 Introduction

Over the past several decades the field of Digital Signal Processing (DSP) has grown to important both theoretically and technologically. In DSP, there are two important types of Systems. The first type of systems performs signal filtering in time domain and hence it is known as digital filters. The second type of systems provide signal representation frequency domain and are known as Spectrum Analyzer. Digital filtering is one of the most powerful tools of DSP. Digital filters are capable of performance specifications that would, at best, be extremely difficult, if not impossible, to achieve with an analog implementation. In addition, the characteristics of a digital filter can be easily changed under software control. Digital filters are classified either as Finite duration unit pulse response (FIR) filters or Infinite duration unit impulse response (IIR) filters, depending on the form of unit impulse response of the system. In the FIR system, the impulse response sequence is of finite duration, i.e., it has a finite number of non zero terms. However, because the error surface of FIR filters is usually nonlinear and multimodal, conventional gradient-based design methods may easily get stuck in the local minima of error surface [1], [2]. Therefore,

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 51–56, 2011.

some researchers have attempted to develop design methods based on modern heuristic optimization algorithms such as genetic algorithm (GA) [1]–[8], simulated annealing (SA) [3], tabu search (TS) [2] etc.

Analytical or simple iterative methods usually lead to sub-optimal designs. Consequently, there is a need of optimization methods (heuristic type) that can be use to design digital filters that would satisfy prescribed specifications. Goldberg presented a detailed mathematical model of Genetic Algorithm [8]. Benvenuto et al. (1992) described the salient features of using a simulated annealing (SA) algorithm in the context of designing digital filters with linear phase digital filter. The algorithm is then applied to the design of FIR filter. The result was not impressive. Moreover, it is computationally very expensive. Ahmadi et al. (2003) used genetic algorithm to design 1-D IIR filter with canonical-signed-digit coefficients restricted to low-pass filter. The drawback of preceding design methods is that the computation time is quite long. To test the optimization procedure, the proposed algorithm is implemented in Matlab and results are found to be very encouraging.

This Paper is organized as follows: In Section II, FIR digital filter design aspects are discussed. In section III, Genetic Algorithm (GA) approach is briefly mentioned. The Genetic Algorithm (GA) related to filter design is proposed in Section IV. The simulation result of design example used is briefly described in Section V. The Conclusion and future scope is described in Section VI.

2 FIR Filter Design Issues

Digital filters are classified as Recursive and Non-Recursive filters [11]. FIR is recursive type filter. The response of the recursive or FIR filters depends only upon present and previous input values. FIR filters have following advantage:-

- Digital finite impulse response (FIR) require less calculation work as compare to equivalent Infinite impulse response (IIR)
- They can have exact linear phase.
- They are always stable.
- They can be realized efficiently in hardware.
- The filters start up transients has finite duration

Consider the FIR filter with the input-output relationship governed by:

$$y[n] = \sum_{i=0}^{N} a_i \; x[n-i]$$

where $x(k)$ and $y(k)$ are the filter's input and output, respectively, and N is the filter order. The transfer function of this FIR filter can be written in the following general form:

$$H(z) == \sum_{I=0}^{N} a_i z^{-i}$$

An important task for the designer is to find values of a_i such that the magnitude response of the filter approximates a desired characteristic while preserving the stability of the designed filter. The stability is assured if all the poles of the filter lie inside the unit circle in the z-plane. The Digital filters have various stages for their design. The flow chart of the Design of Digital filter is shown in Fig. 1[4].

In the design of FIR filter using window technique the order can be calculated using the formula given by:

$$N = \frac{-20\log(\sqrt{\delta_P \delta_S}) - 13}{14.6(f_{sb} - f_p)/f_s}$$

Kaiser window function is given by:

$$W_k(n) = \begin{cases} \dfrac{I_0(\beta)}{I_0(\alpha)} & for|n| \le \dfrac{N-1}{2} \\ 0 & otherwise \end{cases}$$

where α is an independent variable determined by Kaiser. The parameter β is expressed by:

$$\beta = \alpha\left[1 - \left(\frac{2n}{N-1}\right)^2\right]^{0.5}$$

N is order of filter, I_0 (β) I_0 (α) are Bassel function. Also δ_p: pass band ripple, δ_s stop band ripple, f_{sb} stop band frequency, f_p pass band frequency.

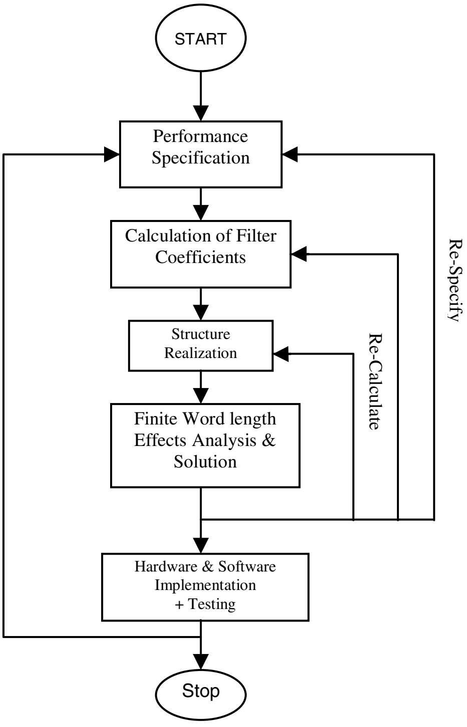

START
Performance Specification
Calculation of Filter Coefficients
Structure Realization
Finite Word length Effects Analysis & Solution
Hardware & Software Implementation + Testing
Stop
Re-Calculate
Re-Specify

Fig. 1. Flow chart of digital filter design

3 Genetic Algorithm

Genetic Algorithms (GA) are stochastic search methods that can be used to search for an optimal solution to the evolution function of an optimization problem. Holland proposed genetic algorithms in the early seventies as computer programs that mimic the natural evolutionary process. De Jong extended the GAs to functional optimization and a detailed mathematical model of a GA was presented by Goldberg in 1970. GAs manipulates a population of individual in each generation. Population of individuals in each *generation* (iteration) where each individual, termed as the *chromosome*, represents one candidate solution to the problem. Within the population, fit individuals survive to reproduce and their genetic materials are recombined to produce new individuals as *offsprings*. The genetic material is modeled by some data structure, most often a finite-length of attributes. As in nature, *selection* provides the necessary driving mechanism for better solutions to survive. Each solution is associated with a *fitness* value that reflects how good it is, compared with other solutions in the population. The recombination process is simulated through a *crossover* mechanism that exchanges portions of data strings between the chromosomes. New genetic material is also introduced through *mutation* that causes random alterations of the strings. The frequency of occurrence of these genetic operations is controlled by certain pre-set probabilities. The selection, crossover, and mutation processes as illustrated in fig. 2 [6] constitute the basic GA cycle or generation, which is repeated until some pre-determined criteria are satisfied. Through this process, successively better and better individuals of the species are generated.

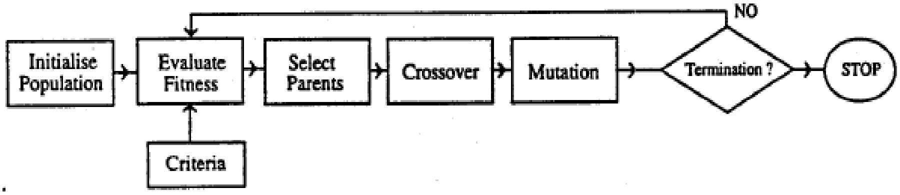


Fig. 2. Basic Genetic Algorithm cycle

4 Genetic Algorithm and Filter Design

The error surface of digital finite-impulse response (FIR) filters is generally nonlinear and multimodal, global optimization techniques are required in order to avoid local minima. In designing FIR digital filter, the values of a_i must be such that the magnitude response of the filter approximates a desired characteristic while preserving the stability of the designed filter. Although Simulated Annealing (SA) is easy to be implemented and good at local convergence, depending on the initial solution, it might often require too many cost function evaluations to converge to the global minima. Relatively little work has been published so far on GAs applied to analogues filters [5]. A number of practical issues are important in analogues filter design. One is the choice of component values. A conventional form of GA is used to perform the design of FIR digital filter. The selected random number in range [0.17 0.76] has been found satisfactory. Fitness is evaluated in the normalized frequency range [0, 1] over a uniform grid of frequency point. Selection is the process of choosing structures for the next generation from the structures in the current generation. In the design of filter the Selection function used is the "*Stochastic Universal Sampling*", which allocate to each individual a portion of the wheel proportional to the individual's fitness. Crossover is the process of generating a child from two parents by taking a part from one of the parents and replaces it with the corresponding part from the second parent and vice versa. IIR Filter designing is performed by *double point crossover* between pairs of individuals and returns the current generation after mating. Mutation is a change done on some of the children resulted from the crossover process by flipping the value of one of the bits randomly. The benefit of such operation is to restore the lost genetic values when the population converges too fast. The filter coefficients were encoded in terms of 16 bit binary string with initial Crossover and Mutation probability of 0.8 and 0.02 respectively, and the population size of 50 was assumed. The GA produced one solution that satisfied both magnitude and phase templates.

5 Simulation Results

This section represents the simulation frame work for the design of FIR filter using genetic algorithm. Simulation is carried out for certain specification such as pass band ripple (r_P) = 0.1dB, stop band ripple (r_s) = 40dB, pass band frequency (f_p) = 150 Hz, stop band frequency (f_{sb}) = 250Hz and sampling frequency (f_s) = 1000Hz. The plots of magnitude and phase response of traditional and proposed schemes are shown in Fig. 3 and Fig. 4. A comparison of traditional and proposed FIR filter coefficients is shown in Table 1.

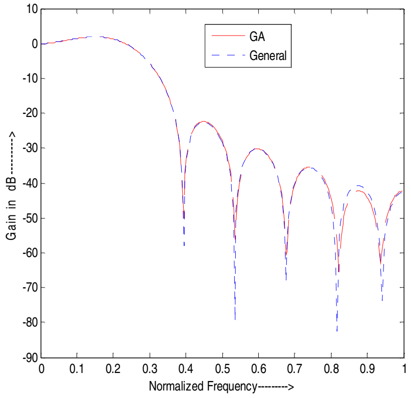


Fig. 3. Magnitude Plot

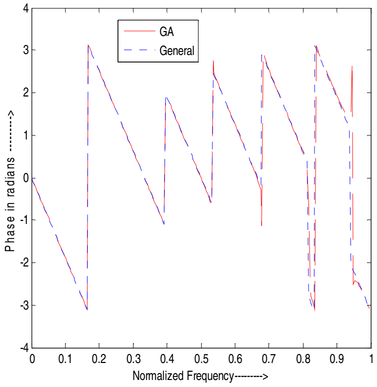


Fig. 4. Phase Plot

Table 1. Comparison of Low-Pass Filter Coefficient

Name of Method	Order of filter	Coefficient of FIR Filter
Kaiser window	13	-0.0346, -0.0707, -0.0520, 0.0364, 0.1682, 0.2861, 0.3330, 0.2861, 0.1682, 0.0364, -0.0520, -0.0707, -0.0346
Proposed method	13	-0.0345, -0.0708, -0.0524, 0.0365, 0.01681, 0.2860, 0.3320, 0.2859, 0.1681, 0.0364, 0.0524, -0.0714, -0.0343

6 Conclusion

This paper presents the design of digital FIR filter based on General Algorithm (GA) and the benefits of GA for designing digital filters have been studied. The examples demonstrate the versatility of the proposed approach. Thus it is believed that the proposed algorithm is capable of quick and high performance. The proposed method can be extended to arbitrary magnitude response specifications and multiband. Further, the other Evolutionary algorithm can be discussed for the design of FIR filters.

References

1. Vaccaro, R.J., Harrison, B.F.: Optimal Matrix-Filter Design. IEEE Transactions on signal processing 14(3), 705–710 (1996)
2. Zhang, X., Iwakura, H.: Design of IIR Digital Filters based on Eigen value Problem. IEEE Transactions on Signal processing 44(6), 1319–1325 (1996)
3. Kacelenga, R.V., Graumann, P.V., Turner, L.E.: Design of filters using simulated annealing. In: IEEE Proc. Int. Symp. on Circuits and Systems, New Orleans, LA, pp. 642–645 (1990)
4. Chauhan, R.S., Kamboj, A.: MATLAB based Design of Digital FIR filter using window technique. In: IEEE Conf. on Art. Int. Systems (AIS), pp. 208–210 (2007)
5. Joelle, S., Stephen, P.B.: Filter Design with Low Complexity Coefficients. IEEE Transactions on Signal processing 56(7), 3162–3170 (2008)
6. Goldberg, D.E.: Genetic Algorithm in search, optimization and machine Learning, Low price Edition. Pearson Education, Delhi (2005)
7. Fabrizio, A., Re, E.D.: Design of IIR Eigen filters in the Frequency domain. IEEE Transactions on Signal processing 46(6), 1694–1700 (1998)
8. William, T., Miller, W.C.: Genetic algorithms for the design of Digital filters. In: Proceedings of IEEE ICSP 2004, vol. 4, pp. 9–12 (2004)
9. Proakis, J.G., Manolakis, D.G.: Digital Signal Processing: Principles, Algorithms, and Applications, 4th edn. Pearson Education, Inc., New Delhi (2007)
10. Cousseau, J.E., Stefan, W., Donate, P.D.: Factorized All-Pass Based IIR Adaptive Notch Filters. IEEE Transaction on Signal Processing 55(11), 5225–5236 (2007)
11. Chauhan, R.S., Arya, S.K.: Design of IIR digital filter using analog to digital mapping. Journal of Neural Computing Systems 3(1), 51–55 (2010)

Some Observations on Algorithms for Computing Minimum Independent Dominating Set

Anupama Potluri and Atul Negi

Department of Computer and Information Sciences, University of Hyderabad,
Gachibowli, Hyderabad - 500046, India
apcs@uohyd.ernet.in, atulcs@uohyd.ernet.in

Abstract. In this paper, we present some observations on the various algorithms proposed to find a Minimum Independent Dominating Set (MIDS). MIDS is proven to be an NP-hard problem. We compared an exact algorithm based on intelligent subset enumeration with another exact algorithm based on matching in graphs. We found that the former performs better than the latter for small graphs despite having a worse asymptotic complexity. There is only one Polynomial Time Approximation Scheme (PTAS) proposed in literature for computing MIDS which works for polynomially bounded growth graphs. We observed that changing the ϵ value in the PTAS reduces the running time quite drastically but does not increase the cardinality returned significantly. We compared the cardinality of the IDS returned by various heuristics for grid, unit disk graph and general graph topologies. The results show that the highest degree heuristic returns the best cardinality amongst all these algorithms in literature for all graphs except grid graphs for which the inter-dominator 3-hop distance heuristic performs better. To the best of our knowledge, this is the first empirical study where the exact, PTAS and heuristic solutions to the MIDS problem have been compared in terms of the quality of the solution returned as well as provide insights into the behavior of these approaches for various types of graphs.

Keywords: Minimum Independent Dominating Set, Heuristics, Clustering, Wireless Networks.

1 Introduction

A study of the survey of clustering algorithms for wireless networks [1], [2] shows that the initial clustering schemes for wireless networks were primarily independent dominating set (IDS) schemes such as in [3], [4], [5] etc. There is a renewed interest in IDS based schemes in the recent times, in the context of wireless sensor networks (WSNs) [6] and wireless sensor and actor networks (WSANs) [7]. These are useful in determining where to place actors or design topologies that lead to energy-efficiency.

A clustering scheme that leads to a minimum independent dominating set (MIDS) would minimize the number of clusters such that the cluster heads do not interfere with each other. While a lot of IDS-based schemes have been proposed, no attempt has been made by anyone so far to determine the quality of the solution returned by the heuristics. It is proven that computing a MIDS is an NP-hard problem [8], including on Unit Disk

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 57–68, 2011.

Graphs (UDGs) [9]. It is also proven that it is very hard to approximate it for general graphs [10]. Exact algorithms that improve the running time for MIDS are still being proposed in literature [11], [12] and [13]. So far, only one PTAS has been proposed for computing MIDS [14] and that too for polynomially bounded growth graphs (see section 2) such as unit disk graphs, quasi-disk graphs and coverage-area graphs. To the best of our knowledge, we are the first to implement the exact and PTAS algorithms for MIDS and compare the quality of solutions of heuristics. This has led us to gain valuable insights into these algorithms and their validity for real-life situations.

We studied the intelligent enumeration algorithm for computing the exact MIDS and the algorithm using graph matching in [11]. While the asymptotic complexity of enumeration algorithm may be worse than that of [11], in practice, it works much better for small number of nodes. So, we use the intelligent enumeration exact algorithm for computing local exact MIDS as part of PTAS implementation.

The rest of the paper is organized as follows: we establish the notation used in the rest of the paper in section 2. We review some of the related work in clustering using IDS, the exact and PTAS algorithms for MIDS in section 3. We present results comparing the running times of the enumeration algorithm and the algorithm by Liu and Song [11] in section 4. Then, we present the new heuristic Inter-Dominator 3-hop distance heuristic (3HD), the results on the cardinality of MIDS returned by the various heuristics and PTAS in section 5. We record the most important observations based on the results obtained in section 6. We conclude with section 7.

2 Preliminaries

A graph is represented as $G = (V, E)$ where V is the set of nodes in the graph and E is the set of edges. We only assume undirected graphs in this paper. A subset $S \subseteq V$ is called an independent set if the nodes in the set are **not** pairwise adjacent. A subset $D \subseteq V$ is called dominating, if every node $v \in V$ is either a member of D or there exists an edge $(u, v) \in E$ where $u \in D$. Every maximal independent set is a dominating set and therefore is called an Independent Dominating Set (IDS). A maximal independent set of the smallest cardinality is called as Minimum Independent Dominating Set (MIDS). We represent MIDS by $\overline{D}$ throughout this paper.

The closed neighborhood of a node $v \in V$ is defined to be $\Gamma(v) = \{u \in V \mid (u, v) \in E\} \cup \{v\}$. In other words, it is the node and all its 1-hop neighbors that form the closed neighborhood. The open neighborhood of v is defined to be only its 1-hop neighbors and does not include the node itself. We use the notation $N(v)$ to represent this.

Unit Disk Graphs are graphs which have been used to model ad hoc wireless communication networks. The wireless nodes are considered to be the center of a disk of unit radius. An edge exists between two nodes if their disks intersect with each other. The UDGs also have the property that the number of independent nodes in the r-hop neighborhood of a node are bounded by a polynomial $f(r) = O(r^c)$ for some constant $c \geq 1$.

3 Related Work

In this section, we, first, review some of the IDS-based heuristic clustering schemes. We then present the exact algorithm by Liu and Song [11] that we implemented. We show that the running time of this algorithm is worse than an intelligent enumeration algorithm for small graph sizes. We finish with the description of the PTAS algorithm for MIDS against which we compare the cardinality of IDS returned by various heuristics.

3.1 IDS Based Clustering Schemes

The lowest ID clustering algorithm described in [3] works as follows: every node in the graph is aware of all its 1-hop neighbors. A node which has the lowest ID amongst all its 1-hop neighbors declares itself as a cluster head. All its neighbors then join this cluster. If a node finds that its lowest ID neighbor has actually joined another cluster as a member and it has the lowest ID amongst the rest of the neighbors, it declares itself as a cluster head.

In k-hop clustering [4], as the name indicates, the nodes of a cluster are at most k hops away from their cluster head rather than one hop away. In this paper, the lowest ID algorithm is generalized to k hops. They also propose an algorithm based on the connectivity of a node. The node with higher connectivity or, in other words, degree, in its k-hop neighborhood becomes the cluster head. If two neighbors have the same degree, the lowest ID node becomes the cluster head.

In [7], the authors propose positioning of mobile actors of a wireless sensor and actor network (WSAN) such that the number of sensors within the range of an actor is maximized. The paper proposes a heuristic that returns a k-hop IDS where the dominating nodes represent the positions where the actors must be placed to achieve maximum coverage. A node is called a border node if it is k hops away from a dominator node. A node is chosen as a dominator in the following way: every node computes a suitability value, S_i based on its degree and its distance to the nearest border node of another dominator. This suitability value is then compared to a random value, R. If $R < S_i$, then, the node becomes a dominator and advertises this fact to its k-hop neighbors.

3.2 Exact Algorithm for Computing MIDS

In the exact algorithm for finding MIDS that we implemented [11], a maximal matching, M, has to be found in the given graph. $V_m \subseteq V$ is the set of nodes that are covered by M. The set $I = V \setminus V_m$ is an independent set. The minimum independent dominating set, $\overline{D}$ satisfies the relationship $\overline{D} \cap I = I \setminus N(\overline{D} \cap V_m)$. We can obtain a candidate for an independent dominating set by using the formula $D_s = S \cup (I \setminus N(S))$, where $S \subseteq V_m$ is a subset enumeration of nodes in V_m. We enumerate all possible subsets of V_m. The D_s which is an IDS with the minimum cardinality is the MIDS. In the worst case, M is a perfect matching and $\mid M \mid = \frac{V}{2}$. Hence, in the worst case, the running time of this algorithm is $3^{\frac{|V|}{2}}$. We present this algorithm to calculate MIDS in Algorithm 1.

Algorithm 1. ***Exact-MIDS(**$G = (V, E), \overline{D}$**)***

```
$\overline{D} := \phi$
$| \overline{D} | := | V |$
Find a maximal matching $M$ of $G$
$V_m \subseteq V$ is the set of nodes covered by $M$
$I := V \setminus V_m$
for all enumerated $S \subseteq V_m$ do
    $D_s := S \cup (I \setminus N(S))$ where $N(S)$ is the open neighborhood of $S$
    if $(Independent(D_s) \bigwedge Dominating(D_s) \bigwedge$
    $(| \overline{D} | > | D_s |))$ then
        $| \overline{D} | := | D_s |$
        $\overline{D} := D_s$
    end if
end for
```

3.3 PTAS for MIDS

There is, so far, only one PTAS proposed for finding MIDS [14]. The primary idea behind this scheme is to divide the given graph into sub-graphs which are separated by at least 2 hops and find locally optimal solutions that satisfy a bound on the cardinality. Then, the union of the locally optimal solutions is computed. If this is found to violate the independence property, the independence property is repaired. This yields the global MIDS such that it is at most $(1 + \epsilon)$ times the optimal solution. The algorithm achieves this by finding a local MIDS which satisfies the following property :$| \ D_{i+3} \ | \ \leq (1 + \epsilon) \cdot \ | \ D_i \ |$, where i is initialized to 0. D_i represents the exact MIDS of the closed i-hop neighborhood, $\Gamma_i(v)$, of a node v and D_{i+3} represents the exact MIDS of the closed $(i + 3)$-hop neighborhood, $\Gamma_{i+3}(v)$, of the given node, v. The nodes that dominate the given set of nodes, say, $\Gamma_i(v)$ can be from the closed neighborhood of the set, i.e., $\Gamma(\Gamma_i(v))$. As long as the bound, given above, on the cardinality of the dominating sets, is not satisfied, the neighborhood is expanded by incrementing i. Thus, at every stage, the local solution is never more than $(1 + \epsilon)$ times the local optimum. For more details of the algorithms and proofs, the reader is referred to [14].

4 Comparison of Exact Algorithms for Computing MIDS

Since the PTAS algorithm requires computing the exact MIDS of the local neighborhood of nodes, we explored the literature for exact exponential algorithms that compute MIDS. We implemented the scheme in [11] as well as an intelligent subset enumeration algorithm. We found that the latter works well in practice for small graphs.

The underlying principle of the intelligent enumeration algorithm is efficient subset enumeration as given in [15] where all possible subsets are enumerated in increasing order of cardinality. It generates a bitmap $B = \{b_0 b_1 ... b_{N-1}\}$ where if bit b_i is set, it implies that the i^{th} element of the set is present in the subset and not otherwise. We use this bitmap to construct a subset of nodes, $\overline{D} \in V$, where $\overline{D}$ is a candidate for an IDS. If the subset so constructed is, indeed, an IDS, the algorithm terminates since the first such set found will be the IDS with minimum cardinality. In fact, this algorithm can be made more efficient by starting enumeration from the lower bound on MIDS, if a specific type of graph has a proven lower bound. The algorithm is given in Algorithm 2.

Algorithm 2. ***Intelligent-Enumeration-MIDS(*** $G = (V, E), \overline{D}$***)***

```
for $i = 1$ to $\mid V \mid$ do
  $\overline{D} = \phi$
  for all $bitmap$ with $i$ bits set do
    Construct $\overline{D} \subseteq V$ such that node $u_j \in \overline{D}$ if $j^{th}$ bit is set
    if $(Independent(\overline{D}) \bigwedge Dominating(\overline{D}))$ then
      RETURN $\overline{D}$
    end if
  end for
end for
```

4.1 Experimentation and Analysis

The intelligent enumeration algorithm has the advantage of simplicity of implementation. To examine how it performs as compared to [11], we ran the two algorithms on grid topologies of sizes ranging from 3×3 to 7×7 (we could not get the results of [11] for grid of size 7×7 as it was taking too long). We found that it works well in practice for these graphs as shown in the running time of the two algorithms in Table 1. We also show the number of enumerations that need to be tested for the two algorithms in the table. The value $\binom{n}{r}$ given for intelligent enumeration includes only the highest four $\binom{n}{k}$ terms as these overwhelm the other terms. For [11], we show the value $3^{\frac{|V|}{2}}$ because we found that for all grid topologies, the maximal matching includes all nodes for a graph with even number of nodes and one less for graphs with odd number of nodes. Thus, for grid topologies, this algorithm always has the worst-case running time. We found the time taken to run, even for sizes as small as 49 nodes, is quite prohibitive for exact algorithms. So, we did not try for higher grid sizes. While the number of combinations tested seems almost equal for the two algorithms, intelligent enumeration exits as soon

Table 1. Time taken and Worst-case Enumerations for the Exact Algorithm by Liu and Song and Intelligent Enumeration

Grid Size	Cardinality	Liu and Song		Intelligent Enumeration	
		Time (sec)	$3^{\frac{\lvert V\rvert}{2}}$	**Time (sec)**	$\binom{N}{r}$**,** $r = \mid \overline{D} \mid$
3×3	3	0	140	0	125
4×4	4	0	6560	0	2380
5×5	7	11	920500	1	710930
6×6	10	39225	$3.8742e + 08$	254	$3.87e + 08$
7×7	12	Unknown	$4.892e + 11$	141242	$1.3167e + 11$

as a solution is found. Hence, it does not go through all the enumerations listed. For [11], however, all enumerations have to be tested before we can declare the MIDS. The other reason for the superior performance of intelligent enumeration is as follows: as can be seen from the Algorithms in 1 and 2, while both of them check candidate sets for domination and independence, Algorithm 1 of [11] has to do a lot more computation. If V_m is a much smaller set of nodes than V, then, the number of enumerations is drastically reduced and the algorithm will run fast. We also found that intelligent enumeration worked better than [11] for a UDG of 50 nodes too.

Based on these results, we went ahead with the implementation of the PTAS algorithm by using intelligent enumeration for finding the exact local MIDS. Since we expect that the number of nodes on which the local MIDS needs to be calculated will be small, it makes more sense to use this than the one in [11].

5 Comparison of PTAS and Heuristics for Computing MIDS

The most popular clustering schemes in literature that produce an independent dominating set are the lowest ID clustering and the highest degree node clustering. The algorithm proposed in [7] is similar to the heuristic explained in the next section. We implemented all three heuristics to compare the performance with respect to the cardinality returned by them. The results of the centralized algorithms in our implementation are the best case results for the distributed algorithms. If the distributed algorithms do not implement a time out, then, they will arrive at the same solution as the centralized algorithm after $O(n)$ communication rounds, in the worst case.

5.1 Inter-dominator 3-hop Distance Heuristic

In this algorithm, we always pick dominators that are 3-hops apart. This allows the neighbors of both nodes to be disjoint and hence result in maximum coverage with minimum number of dominators. We call this 3HD heuristic in the rest of the paper. The algorithm can be described as follows: we pick any node $v \in V$ of the graph $G = (V, E)$ and add it to the dominating set $\overline{D}$ and remove the closed neighborhood of v, $\Gamma(v)$ from the graph. We, then, find u, a 3-hop neighbor of v with the highest degree and add it to the dominating set. We repeat this algorithm until either V is empty or we cannot find the next candidate node. If no candidate node is found because some node has no neighbors, we arbitrarily pick any node from the remaining nodes of the graph and repeat the algorithm stated above. The pseudo-code of this scheme is given in Algorithm 3.

5.2 Experimental Design

All the programs were written in C and executed on a server having Intel(R) Xeon(TM) CPU 3.00GHz dual processor quad core with 8GB RAM running Linux version 2.6.9-22.ELsmp. We downloaded the unit disk graph topologies of 50 and 100 nodes included

Algorithm 3. ***Greedy Heuristic for Computing MIDS($G = (V, E), \overline{D}$)***

```
$\overline{D} := \phi$
while $V \neq \phi$ do
    Pick $v \in V$
    while $\exists u$ such that $((v, x) \in E$ and $(x, y) \in E$ and $(y, u) \in E)$ and $u \neq v$ and $(v, u) \notin E$ do
        find $u$ with the highest degree
        $\overline{D} := \overline{D} \cup \{v\}$
        $V := V \setminus \Gamma(v)$
        $v := u$
    end while
    $\overline{D} := \overline{D} \cup \{v\}$
    $V := V \setminus \Gamma(v)$
end while
```

in [16]. There are 300 topologies for each graph size. We calculated the average of the results returned by PTAS and the heuristics on all the instances for both 50 and 100 node graphs. We also used the topology generator included in this software to generate 300 instances of graphs with 200-1000 nodes. We used these for comparing results from the heuristics.

In addition, we also use the BRITE topology generator [17] to generate general graphs, specifically, the Waxman Router model topologies. We also extended BRITE to generate grid topologies. Since the PTAS is applicable only to UDGs, we compared only the heuristics for general graphs. We generated graphs with 50, 100, 250, 500, 800 and 1000 nodes, each having 20 instances. The results were averaged over the 20 instances for each heuristic for all graph sizes.

Grid Topologies. We have run the lowest ID, highest degree and 3HD heuristics along with PTAS on complete grid topologies. In these topologies the 3HD heuristic performs best as observed in figure 1. This is because when the highest degree node is chosen, it leaves many other nodes isolated, all of which have to be added to the IDS increasing its cardinality. The lowest ID does not work well at all because it neither separates the dominators nor does it cover as many neighbors as possible when selecting dominators. A 3-hop separation starting with a non-corner node works best for grid topologies and hence 3HD returns the best cardinality.

Unit Disk Graph Topologies. We ran the PTAS along with heuristics on UDG instances of 50 and 100 nodes each. We found that for small graphs, the heuristics give similar results to the PTAS in much smaller times. For larger graphs, we were unable to run the PTAS because it uses exact algorithm until the condition $| D_{i+3} | \leq | D_i |$ is met to stop expanding the neighborhood. We found that the 5-hop or the 6-hop neighborhood of a node has nearly 60 nodes which becomes too large for the exact algorithm to finish in practical time. Hence, for larger graphs of 200-1000 nodes and general graphs, we only compared the heuristics and present those results here.

5.3 Discussion of Results

Grid Topologies. It is clear from the results for Grids that there are some topologies where the maximum degree heuristic will result in many isolated nodes that can increase the cardinality of the IDS. In such cases, the 3HD heuristic seems a better choice to determine MIDS.

Unit Disk Graph Topologies. We ran the PTAS and heuristics for the unit disk graph topologies with 50 and 100 nodes. The results of this experiment are shown in Figures 2 and 3. We used $\epsilon = 6$ for 100 node graphs as using even $\epsilon = 4$ was taking a lot of time for the PTAS to terminate. As seen later in this section, we observed that the quality of the solution is not affected by increasing ϵ, whereas the time taken is reduced quite drastically.

We can observe from figure 2 that the highest degree node heuristic performs best amongst all the algorithms studied including the PTAS. The 3HD heuristic is better than the lowest ID which is the worst heuristic for minimizing cardinality. The 3HD heuristic gives results comparable to PTAS and maximum degree for 50 nodes.

In figure 4, we can once again see that the highest degree heuristic is consistently better than 3HD and lowest ID for all graph sizes. As stated earlier, the results are averaged over 300 instances for each graph size. So, we can conclude that using highest degree heuristic is best for minimizing the cardinality of the MIDS. Lowest ID may be an easy heuristic for a distributed algorithm but does not minimize the cardinality at all and hence is not of use.

Waxman Router Model Topologies. In figure 5, we present the cardinality of MIDS returned by the three heuristics. We cannot use PTAS for these graphs as these are not polynomially bounded growth graphs. Once again, we see that the highest degree

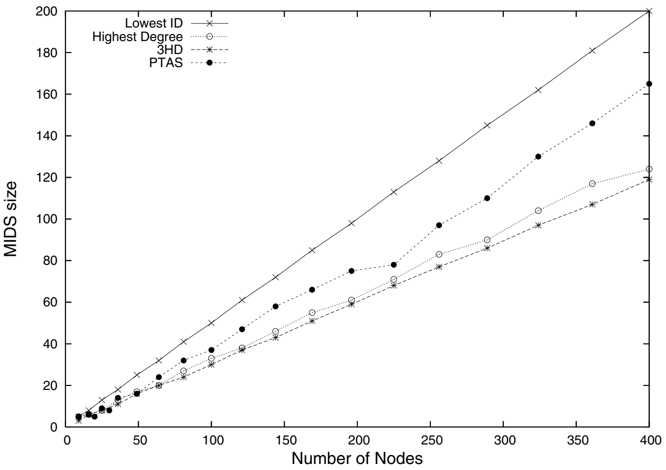


Fig. 1. Average size of the MIDS returned by PTAS with $\epsilon = 2$, Lowest ID, Highest Degree and 3HD on Grid topologies for each graph size

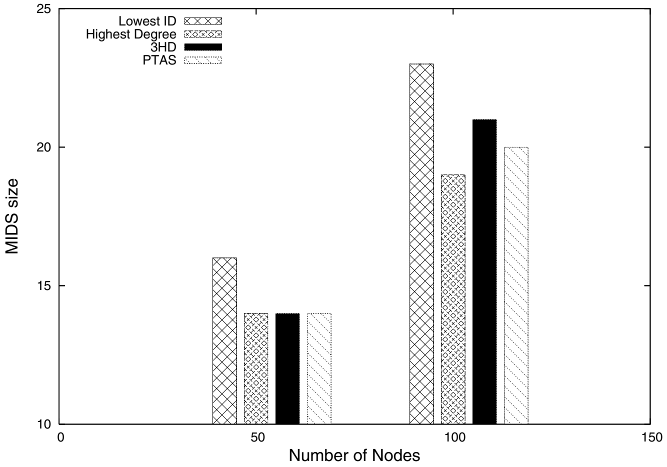


Fig. 2. Average size of the MIDS returned by PTAS with $\epsilon = 2$ for a graph of 50 nodes and $\epsilon = 6$ for a graph of 100 nodes, Lowest ID, Highest Degree and 3HD on 300 UDG topologies for each graph size

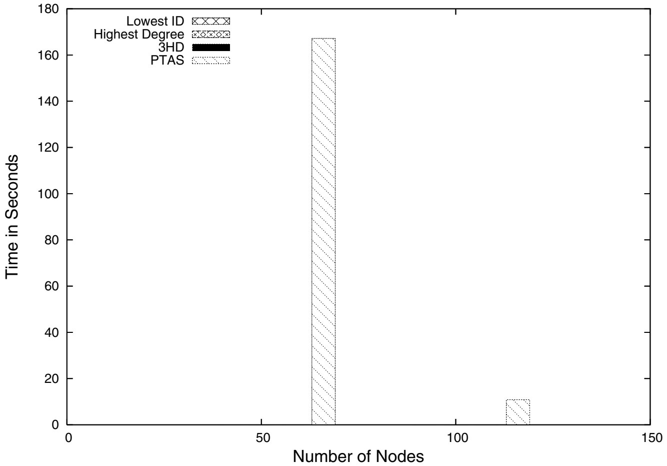


Fig. 3. Average time taken to compute MIDS returned by PTAS with $\epsilon = 2$ for a graph of 50 nodes and $\epsilon = 6$ for a graph of 100 nodes, Lowest ID, Highest Degree and 3HD on 300 topologies for each graph size. **Note:** Heuristics takes an average time in the order of milli/microseconds.

heuristic returns the best cardinality, followed by 3HD heuristic and the worst is lowest ID. In fact, we can observe that the difference in cardinality between these three heuristics is quite large for general graphs such as the Waxman Router model topologies. This is easily explained by the fact that a node with the highest degree covers the maximum number of uncovered nodes. Thus, we need fewer nodes to cover all the nodes in the graph. In 3HD heuristic, we start at some arbitrary point and proceed to choose

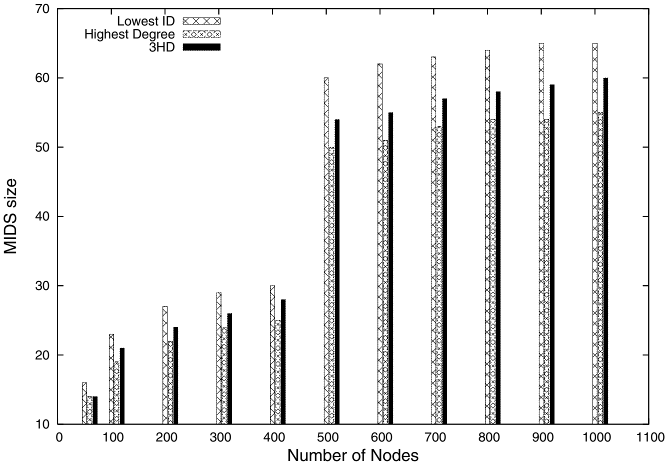


Fig. 4. Average size of the MIDS returned by Lowest ID, Highest Degree and 3HD on 300 UDG topologies for each graph size

nodes which are 3 hops away from the previously chosen node. This ensures that we do not choose 2-hop neighbors to ensure maximum independence but does not take into account the degree of the remaining nodes in the graph. Lowest ID does not result in either a hop separation between the dominators nor is it guaranteed to cover the maximum number of nodes amongst the remaining nodes, both of which try to minimize the cardinality. Hence, it performs really badly.

6 Some Important Observations and Recommendations

Exact Algorithms. The asymptotic complexity of the intelligent enumeration algorithm is, in the worst case, $O(2^n)$ compared to that of [11] which is $O(\sqrt{3}^n)$. However, in practice, for small graphs, the former works much better than the latter. For large graphs, neither is practical. Thus, it is better to use intelligent enumeration over other exact algorithms in practice when an exact algorithm for small graphs is needed.

PTAS. We found that in many instances there was no change at all in the cardinality of the IDS returned by the PTAS when the value of ϵ is changed from 2 all the way to 20. In fact, surprisingly, we found that in some instances, the cardinality returned for $\epsilon = 6$ is less than that for $\epsilon = 4$. On the other hand, the time taken is drastically reduced when ϵ is increased. This is because the stopping criterion for dividing the graph into 2-separated sub-graphs is more easily reached with a higher value of ϵ. On the other hand, as the number of nodes in a neighborhood increases, the exact solution for the neighborhood takes a long time to complete and so the PTAS also takes a prohibitively long time. In fact, with high density graphs, the PTAS is no longer practical. We recommend the use of PTAS in low degree graphs with higher values of ϵ.

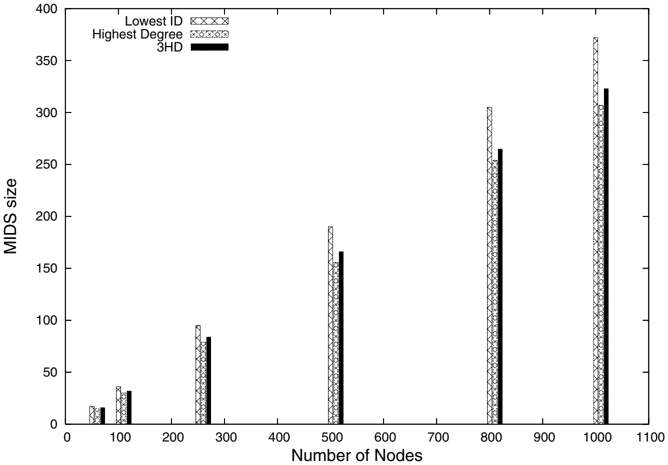


Fig. 5. Average size of the MIDS returned by Lowest ID, Highest Degree and 3HD on 20 BRITE topologies for each graph size

Heuristics. As we can see from the results on various topologies, except for grid topologies, the maximum degree heuristic performs best. However, we can also see that in topologies such as grids, it can lead to many isolated nodes which need to be added to the IDS increasing the cardinality. When we look at the distributed solutions based on these heuristics, it is easy to see that lowest ID and highest degree are more amenable to distribution compared to 3HD. In fact, the algorithm presented in [7], which is similar to 3HD, will under certain circumstances, result in a dominating set that is not k-IDS. At the same time, it is likely to result in larger number of nodes in the MIDS as compared to the maximum degree distributed solution. Thus, we recommend that the maximum degree heuristic be used except for special topologies like the grid.

7 Conclusions

In this paper, we present the insights gained through empirical study about the performance of the various algorithms for computing MIDS on different types of graphs. We found that an intelligent enumeration algorithm takes less execution time than the algorithm presented in [11] for small N despite a higher asymptotic complexity. We discovered that changing the value of ϵ in PTAS does not radically alter the cardinality of MIDS returned, whereas the running time can be reduced to be of more use in practice. We compare the performance of various heuristics for many types of graphs such as grid, UDG and Waxman Router Model. We find that the maximum degree heuristic performs best for all types except grids. For grid graphs, the 3HD heuristic performs better than all other algorithms. In general, we recommend using maximum degree for determining MIDS as a distributed algorithm for it is easy to implement and it gives best results in terms of cardinality returned.

References

1. Erciyes, K., Dagdeviren, O., Cokuslu, D., Ozsoyeller, D.: Graph Theoretic Clustering Algorithms in Mobile Ad Hoc Networks and Wireless Sensor Networks. Appl. Comput. Math. (2), 162–180 (2007)
2. Chen, Y.P., Liestman, A.L., Liu, J.: Clustering algorithms for ad hoc wireless networks. Ad Hoc and Sensor Networks, 145–164 (2006)
3. Lin, C.R., Gerla, M.: Adaptive clustering for mobile wireless networks. IEEE Journal on Selected Areas in Communications 15(7), 1265–1275 (1997)
4. Nocetti, F.G., Gonzalez, J.S., Stojmenovic, I.: Connectivity-based k-hop clustering in wireless networks. Telecommunication Systems 22(1-4), 205–220 (2003)
5. Basagni, S.: Distributed clustering for ad hoc networks. In: Proceedings ISPAN, International Symposium on Parallel Architectures, Algorithms and Networks, pp. 310–315 (1999)
6. Santos, A.C., Bendali, F., Mailfert, J., Duhamel, C., Hou, K.M.: Heuristic for designing energy-efficient wireless sensor network topologies. Journal of Networks 4(6), 436–444 (2009)
7. McLaughlan, B., Akkaya, K.: Coverage-based clustering of wireless sensor and actor networks. In: Proceedings International Conference on Pervasive Services, pp. 45–54 (2007)
8. Garey, M.R., Johnson, D.S.: Computers and Tractability, A guide to the theory of NP-Completeness. Freeman and Company, New York (1979)
9. Clark, B.N., Colbourn, C.J.: Unit disk graphs. Discrete Mathematics 86(1-3), 165–177 (1990)
10. Halldrsson, M.M.: Approximating the minimum maximal independence number. Information Processing Letters 46(4), 169–172 (1993)
11. Liu, C., Song, Y.: Exact algorithms for finding the minimum independent dominating set in graphs. In: Asano, T. (ed.) ISAAC 2006. LNCS, vol. 4288, pp. 439–448. Springer, Heidelberg (2006)
12. Gaspers, S., Liedloff, M.: A branch-and-reduce algorithm for finding a minimum independent dominating set in graphs. In: Fomin, F.V. (ed.) WG 2006. LNCS, vol. 4271, pp. 78–89. Springer, Heidelberg (2006)
13. Bourgeois, N., Escoffier, B., Paschos, V.T.: Fast algorithms for min independent dominating set. CoRR abs/0905.1993 (2009)
14. Hurink, J.L., Nieberg, T.: Approximating minimum independent dominating sets in wireless networks. Information Processing Letters, 155–160 (2008)
15. Loughry, J., van Hemert, J., Schoofs, L.: Efficiently enumerating the subsets of a set (2000), `http://applied-math.org/subset.pdf`
16. Mastrogiovanni, M.: The clustering simulation framework: A simple manual (2007), `http://www.michele-mastrogiovanni.net/software/download/README.pdf`
17. Medina, A., Lakhina, A., Matta, I., Byers, J.: Brite: An approach to universal topology generation. In: Proceedings of the International Workshop on Modeling, Analysis and Simulation of Computer and Telecommunications Systems, MASCOTS 2001 (2001)

SDDP: Scalable Distributed Diagnosis Protocol for Wireless Sensor Networks

Arunanshu Mahapatro and Pabitra Mohan Khilar

Department of CSE, National Institute of Technology, Rourkela, India
arun227@gmail.com, pmkhilar@nitrkl.ac.in

Abstract. This paper proposes a distributed solution for fault diagnosis in wireless sensor networks (WSN). Fault diagnosis is achieved by comparing the heartbeat sequence number generated by neighbouring nodes and dissemination of decision made at each node. The proposed protocol considers channel error where the channel is modelled as two state Markov model. Theoretical analysis and simulation results show that both message and time complexity of proposed protocol is $O(n)$ for an n-node WSN. This work investigates the energy consumed in diagnosing a fault event. The per-node energy overhead is substantially reduced and becomes scalable.

Keywords: Distributed diagnosis, Comparison based model, WSN, scalable diagnosis.

1 Introduction

Wireless sensor networks are envisioned to consist of thousands of nodes, each subject to tight communication, storage and computation constraint. Irrespective of their purpose, all WSNs are characterized by the requirement for energy efficiency, scalability and fault tolerance. There are certain issues which need to be addressed for the sustained operations of WSN- 1) WSN consisting of sensor nodes may be deployed in unattended and possibly hostile environments which increases probability of node failure, 2) unlike wireless local area networks, the path between the source and the destination in wireless sensor networks normally contains multiple wireless links (hops). The wireless links between nodes are susceptible to wireless channel fading which causes channel errors, 3) data generated by each sensor node is routed to the sink node. Erroneous data generated by faulty sensor nodes must be protected from entering the network for effective bandwidth utilization. These issues have been studied and system level diagnosis appears to be a viable solution to the above posed problem.

Diagnosis by comparison [1,2] is a realistic approach to the fault diagnosis of massive networks. In comparison based diagnosis model, same job is assigned to pair of nodes and the outcome of this job is compared. The agreements and the disagreements among the nodes are the basis for identifying the fault free population. This paper follows this diagnosis approach where heartbeat message

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 69–80, 2011.

is broadcasted periodically by each node and the diagnosis job assigned to nodes is to increment the heartbeat sequence number.

The process of local detection and global diagnosis from a given fault instance is a multifaceted problem. The specific contributions of this paper are listed as follows:

1. Proposes a generic diagnosis scheme that identifies hard and soft faults[1] with high accuracy by maintaining low time, message and energy overhead.
2. Investigates fault diagnosis in presence of channel fault.
3. presents both analytical and simulation analysis to prove the correctness and completeness of the algorithm.

The protocol comprises of two main stages: detection phase and disseminating phase. In detection phase each node generates a heartbeat message and then broadcast it to neighbouring nodes. Each fault free and soft faulty node responds to the heartbeat request with in bounded time. At the end of this stage each node has the local view regarding the fault status of its 1-hop neighbours. The dissemination phase disseminates this local view in the network with the aid of a spanning tree (ST) where the ST spans all the fault free nodes in the WSN. The ST is created at the time of deployment and is maintained at each diagnosis round.

2 Related Works

System-level fault diagnosis was introduced by Preparata, Metze and Chien in 1967 [3], as a technique intended to diagnose faults in a wired inter connected system. Previously developed distributed diagnosis algorithms were designed for wired networks [3,2,4,1] and hence not well suited for wireless networks.

The problem of fault detection and diagnosis in wireless sensor networks is extensively studied in literatures [5,6,7,8,9,10,11]. The problem of identifying faulty nodes (crashed) in WSN has been studied in [5]. This article proposes the WINdiag diagnosis protocol which creates a spanning tree (ST) for dissemination of diagnostic information. Thomas *et al.* [6] have investigated the problem of target detection by a sensor network deployed in a region to be monitored. The performance comparison was performed both in the presence and in the absence of faulty nodes. Elhadef *et al.* have proposed a distributed fault identification protocol called Dynamic-DSDP for MANETs which uses a ST and a gossip style dissemination strategy [7]. In [8], a localized fault diagnosis algorithm for WSN is proposed that executes in tree-like networks. The approach proposed is based on local comparisons of sensed data and dissemination of the test results to the remaining sensors.

In [9] the authors present a distributed fault detection algorithm for wireless sensor networks where each sensor node identifies its own state based on local

[1] A different approach of classifying faults could be found in literature. Faults are classified as: crash, omission, timing, and Byzantine. Crash faults are hard faults, and all the others can be treated soft faults.

comparisons of sensed data against some thresholds and dissemination of the test results. The fault detection accuracy of a detection algorithm would decrease rapidly when the number of neighbour nodes to be diagnosed is small and the nodes failure ratio is high. Krishnamachari *et al.* have presented a Bayesian fault recognition algorithm to solve the fault-event disambiguation problem in sensor networks [10].

3 Modeling and Problem Statement

3.1 Communication Model and Assumptions

The system under consideration accommodates n number of nodes. Each node occupies a position (x, y) inside of a fixed geographic area $(l \times l\ m^2)$ and are uniformly distributed. Two nodes n_i and n_j are within transmission range r_{tx}, if the Euclidean distance $d(n_i, n_j)$ between n_i and n_j is less than r_{tx}. The topology graph $G = (V, E)$ consists of a set of vertices V representing the nodes of the network and the set E of undirected edges corresponding to communication links between nodes.

The model makes the following assumptions: 1) Static fault situation i.e no node is allowed to be faulty during diagnosis period T_{diag}, 2) Links are symmetric, i.e., two nodes v_i and v_j can communicate using the same transmission power level, 3) Energy consumption is not uniform for all nodes and 4) Communication channels between the nodes have bounded delay.

3.2 Fault Model

The proposed scheme considers both hard and soft fault. In hard-fault situation the sensor node is unable to communicate with the rest of the network (transceiver is faulty or battery is drained or node is completely damaged), whereas a node with soft-fault continues to operate and communicate with altered behaviour. These malfunctioning (soft faulty) sensors could participate in the network activities since still they are capable of routing information. The algorithm uses a time-out mechanism for finding the hard fault population. Since WSN is an arbitrarily connected network we define the connectivity (k) as: the minimum number of sensor nodes, the removal of which can cause the network to become disconnected. This definition puts a restriction on happening of number of faulty nodes in the network at any given instant of time.

3.3 Energy Consumption Model

In our model first order radio model [12] is used to formulate the energy consumed by transceiver system. The energy consumed in transmitting a bit to distance d, E_{TX}, and the energy in receiving a bit, E_{RX}, are respectively

$$E_{TX} = E_{elec} + \epsilon_{amp} \times d^{\alpha} \tag{1}$$

$$E_{RX} = E_{elec} \tag{2}$$

Where E_{elec} is the energy consumed by the transmitter/receiver electronics per bit, ϵ_{amp} is the energy dissipated by power amplifier in Joules per bit per m^2 and α is the path loss parameter. The proposed model takes α as 2.

3.4 Channel Model

The two-state Markov channel model [13] is adopted with two states: G (good) state and B (bad) state. In the good state the bits are received incorrectly with probability P_{good} and in the bad state the bits are received incorrectly with probability P_{bad}. For this algorithm it is assumed that $P_{good} \ll P_{bad}$. To simulate burst noise, the state of B and G must tend to persist: i.e., the transition probability $T_{GB} = P(G \rightarrow B)$ and $T_{BG} = P(B \rightarrow G)$ will be small and the probability remaining in G and B is large. The steady-state probability of a channel being in the bad state is $P_B = T_{GB}/(T_{GB} + T_{BG})$ and the average bit error probability of the channel is $P_e = P_{bad} \times P_B + P_{good}(1 - P_B)$.

3.5 The Diagnosis Problem

Assume that n nodes are dispersed in a field and the above assumptions hold. Our goal is to achieve correct and complete diagnosis. To achieve this the following requirements (discussed in section 5) must be met: 1) Diagnosis is completely distributed, 2) At the end of diagnosis round, each node is diagnosed as either a faulty or fault free, 3) Regardless of network diameter, diagnosis terminates within a fixed amount of time and 4) Diagnosis should be efficient in terms Message, time and energy complexity.

4 The Proposed Algorithm

This section introduces a new energy aware scalable diagnosis protocol for WSN. This work follows the general principle of diagnosis algorithms where working nodes perform their own independent diagnosis of the system. In the proposed work each node maintains a fault table (FT) of n entries. The j^{th} entry in the table stored in node w contains node w' s view of node j. Possible values are faulty (1) and fault free (0). Each node maintains a data structure. The data structure stored in node w contains; Neighbour: node IDs of $N(w)$, Depth: depth of node in ST, Children: IDs of child nodes in ST, Parent: ID of the parent node in ST. The proposed diagnosis protocol follows two phases: 1) Detection phase and 2) Dissemination phase.

4.1 Detection Phase

As discussed earlier each node in the WSN broadcast a heartbeat message periodically to acquire the local diagnosis view of its 1-hop neighbours. A heartbeat message has the following fields: identification number (ID) of the node that

Algorithm 1. Detection Phase

```
// response_flag is initialized to false
Generate and broadcast test message.
Increment the heartbeat sequence number.
set timer $T_{out}$
repeat
    if $(v.seq\_no \neq w.seq\_no - 1)$ $OR$ $(v.seq\_no \neq w.seq\_no)$ then
        $F_w \leftarrow F_w \cup \{v\}$ // soft faulty
    else
        if $(v.seq\_no = w.seq\_no - 1)$ $AND$ $(v \notin F_w)$ then
            if $response\_flag = false$ then
                $v.seq\_no \leftarrow v.seq\_no + 1$
                $broadcast[w, v.seq\_no]$
                $response\_flag \leftarrow true$
            end if
        end if
        if $v.seq\_no = w.seq\_no$ then
            $FF_w \leftarrow FF_w \cup \{v\}$
        else
            $F_w \leftarrow F_w \cup \{v\}$
        end if
        if $T_{out} = true$ then
            $F_w \leftarrow F_w \cup \{N(w) - (F_w \cup FF_w)\}$ // Initially detected as hard faulty
        end if
    end if
until $(F_w \cup FF_w \neq N(w))$
if $w.parent \in F_w$ then
    Find the node with lowest depth from $FF_w$ and declare it as new parent of $w$
end if
```

initiated the heartbeat message, physical sequence number (seq_no) of the heartbeat. After broadcasting the test message each node increments the heartbeat sequence number and starts a timer. Upon reaching the predefined upper bound the heartbeat sequence number is reset to 1. The detection phase uses timeout mechanism to detect hard faulty nodes. In the proposed algorithm every node maintains a neighbor table. The node w_i declares node $w_j \in N(w_i)$ as possibly hard faulty (initial detection status), if w_i does not receive heartbeat message before T_{out}. In this scheme w_j cannot report to w_i, if either the communication subsystem of w_j is faulty or the communication channel E_{ij} is faulty. For faulty communication channel w_i will mark w_j as hard faulty which may not be possibly correct. Final decision regarding w_j (hard faulty or fault free) is taken during dissemination phase. T_{out} should be chosen carefully so that all the fault free nodes $w_j \in N(w_i)$ connected by fault free channels E_{ij} must report node w_i before T_{out}. This phase first checks the consistency of the message received by evaluating the heartbeat sequence number of the sender. If the received message failed in consistency check, then sender of the message is declared as soft faulty. The proposed scheme considers the heartbeat sequence number as a test task and incrementing the heartbeat sequence number as response result. The received message can be a test message or a response message or a time out occurrence.

Test message : When an arbitrary node w_i receives a test message from a node $v \in N(w_i)$ at time t, it increments the heartbeat sequence number of received test message from v and broadcasts the test response at time t_1, where $t \leq t_1 \leq t + T_{out}$. The proposed algorithm allows the detection stage to respond only one test message (i.e. the earliest test message received), which reduces the communication complexity.

Response message : The IEEE 802.15.4 MAC layer ensures network synchronization. Thus, at time t the heartbeat sequence number of all sensor nodes of the WSN holds the same value. Upon receiving the response message from $v \in N(w_i)$, node w_i compares the heartbeat sequence number of v with its own heartbeat sequence number. If the comparison output agrees, node v diagnosed as fault free or else soft faulty.

At the end of the detection phase, each mobile aware of the fault status of its 1-hop neighbours. Upon time out occurred, the protocol starts ST maintenance phase where every node checks the status of their parent. If a node w_i discovers its parent in the fault set i.e. $w_i.parent \in F_w$; it searches for a new parent in the ST with lowest depth, where $newparent \in N(w_i)$. This ensures the existence of single ST.

4.2 Disseminating Phase

Dissemination phase starts at leaf nodes of the ST. Each parent must wait until it collects diagnostic information from all its children. Now, the parent combines the collected information to its own local diagnostic and sends this information to its parent in the ST. This process continues till the sink acquires the global view of the WSN. The sink node collects all the local diagnostics, and it disseminates the global diagnosis view down the tree to all sensor nodes. At the end of this phase every node in the WSN has the global view of the network and this completes the diagnosis. The ambiguity in detecting hard faulty nodes due to communication channel failure is handled by the nodes present at each level of ST. The ancestor sensor nodes in ST compares fault tables of their successor sensor nodes and take decision on hard fault. Let's assume, FT_{w_i} has marked w_j as hard faulty but FT_{w_k} detected it as fault free, then parent node declares w_j as fault free. If none other than w_i has any information regarding w_j then parent node follows FT_{w_i} which may not be correct. This incorrect decision is rectified in the higher level of the hierarchy in ST.

5 Algorithm Analysis

In this section, we analyze three performance metrics of the proposed protocol in a multi-hop wireless network; energy consumption, time and message complexity.

Algorithm 2. Dissemination Phase

```
$temp\_children \leftarrow \phi$
$local\_diagnosed \leftarrow global\_diagnosed \leftarrow false$
repeat
    if $v \in w.children$ then
        Compare $FT_w$ with $FT_v$ for correct set of hard faulty nodes.
        Update fault table $FT_w$ with $FT_v$.
        $temp\_children \leftarrow temp\_children \cup v$.
        if $|w.children| = |temp\_children|$ then
            Broadcast $FT_w$
        end if
    else if $w = sink$ then
        Start global dissemination by broadcasting $FT_{sink}$.
        $local\_diagnosed \leftarrow true$
    end if
until ($local\_diagnosed = false$)
repeat
    if $w.parent = v$ then
        Update fault table $FT_w$ with $FT_v$
        Broadcast $FT_w$
        if $w.children = \phi$ then
            $global\_diagnosed \leftarrow true$
        end if
    end if
until ($global\_diagnosed = false$)
```

The proposed protocol described meets the requirements listed in Section 3.5, as discussed next.

Observation 1. *The proposed model is completely distributed (requirement 1). Each sensor node makes a decision about its neighbour in the face of the evidences T_{out} and heartbeat sequence number.*

Lemma 1. *The set of faulty nodes in the WSN is uniquely identified based on the test results, when the fault population does not exceeds a upper bound δ, where $\delta \leq k - 1$.*

Proof. We analyse the upper bound for diagnosability of a WSN $G = (v, E)$ as a function of connectivity (k) of the network. Assume, on the contrary, that $\delta > k - 1$ and the WSN is δ diagnosable. For $\delta > k - 1$, the network loses its connection resulting formation of sub networks. Let G_1 and G_2 are the resulting sub networks, such that node $v_i \in G_1$, node $v_j \in G_2$ and $v_i, v_j \in G$. Now v_i sends disseminating message to v_j but failed, since $v_i \notin G_2$. Hence, diagnosis will never complete, which is a contradiction. It follows that the WSN is δ diagnosable, where $\delta \leq k - 1$.

Lemma 2. *Let, an arbitrary connected graph represents a δ diagnosable WSN. At the end of disseminating phase each node in WSN holds a unique fault set.(requirement 2)*

Proof. Lemma 1 ensures that the WSN remains always connected and fault free status of a node is at least known to one neighbour. Without losing the generality we can assume that a fault free node correctly checks the status of a node, i.e. the fault set F_w and the fault free set FF_w of an arbitrary fault free node w is correct. Thus, each working node maintains a correct status (i.e local diagnostic) of its neighbours. Lemma 1 also ensure the presence of a single ST during the dissemination phase. Since the ST spans all the working nodes, at the end of local dissemination phase the root holds a unique fault set.

The upper bound time complexity will be expressed in terms of T_p: an upper bound to the time needed to propagate a message between sensor nodes.

Lemma 3. *The proposed diagnosis algorithm terminates before time $T_p(2d_{st} + 3) + T_{out}$. Where, d_{st} is the depth of the spanning tree (requirement 3).*

Proof. All nodes initiate a heartbeat message simultaneously, which reaches at the neighbouring nodes by at most T_p of time. Each node on reception of heartbeat message evaluates the heartbeat sequence number and then initiates a response message. The farthest neighbouring node receives this response message after time T_p. Any fault-free node diagnoses at least one fault-free neighbor in at most T_{out} time. At the end of this phase, nodes with faulty parents send the adopt request which needs at most T_p amount of time. In at most $d_{st}T_p$, the sink node has collected all diagnostic views and disseminates the global diagnostic view that reaches the farthest mobile in at most $d_{st}T_p$. Thus, the disseminating phase requires $2d_{st}T_p$ time to complete. Now, the upper bound time complexity can be expressed as $T_{cost} = T_p(2d_{st} + 3) + T_{out}$

Lemma 4. *The proposed model has a worst-case message exchange complexity $O(n)$ in the network (requirement 4).*

Proof. The diagnosis starts at each node by sending the test message to neighbours costing one message per node i.e. n messages in the network. In next phase as each node responds to at most one test message (earliest received message), n number of messages are exchanged in the network. At the end of detection phase, nodes with missing parent (parent may be faulty or out of transmission range) updates its parent field. In worst case $n - 1$ messages are exchanged in the network. Each node, excluding the sink, sends one local diagnostic message. Each node, excluding the leaf nodes, sends one global diagnostic message and in worst case depth of ST is $n - 1$. Thus, Message cost for disseminating diagnostic messages is $2n - 1$. Now, the total number of exchanged messages is $M_{cost} = 5n - 3 = O(n)$.

5.1 Energy Consumption

Based on the description of the proposed protocol, each node broadcasts the test message and responds to at most one test message. The energy cost for this is given by

$$E_1 = 2nrE_{TX} \tag{3}$$

Where, r is the number of bits per test and response message. Upon receiving the test message, each node generates and broadcast response message. Energy spent in this process is

$$E_2 = 2MnrE_{RX} \tag{4}$$

Where M is the number of neighbours. The worst case energy spent in maintaining ST is given by

$$E_3 = (n-1)rE_{TX} + M(n-1)rE_{RX} \tag{5}$$

The energy cost in disseminating diagnostic information is given by

$$E_{local} = n_{ft}\left(\sum_{i=1}^{n_c} E_{TX} + \sum_{\forall i \neq leaf\ \ node}^{n_c} E_{RX}\right) \tag{6}$$

$$E_{global} = n_{ft}\left(NE_{RX} + \sum_{\forall i \neq leaf\ \ node}^{n_c} E_{TX}\right) \tag{7}$$

Where, n_{ft} is the number of bits per fault table. Finally, the energy overhead of the proposed protocol is

$$E_{overhead} = \frac{E_1 + E_2 + E_3 + E_{local} + E_{global}}{n} \tag{8}$$

5.2 Simulation Results

This work uses ns-3 as the simulation tool. The free space physical layer model is adopted where all nodes within the transmission range of a transmitting node receive a packet transmitted by the node after a very short propagation delay. The set of simulation parameters are summarized in Table 1. Connectivity of the WSN is fixed at $k = 55$. For simplicity in the simulation P_{good} is taken as 0 and P_{bad} is taken as 1. T_{BG} is fixed to 1/8 and T_{GB} is varied to get different channel error probabilities P_e. Every result shown is the average of 100 experiments. Each experiment uses a different randomly-generated topology.

Fig. 1.a demonstrates the time complexity of the proposed scheme. From Lemma 3 it is obvious that dissemination of diagnostics contributes more to diagnosis latency. The depth of the ST decides the diagnosis latency, as it is used to disseminate diagnostics. Thus, as expected the time required to diagnose the WSN increases almost linearly with increase of number of nodes. Fig. 1.b analyses the diagnosis latency with respect varying size of fault population. Since the

Table 1. Simulation Parameters

Parameter	Value
Number of nodes	1000
Network grid	From (0, 0) to (1000m, 1000m)
Sink	At (100,150)
Simulation time	150 sec.
MAC protocol	IEEE 802.15.4
Frequency	2.4 GHz
Initial energy	2J/battery
Propagation delay	10μ sec.
E_{elect}	5×10^{-9} J/bit
ϵ_{amp}	100×10^{-12} $J/bit/m^2$
Antenna model	Omni directional

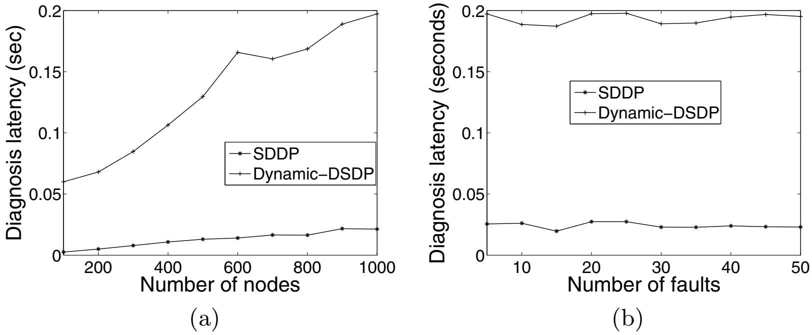


Fig. 1. Time complexity

protocol ensures the creation of the ST at the network deployment time, hence increase of fault population has negligible effect on its depth. This statement is validated by the simulation results shown in Fig. 1.b, where the diagnosis latency is less affected by change in fault population. Here we generate random sets of faulty mobile nodes, where the number of faulty mobiles ranged from 10 to 50.

Fig. 2.a shows the communication complexity of the proposed protocol. From the simulation result it is evident that the communication complexity of this work outperforms dynamic DSDP. Energy consumption by each node is proportional to the amount of traffic it generates or receives. Thus, the energy overhead of the proposed protocol is less than Dynamic-DSDP which in turn improves the network lifetime of a WSN. Fig. 2.b illustrates the message complexity verses the number of faults. As shown, the message complexity decreases with increase of number of faults since participating nodes in dissemination of diagnostics decreases. Fig.3 presents a comparison view of energy overhead in diagnosing fault events. As expected energy overhead of proposed algorithm is less than dynamic-DSDP and Chessa *et al.* model. Table 2 provides a comparison of time and message complexity to related approaches. The proposed model outperforms

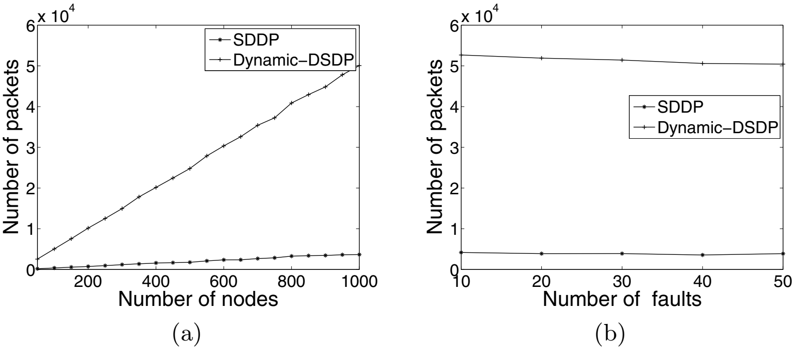


Fig. 2. Message complexity

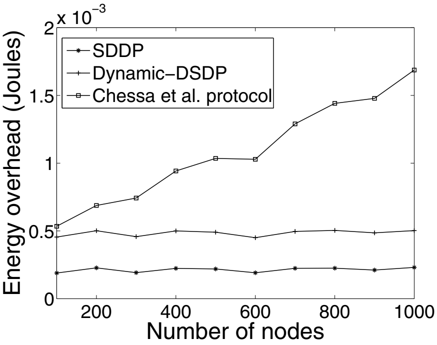


Fig. 3. Energy complexity

that of these self diagnosis schemes from both time and message complexity perspective. Message complexity of [14] is $O(N^2)$ and message complexity of [7] will be $O(N^2)$ for $k = N$ (i.e. fully connected network), where as message complexity of proposed model in $O(N)$.

Table 2. Comparison with related works

	Message complexity	Time complexity
Chessa *et al.* model	$nd_{max} + n(n+1)$	$\delta_G T_{gen} + \delta_G T_p + T_{out}$
Dynamic-DSDP	$nk + 3n - 1$	$\delta_G T_{gen} + 3d_{st} T_p + 2T_{out}$
SDDP	$5n - 3$	$T_p(2d_{st} + 3) + T_{out}$

d_{max}: The maximum of the node degree.

δ_G: The diameter of graph G.

T_{send}: The upper bound to the time need to solve contention.

T_{gen}: upper bound to time between reception of the first diagnostic. message and the generation of test request.d_{st}: Depth of the ST.

6 Conclusion

This paper addresses the fundamental problem of identifying faulty (soft and hard) sensors in WSN under channel impairment. The message complexity of the proposed algorithm is $O(n)$ which is significantly low compared to present state of art approaches. Due to low energy consumption and reduced complexity the algorithm could be integrated to error resilient transport protocols in wireless sensor networks. A natural extension of the algorithm is to solve the transient and intermittent fault problem. Currently work is going on to develop a algorithm to identify transient and intermittent faults with lower message cost and same or less latency.

References

1. Malek, M.: A comparison connection assignment for diagnosis of multiprocessor systems, pp. 31–36. ACM Press, New York (1980)
2. Blough, D.M., Brown, H.W.: The broadcast comparison model for on-line fault diagnosis in multicomputer systems: theory and implementation. IEEE Transactions on Computers 48(5), 470–493 (1999)
3. Preparata, F.P., Metze, G., Chien, R.T.: On the connection assignment problem of diagnosable systems. IEEE Transactions on Electronic Computers EC-16(6), 848–854 (1967)
4. Subbiah, A., Blough, D.: Distributed diagnosis in dynamic fault environments. IEEE Transactions on Parallel and Distributed Systems 15(5), 453–467 (2004)
5. Chessa, S., Santi, P.: Crash faults identification in wireless sensor networks. Computer Communications 25(14), 1273 (2002)
6. Clouqueur, T., Saluja, K.K., Ramanathan, P.: Fault tolerance in collaborative sensor networks for target detection. IEEE Transactions on Computers 53(3), 320–333 (2004)
7. Elhadef, M., Boukerche, A., Elkadiki, H.: A distributed fault identification protocol for wireless and mobile ad hoc networks. Journal of Parallel and Distributed Computing 68(3), 321–335 (2008)
8. Xu, X., Chen, W., Wan, J., Yu, R.: Distributed fault diagnosis of wireless sensor networks. In: ICCT, pp. 148–151 (November 2008)
9. Lee, M.-H., Choi, Y.-H.: Fault detection of wireless sensor networks. Computer Communications 31(14), 3469 (2008)
10. Krishnamachari, B., Iyengar, S.: Distributed bayesian algorithms for fault-tolerant event region detection in wireless sensor networks. IEEE Transactions on Computers 53(3), 241–250 (2004)
11. Chen, J., Kher, S., Somani, A.: Distributed fault detection of wireless sensor networks. In: DIWANS, pp. 65–72. ACM Press, New York (2006)
12. Heinzelman, W., Chandrakasan, A., Balakrishnan, H.: Energy-efficient communication protocol for wireless microsensor networks, vol. 2, p. 10 (January 2000)
13. Gilbert, E.N.: Capacity of a burst-noise channel. Bell Syst. Tech. J., 1265–1253 (1960)
14. Chessa, S., Santi, P.: Comparison-based system-level fault diagnosis in ad hoc networks. In: 20th IEEE Symposium on Reliable Distributed Systems, pp. 257–266 (2001)

Cryptanalysis of Chaos Based Secure Satellite Imagery Cryptosystem

Musheer Ahmad

Department of Computer Engineering, Faculty of Engineering and Technology,
Jamia Millia Islamia, New Delhi 110025, India

Abstract. Recently Usama et al. proposed a chaos-based satellite image cryptosystem, which employed multiple one-dimensional chaotic maps in novel manner to enhance the robustness, security and efficiency of sensitive satellite imagery. It is very efficient in terms of encryption time. The authors of the cryptosystem under study claimed that it has high level of security and can be applied to transmit confidential multimedia images over Internet/shared network. Unfortunately, the security analysis of the cryptosystem reveals that it has serious security flaws. Consequently, it is susceptible to a number of attacks. In this paper, the cryptanalysis of original cryptosystem is presented and it is shown that the attacker can recovers the plain-image from given cipher-image under three types of classical cryptographic attacks without knowing the secret key. The simulation results of cryptanalysis demonstrate that the cryptosystem highly lacks security and cannot be utilized for the protection of confidential/sensitive multimedia images such as the satellite imagery.

Keywords: Satellite image cryptosystem, chaotic maps, security, cryptanalysis, cryptographic attacks.

1 Introduction

Nowadays, modern multimedia and telecommunications technologies make possible to share, exchange and transmit large amount of multimedia data more frequently. This brings challenges to build faster and stronger security solutions for confidential and sensitive multimedia data to be transmitted over the public wired or wireless networks. Inadequate security can leads to unauthorized access, usage or disruption of data. Traditional cryptographic algorithms such as DES, triple-DES, AES, are considered inefficient in providing ample security to multimedia data that has bulk data capacity and high redundancy [1]. The features of chaotic systems like high sensitivity to initial conditions/parameters, non-periodicity, high randomness, mixing property etc have been highly exploited for the design of efficient security methods that suit for multimedia data. An enormous numbers of chaos-based multimedia image and video encryption proposals have been suggested [2-19] since the arrival of first such proposal given by R. Mathews in 1989 [2]. For a thorough discussion of chaos-based image and video encryption techniques, readers are referred to some review and study [20-21]. The work of assessing the security of the proposed

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 81–91, 2011.

multimedia encryption techniques is equally significant; it has been performed with the intent to arrive at more robust, reliable and efficient security solutions. As a consequence, the security analyses of proposed chaos-based multimedia encryption techniques have also been performed. It has been found that some of them suffer from various security weaknesses and are incompetent to withstand even the classical and other types of cryptographic attacks, as exposed by many cryptanalysts in the literature [22-27].

Recently, Usama *et al.* [16] proposed a new chaos-based satellite imagery cryptosystem. The cryptosystem is a block cipher which employed multiple one-dimensional chaotic maps e.g. Logistic map, Henon map, Tent map, Cubic map, Sine map and Chebyshev map for enhancing the key space, robustness and security of satellite imagery in novel manner. The experimental and security analyses illustrate that the cryptosystem has high robustness and security. The cryptosystem is very fast as it incurs a very low encryption time. Moreover, the distinctive feature of the algorithm is that it uses a variable length secret key and generates a number encryption keys out of it. In spite of this, the security analysis of the proposed satellite image cryptosystem exposes its serious security flaws from cryptographic viewpoint. Consequently, it is susceptible to the classical cryptographic attacks. In this paper, the satellite image cryptosystem described in [16] is successfully cryptanalyzed. It is shown that we can recover the original plain-image from given cipher-image using three types of attacks (chosen-plaintext, chosen-ciphertext and known-plaintext attacks) without knowing the secret key. Moreover, it is also shown that the cryptosystem is not at all sensitive to a small change in the plain-image, which is a very desirable feature of a good cryptosystem. The outline of the rest of paper is as follows: Section 2 briefly describes the satellite image cryptosystem under study. The cryptanalysis results with simulations are illustrated in Section 3 and finally the conclusions are drawn in the Section 4.

2 Brief Description of Cryptosystem under Study

This section concerns with the review and description of the cryptosystem recently proposed in [16]. The cryptosystem is a block cipher in which the efficiencies of six one-dimensional chaotic maps are exploited to improve the security of the cryptosystem by enhancing its confusion and diffusion properties. The 1D chaotic maps namely Chebyshev Map, Logistic Map, Cubic Map, Sine Map, Henon Map and Tent Map are employed in the system, the governing equations of chaotic maps are:

Chebyshev Map	:	$x_{n+1} = cos(\lambda cos^{-1}(x_n))$
Logistic Map	:	$x_{n+1} = \lambda x_n(1-x_n)$
Cubic Map	:	$x_{n+1} = \lambda x_n(1-x_n * x_n)$
Sine Map	:	$x_{n+1} = \lambda sin(\pi x_n)$
Henon Map	:	$x_n = 1+ \lambda(x_{n-2} - x_{n-3})+ax_{n-2} * x_{n-2}$
Tent Map	:	$x_{n+1} = x_n/\mu$ if $x_n \leq \mu$ else $(1-x_n)/(1-\mu)$

The Usama *et al.* cryptosystem takes an integer value *n* and a variable length secret key *S* of ρ (=128/256/512) bits as input. It evaluates the initial conditions of all the 1D chaotic maps using the secret key *S*. Reader may consult the Table 2 in [16] for the initial values of system parameters taken. The chaotic maps generate conjointly *n* number of encryption keys $\boldsymbol{K_i}$ each of ρ bits. The plain-image *P* is broken into *m* number of blocks of size ρ. The blocks of satellite plain-image are encrypted sequentially using generated encryption keys K_i to produce blocks of cipher-image *C*.

Secret key	:	$S = S_1S_2S_3 \ldots\ldots\ldots S_\rho$
Plain-image	:	$P = P_1P_2P_3 \ldots\ldots P_m$
Cipher-image	:	$\mathrm{C} = C_1C_2C_3 \ldots\ldots C_m$

The size of block P_j/C_j (j = 1 to *m*) is equal to size of secret key *S*. The secret key *S* is converted into byte format as: $S = B_1B_2B_3 \ldots\ldots\ldots B_\sigma$ (where $\sigma=\rho/8$) and the same initial condition *IC* of all one-dimensional chaotic maps is calculated from the secret key *S* through following equations:

$$N = \sum_{i=1}^{\sigma} (decimal(B_i)/256)$$

$$IC = N - floor(N)$$

Using the initial condition *IC*, the chaotic maps are iterated to produce *n* keys, each of ρ bits, the keys generated from each map can be expressed as.

Chebyshev map keys	$\boldsymbol{BK}$: $BK_1, BK_2, BK_3, \ldots\ldots\ldots\ldots\ldots\ldots BK_n$
Logistic map keys	$\boldsymbol{LK}$: $LK_1, LK_2, LK_3, \ldots\ldots\ldots\ldots\ldots\ldots L\ K_n$
Cubic map keys	$\boldsymbol{CK}$: $CK_1, CK_2, CK_3, \ldots\ldots\ldots\ldots\ldots\ldots CK_n$
Sine map keys	$\boldsymbol{SK}$: $VK_1, SK_2, SK_3, \ldots\ldots\ldots\ldots\ldots\ldots SK_n$
Henon map keys	$\boldsymbol{HK}$: $HK_1, HK_2, HK_3, \ldots\ldots\ldots\ldots\ldots\ldots HK_n$
Tent map keys	$\boldsymbol{TK}$: $TK_1, TK_2, TK_3, \ldots\ldots\ldots\ldots\ldots\ldots\ldots TK_n$

The cryptosystem creates *n* keys from each chaotic map using single secret key *S*. The above set of keys are then combined through XOR operation to generate the *n* distinct encryption keys $\boldsymbol{K_i}$, where i = 1, 2, 3, …. *n*.

$$\boldsymbol{K_i} = BK_i \oplus LK_i \oplus CK_i \oplus SK_i \oplus HK_i \oplus TK_i \qquad where\ i = 1\ to\ n$$

The block diagram of key generation procedure is shown in Figure 1. These keys are actually used to encrypt the blocks of plain-image using the XOR operation to get the blocks of cipher-image. The block diagram of encryption mechanism involved in the cryptosystem is shown in Figure 2. The decryption process is similar to the encryption process i.e. the encrypted image is decoded by simply XORing the blocks of cipher-image with keys generated through key generation process depicted in Figure 1 to obtained the whole decrypted image.

In Usama *et al.* cryptosystem, the way in which the features of multiple 1D chaotic maps are utilized for the sake of improved security of the cryptosystem is appreciable. Accordingly, the results of statistical analyses such as maximum deviation, information entropy, key space analysis, key sensitivity analysis and encryption time analysis given in section 4 of [16] reveal the virtuous and excellent performance of the cryptosystem. However, there still exist some weaknesses that may be exploited by the attacker to break the system.

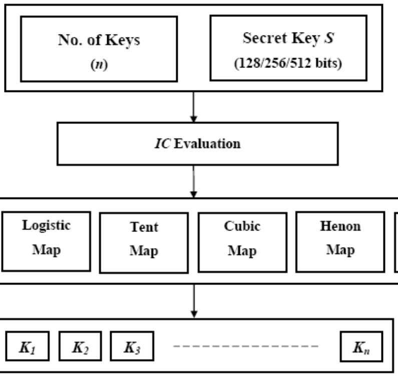


Fig. 1. Key generation process in encryption/decryption

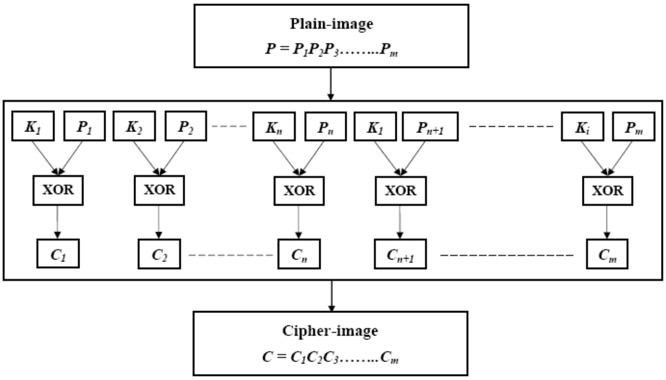


Fig. 2. Block diagram of Encryption mechanism

3 Cryptanalysis of Usama *et al.* Cryptosystem

In this section, the method of breaking the cryptosystem under study is reported. A cryptosystem is supposed to be secure if it resists all known types of cryptographic attacks. An attempted cryptanalysis is called an attack. In cryptanalysis, the fundamental assumption enunciated by Kerchoff is that the attacker knows the complete details of cryptographic algorithm and its implementation. This is known as Kerchoff's principle [1]. In other words, the attacker has the temporary access to the encryption and decryption machine. The goal of the attacker is to recover the plaintext without having any knowledge of secret key used. This is because recovering the plaintext is as good as deducing the secret key. Now, there are four types of classical cryptographic attacks. A brief description of each attack is given below:

1. ***Chosen-Plaintext attack***: In this case, the attacker has the temporary access to encryption machine; he cleverly selects one or more plaintext(s) and gets the corresponding ciphertext(s), which in turn allows the attacker to decode the received encrypted plaintext. This attack is one of the potential classical attacks.

2. ***Chosen-Ciphertext attack***: In this case, the attacker has the temporary access to decryption machine; he selects special ciphertext(s) and gets the associated plaintext(s), which helps in decoding the received encrypted plaintext.

3. ***Known-Plaintext attack***: In this case the attacker has the access not only to ciphertext(s) but also to the plaintext(s) of those ciphertext(s). This facilitates the attacker to deduce the key using these pair(s).

4. ***Ciphertext-only attack***: This is hardest type of attack, as the attacker posses only the ciphertexts of several plaintexts, all of which are encrypted using same encryption algorithm. The attacker tries to analyze them in order to deduce the key.

Any cryptographic algorithm which cannot resist any of the attack is said to be insecure. Therefore, the best cryptographic algorithms are the ones that have been made public, have been attacked by the world's best cryptanalysts for years, and are still unbreakable [1]. The serious weakness of the cryptosystem under study lies in the key generation process, which is same for every plain-image/cipher-image i.e. it is neither depend on the plain-image nor does it depend on the cipher-image, hence it remain unchangeable in every encryption/decryption process. This makes the classical attacks applicable to the cryptosystem. The details of cryptanalysis along with simulation under three different attacks are described in the following subsections.

3.1 Chosen-Plaintext Attack

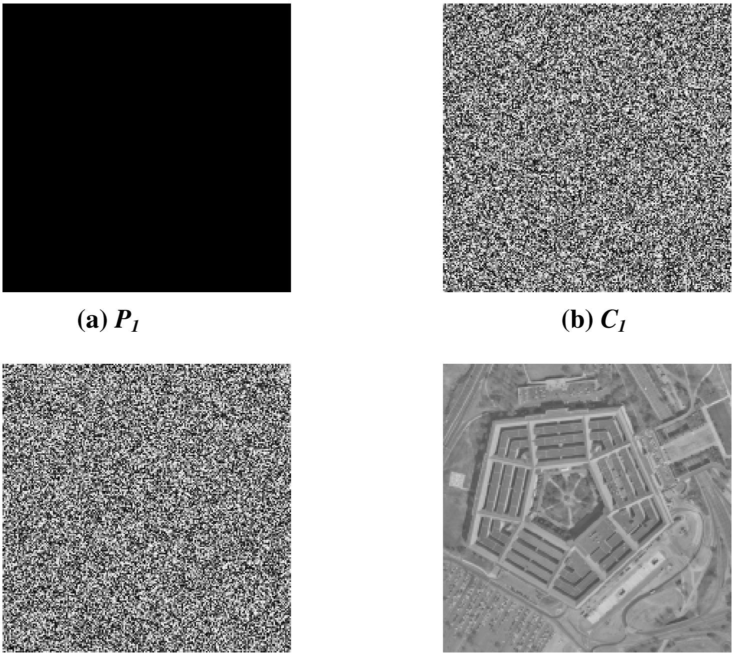
Assume that we have temporary access to the encryption machine and ciphertext C_2 which is to be decoded. Let us select a plain-image that consists of all zero-valued pixels i.e. $P_1=000......0$. The cipher-image C_1 corresponding to the plain-image P_1 is obtained using the encryption machine.

$$C_1 = P_1 \oplus K = (000......0) \oplus K = K$$

It can be easily understood that the cipher-image C_1 is nothing but the key K which was generated to encrypt the plain-image P_1 during the encryption process. As the key generation process is independent to the plain-image to be encrypted. This means that the same key K is used every time to encrypt any plain-image. Thus, the plain-image P_2 can be recovered from the received cipher-image C_2 as:

$$C_2 \oplus C_1 = C_2 \oplus K = P_2$$

The simulation of the chosen-plaintext attack is shown in Figure 3.



(a) $\boldsymbol{P_1}$ **(b)** $\boldsymbol{C_1}$

(c) $\boldsymbol{C_2}$ **(d)** $\boldsymbol{P_2}$

Fig. 3. Simulation of chosen-plaintext attack: (a) Selected plain-image $P_1=00.....0$, (b) Cipher-image C_1 of P_1 which is equal to the key (c) Received cipher-image C_2 to be decoded and (d) Recovered image $P_2 = C_2 \oplus C_1$

3.2 Chosen-Ciphertext Attack

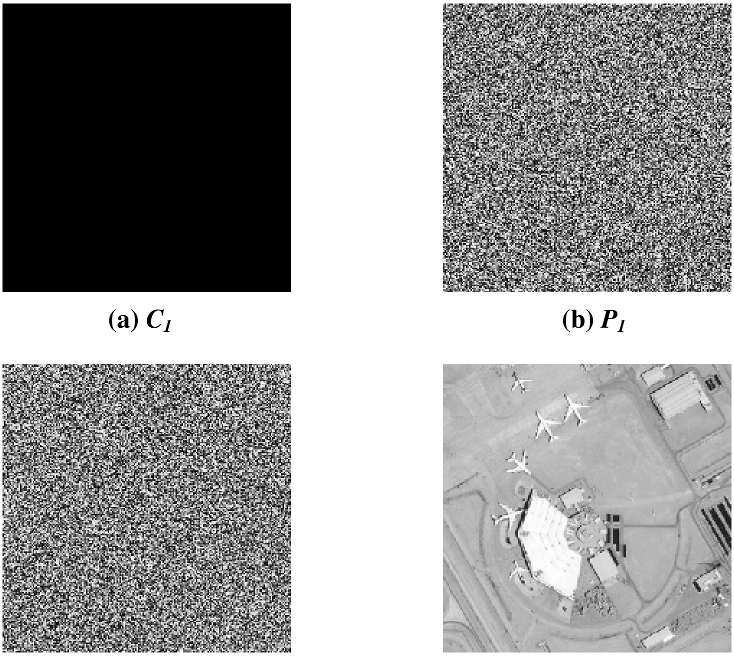
The approach of this attack is somewhat similar to the previous one. Assume that we have temporary access to the decryption machine and ciphertext C_2 which is to be decoded. We select a special cipher-image that consists of all zero-valued pixels i.e. $C_1 = 000......0$. The plain-image P_1 associated to the cipher-image C_1 is obtained using the decryption machine.

$$P_1 = C_1 \oplus K = (000......0) \oplus K = K$$

We get the secret key K as the plain-image P_1 for the chosen cipher-image. Thus, the plain-image P_2 can be recovered from the received cipher-image C_2 as:

$$C_2 \oplus P_1 = C_2 \oplus K = P_2$$

The simulation of the chosen-ciphertext attack is shown in Figure 4.



(a) $\boldsymbol{C_1}$ **(b)** $\boldsymbol{P_1}$

(c) $\boldsymbol{C_2}$ **(d)** $\boldsymbol{P_2}$

Fig. 4. Simulation of chosen-ciphertext attack: (a) Selected cipher-image $C_1=00.....0$, (b) Plain-image P_1 of C_1 which is equal to the key (c) Received cipher-image C_2 to be decoded and (d) Recovered image $P_2 = C_2 \oplus P_1$

3.3 Known-Plaintext Attack

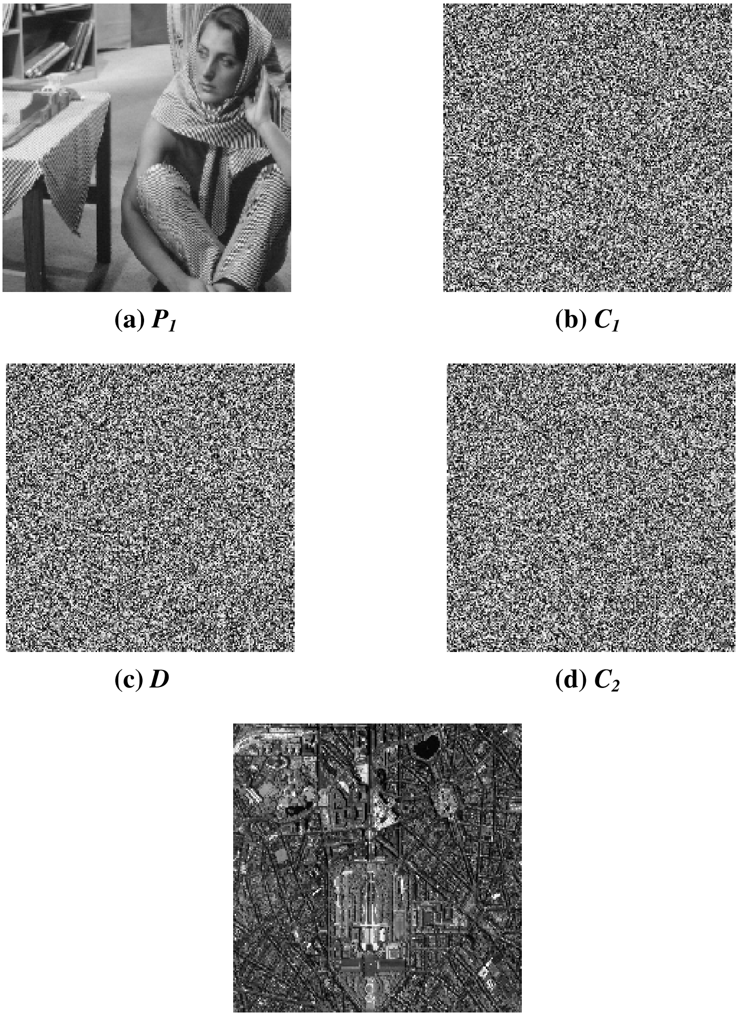
In this attack, we do not choose any special plain-image or cipher-image. Instead, we have access to a pair consists of plain-image P_1 and its associated cipher-image C_1, where

$$C_1 = P_1 \oplus K$$

Let we have to decode the received cipher-image C_2. Consider an intermediate image D which is XOR of images P_1 and C_1 i.e. $D = C_1 \oplus P_1$, the plain-image P_2 can be recovered from the received cipher-image C_2 under *KPA* attack as:

$$\begin{aligned} C_2 \oplus D &= C_2 \oplus \{C_1 \oplus P_1\} \\ &= C_2 \oplus \{(P_1 \oplus K) \oplus P_1\} \\ &= C_2 \oplus K \oplus \{P_1 \oplus P_1\} \\ &= C_2 \oplus K = P_2 \end{aligned}$$

The simulation of the chosen-ciphertext attack is shown in Figure 5.



(a) ***P_1***

(b) ***C_1***

(c) ***D***

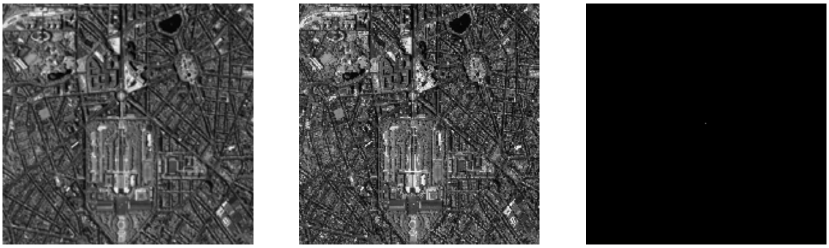
(d) ***C_2***

(e) ***P_2***

Fig. 5. Simulation of known-plaintext attack: (a) Plain-image P_1 (b) Cipher-image C_1 of P_1 (c) Intermediate-image $D=C_1 \oplus P_1$ (d) Received cipher-image C_2 and (e) Recovered image $P_2=C_2 \oplus D$

3.4 Sensitivity to Plain-Image

To fulfill the Shannon's requirements of confusion and diffusion properties in a cryptographic system for secure encryption, the cryptosystem should be very sensitive to a tiny change in the plain-image. Unfortunately, the cryptosystem under study is not at all sensitive to a small change in the plain-image. To understand the severity of the problem clearly, let us consider two plain-images I_1 and I_2 with only one pixel value difference at central position i.e. the pixel-values of image I_1 are identical to the pixels-values of image I_2 except the pixel-value positioned at centre. Since, the pixels of the two images are identical except the central one, the cipher-images J_1 and J_2 obtained after encrypting them using the cryptosystem under study will also be identical to each other. This is because the cryptosystem is not made sensitive to a change in plain-images. This weakness of poor sensitivity to plain-image is illustrated through a simulation example shown in Figure 6. It can be easily seen in the Figure 6 that the differential cipher-image is zero for all pixels except the central one.



(a) (b) (c)

Fig. 6. Sensitivity to Plain-image: (a) Plain-image I_1 (b) Plain-image I_2 and (c) Differential cipher-image $J_1 \oplus J_2$

4 Conclusion

In this paper, the security of chaos-based satellite imagery cryptosystem recently proposed by Usama *et al.* has been thoroughly analyzed. The cryptosystem has some good cryptographic properties. However, it has been found that the cryptosystem is susceptible to classical attacks like chosen-plaintext attack, chosen-ciphertext attack and known-plaintext attack. It has been shown that the plain-image can be recovered without knowing the secret key under the above attacks and only a pair of plain-image/cipher-image is needed to completely break the cryptosystem. Moreover, the cryptosystem has poor sensitivity to small change in the plain-image. The serious weakness of the cryptosystem lies in the key generation process, which is independent to the plain-image/cipher-image. One of the solutions to make the above attacks impractical is that design such cryptosystem in Cipher Block Chaining (CBC) mode of block encryption. Hence, the complete cryptanalysis of the cryptosystem is presented along with simulation. The work demonstrates that the Usama *et al.* cryptosystem highly lacks security and cannot be utilized for the protection of sensitive multimedia images such as satellite imagery.

References

1. Schneier, B.: Applied Cryptography: Protocols Algorithms and Source Code in C. Wiley, New York (1996)
2. Matthews, R.: On the Derivation of a Chaotic Encryption Algorithm. Cryptologia 13(1), 29–42 (1989)

3. Fridrich, J.: Symmetric Ciphers based on two-dimensional Chaotic Maps. International Journal of Bifurcation and Chaos 8(6), 1259–1284 (1998)
4. Chen, G.Y., Mao, Y.B., Chui, C.K.: A Symmetric Image Encryption Scheme based on 3D Chaotic Cat maps. Chaos, Solitons & Fractals 21(3), 749–761 (2004)
5. Mao, Y., Lian, S., Chen, G.: A Novel Fast Image Encryption Scheme based on 3D Chaotic Baker maps. International Journal of Bifurcation and Chaos 14(10), 3616–3624 (2004)
6. Lian, S., Sun, J., Wang, Z.: A Block Cipher based on a Suitable use of Chaotic Standard map. Chaos, Solitons and Fractals 26(1), 117–129 (2005)
7. Lian, S., Sun, J., Wang, J., Wang, Z.: A Chaotic Stream Cipher and the Usage in Video Protection. Chaos, Solitons & Fractals 34(3), 851–859 (2007)
8. Tong, X., Cui, M.: Image Encryption Scheme based on 3D Baker with Dynamical Compound Chaotic Sequence Cipher Generator. Signal Processing 89(4), 480–491 (2008)
9. Patidar, V., Pareek, N.K., Sud, K.K.: A New Substitution-Diffusion based Image Cipher using Chaotic Standard and Logistic Maps. Communication in Nonlinear Science and Numerical Simulation 14(7), 3056–3075 (2009)
10. Tang, Y., Wang, Z., Fang, J.: Image Encryption using Chaotic Coupled Map Lattices with Time Varying Delays. Communication in Nonlinear Science and Numerical Simulation 15(9), 2456–2468 (2009)
11. Lian, S.: Efficient Image or Video Encryption based on Spatiotemporal Chaos System. Chaos, Solitons & Fractals 40(5), 2509–2519 (2009)
12. Wang, Y., Wong, K., Liao, X., Xiang, T., Chen, G.: A Chaos-based Image Encryption Algorithm with Variable Control Parameters. Chaos, Solitons & Fractals 41(4), 1773–1783 (2009)
13. Wang, Y., Wong, K., Liao, X.: A Block Cipher with Dynamic S-boxes based on Tent Map. Communication in Nonlinear Science and Numerical Simulation 14(7), 3089–3099 (2009)
14. Lian, S.: A Block Cipher based on Chaotic Neural Networks. Neurocomputing 72(4-6), 1296–1301 (2009)
15. Corron, N.J., Reed, B.R., Blakely, J.N., Myneni, K., Pethel, S.D.: Chaotic Scrambling for Wireless Analog Video. Communication in Nonlinear Science and Numerical Simulation 15(9), 2504–2513 (2010)
16. Usama, M., Khan, M.K., Alghathbar, K., Lee, C.: Chaos-based Secure Satellite Imagery Cryptosystem. Computers and Mathematics with Applications 60(2), 326–337 (2010)
17. Amin, M., Faragallah, O.S., Abd El-Latif, A.A.: A Chaotic Block Cipher Algorithm for Image Cryptosystem. Communication in Nonlinear Science and Numerical Simulation 15(11), 3484–3497 (2010)
18. Ahmad, M., Farooq, O.: A multi-level blocks scrambling based chaotic image cipher. In: Ranka, S., Banerjee, A., Biswas, K.K., Dua, S., Mishra, P., Moona, R., Poon, S.-H., Wang, C.-L. (eds.) IC3 2010. Communications in Computer and Information Science, vol. 94, pp. 171–182. Springer, Heidelberg (2010)
19. Chen, Z., Ip, W.H., Cha, C.Y., Yung, K.: Two-level Chaos based Video Cryptosystem on H.263 Codec. Nonlinear Dynamics 62(3), 647–664 (2010)
20. Furht, B., Kirovski, D.: Multimedia Security Handbook. CRC Press, Boca Raton (2005)
21. Lian, S.: Multimedia Content Encryption: Techniques and Applications. CRC Press, Boca Raton (2008)
22. Wang, K., Pei, W., Zou, L.: On the Security of 3D Cat Map based Symmetric Image Encryption Scheme. Physics Letters A 343(6), 432–439 (2005)
23. Alvarez, G., Li, S.: Breaking an Encryption Scheme based on Chaotic Baker Map. Physics Letters A 352(1-2), 78–82 (2005)

24. Rhouma, R., Solak, E., Belghith, S.: Cryptanalysis of a New Substitution-Diffusion based Image Cipher. Communication in Nonlinear Science and Numerical Simulation 15(7), 1887–1892 (2010)
25. Rhouma, R., Belghith, S.: Cryptanalysis of a Spatiotemporal Chaotic Image/Video Cryptosystem. Physics Letters A 372(36), 5790–5794 (2008)
26. Li, C., Li, S., Asim, M., Nunez, J., Alvarez, G., Chen, G.: On the Security Defects of an Image Encryption Scheme. Image and Vision Computing 27, 1371–1381 (2009)
27. Rhouma, R., Belghith, S.: Cryptanalysis of a Chaos-based Cryptosystem on DSP. Communication in Nonlinear Science and Numerical Simulation 16(2), 876–884 (2011)

Efficient Regular Expression Pattern Matching on Graphics Processing Units

Sudheer Ponnemkunnath and R.C. Joshi

Department of Electronics and Computer Engineering,
Indian Institute of Techology Roorkee, Roorkee, India
{sudhipec,rcjosfec}@iitr.ernet.in,
p.sudheer21@gmail.com

Abstract. Regular expression signature matching has been used increasingly in network security applications like intrusion detection systems, virus scanners, network forensics, spam filters etc. However, signature matching causes decrease in performance on the host when load increases due to the large requirements in terms of memory and processing power. This is mainly because every byte and possibly a combination of bytes of the input have to be matched against a large set of regular expressions. Modern Graphics Processing Units (GPUs) are capable of high performance computing and recently are being used for general purpose computing. The large performance throughput and data parallelism of these modern GPUs is used to perform matching on the input data in parallel. Experimental results show that our GPU implementation is up to 12 times faster than the traditional CPU implementation while being up to 4 times faster than the GPU implementation using texture memory.

Keywords: Signature matching, GPU, pattern matching, parallel computing.

1 Introduction

Signature matching using regular expressions is of great importance in network security to detect and prevent well known attacks. Many network intrusion detection systems like Snort [1] and Bro [2] use deep packet inspection for checking whether a packet contains an attack vector or not. Earlier string literals were used for performing this detection. However, making use of string literals for detection causes a large amount of false positives during the detection process due to the existence of loose signatures [3]. Whereas regular expressions being more flexible can be used to describe a large number of signatures and have been increasingly being used to represent attack patterns for virus scanners and network intrusion detection systems.

However Signature matching is a highly computationally intensive process, accounting for about 70% of the total CPU processing time of modern NIDSs like Snort [4]. A conventional way to improve performance is to make use of hardware technologies like ASICs, FPGAs and TCAMs [5,6,7,8]. They can be used to achieve high performance throughput; however they have weak scalability, almost no flexibility and are expensive. Multi core chips can also be used but even they are not

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 92–101, 2011.

fast enough. Network processors are also being used for intrusion detection as they are optimized for network packet processing but they may not be suitable for network forensics or for antivirus applications and they are expensive.

Commodity graphics processing units (GPUs) have been proven to be very efficient for accelerating the string searching operations of network security applications. The current generation of GPUs have many stream processors and support thousands of concurrent threads and hence can be used for compute intensive applications. Many attempts have been made to use the processing power of the GPUs for security purposes of intrusion detection, network forensics and virus detection.

This paper proposes the use of GPUs to perform regular expression pattern matching efficiently while accelerating the string searching operations that can be used with any of the network security applications. Improvements have been made to the implementation proposed by Vasiliadis et al [9] which provides up to 4 times improvement in performance. These improvements cause the GPU implementation to be up to 15 times faster than the CPU implementation.

We proceed in the remainder of the paper as follows. In section 2 a survey of the related work is provided that tries to offload pattern matching computation on the GPU. In section 3 we provide a brief introduction of GPUs and its programming model. In section 4 we provide the introduction to pattern matching using Finite automaton and some modification that are done to create the automaton for use in our pattern matching engine. In section 5 we give the model and implementation details of implementing the pattern matching engine on the GPU. In section 6 we provide the experiments and show results that show that this implementation is faster than the previous implementations. Finally in section 7 we draw the conclusion and discuss future directions for this work.

2 Related Work

One of the first works in using GPUs for pattern matching was done by N. Jacob, et al which they showed that GPUs can be used to accelerate pattern matching [10].Their implementation “PixelSnort” was done by porting the pattern matching code on Cg. Since the coding was done by the traditional GPU programming model, utilizing the computational units on the graphics card had to be done in a restrictive and differentiated way. For using the GPU for the pattern matching, the task had to be coded to pretend like graphics.

L.Marziale, et al. used the SIMD GPU for accelerating the binary string searches [11]. In their work, they used large number of threads which searched for patterns on input data independently. They used the brute force algorithm to prevent thread divergence and even so the GPU performance was better than multi core CPU implementation employing the Boyer-Moore algorithm.

G.Vasiliadis, et al. illustrated Snort based prototype system “Gnort” using the GPU for performing signature matching operations [12]. They used the Aho-Corasick multi pattern matching algorithm which involved the creation of the DFA state table as a 2-dimensional array for the signatures and transferring them to the GPU. Each thread then processed each packet in isolation. Use of texture memory was employed since the accesses are cached. The implementation outperformed the traditional Snort by a factor of 2.

3 Graphics Processing Units and GPGPU Programming Model

GPUs were originally designed to accelerate graphics tasks like image rendering which uses mostly floating point arithmetic thereby making them useful for high performance computing. This is the basic fundamental of General purpose computing on Graphics Processing Units (GPGPU). Graphic processors have now evolved into a highly parallel, multithreaded, many-core processor with tremendous computational horsepower and very high memory bandwidth.

The Graphics card used in this work is the Nvidia GTS 250 which has the GeForce 9 Series (G9x) architecture. This architecture offers a rich programming environment and flexible abstraction models through the Compute Unified Device Architecture (CUDA) SDK [19]. The CUDA programming model extends the C programming language with directives and libraries that abstract the underlying GPU architecture and make it more suitable for general purpose computing. CUDA also offers a highly optimized data transfer operations to and from the GPU [9].

Programming through CUDA, the GPU can be seen as a device capable of executing a very high number of threads in parallel. A kernel of code can be launched on different threads and on different blocks of threads, called grid. Threads from the same block share data through a fast shared on-chip shared memory and can be synchronized through apposite synchronization points. In CUDA the programmer has access to the device's DRAM and on-chip memory through a number of memory spaces. The choice of the memory to use depends on different factors as speed, amount of memory needed and operations to do on stored data.

4 Regular Expression Pattern Matching Using Finite Automata

Regular expressions being more flexible can be used to describe a large number of signatures and have been increasingly being used to represent attack patterns for virus scanners and network intrusion detection systems. Regular expressions can be matched efficiently on input data by compiling the expressions into state machines either deterministic (DFA) [13] or non-deterministic (NFA) [14].

An NFA can represent multiple signatures with lesser number of states but may result to long matching times, due to the presence of multiple paths. However a DFA is very efficient in terms of speed although a larger number of states may be required. Since in DFA on any state and a particular input the automaton will proceed to at most one state, a sequence of n bytes can be matched in n operations.

A major concern when converting regular expressions into DFAs is the state-space explosion that may occur during compilation. A theoretical worst case study shows that a single regular expression of length n can be expressed as a DFA of up to O(m^n) states, where m is the size of the alphabet, 128 for the extended ASCII character set.

To obtain the equivalent DFA for a regular expression we have to first obtain an NFA corresponding to the expression and then convert the NFA to its equivalent DFA. Regular expressions and NFAs turn out to be exactly equivalent in power. There are multiple ways to translate regular expressions into NFAs but we will be using Thompson's algorithm [15].

To obtain the equivalent DFA corresponding to an NFA we use here a modification of the Subset construction method. By using Thompson's algorithm on an expression like "ababc" we get an NFA as given below.

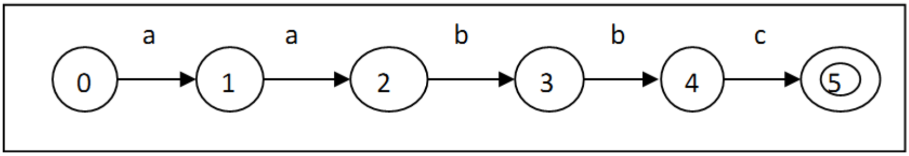


Fig. 1. NFA for regular expression "ababc" given by Thompson's algorithm

The automaton gets any input other than the symbol shown then it goes into the reject state. If on state 1 if we obtain another 'a' then the automaton should remain in state 1 waiting for a 'b'. But the automaton given by Thompson's algorithm would reject the input string. So we need a transition from state 1 to state 1 when the input is 'a'. So the required finite state automaton should be as follows:

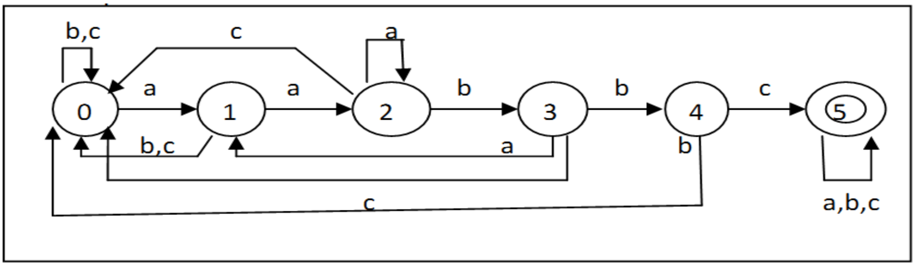


Fig. 2. Required automaton for regular expression "ababc"

To implement this, we create a DFA which has these transitions by making use of a modification to the subset construction algorithm.

Our Subset construction algorithm uses 2 functions:

- Epsilon Closure: This function takes as a parameter, a set of states T and returns again a set of states containing all those states, which can be reached from each individual state of the given set T on Epsilon transition.
- Move: Move takes a set of states T and input character 'a' and returns all the states that can be reached on given input character form all states in T.

Now using these 2 functions, we can perform the transformation:

1. The start state of DFA is created by taking the Epsilon closure of the start state of the NFA.
2. For each new DFA state, perform the following for each input character:
    a. Perform move using the set of states corresponding to the DFA state.
    b. Add epsilon closure of the start state to the result of the move.
    c. Create new state by taking the Epsilon closure of the result.
3. For each newly created state, perform step 2.
4. Accepting states of DFA are all those states, which contain at least one of the accepting states from NFA.

Applying the modified algorithm on the NFA as shown in fig 1, we get the DFA state table as shown below, which corresponds to the DFA in fig 2.

Table 1. DFA sate table for regular expression "ababc" given by our modified subset construction algorithm

State input	{0}	{0,1}	{0,1,2}	{0,3}	{0,4}	final
a	{0,1}	{0,1,2}	{0,1}	{0,1}	{0,1}	final
b	{0}	{0}	{0,3}	{0,4}	{0}	final
c	{0}	{0}	{0}	{0}	final	final
d	{0}	{0}	{0}	{0}	{0}	final

5 Regular Expression Pattern Matching on GPU

The model for the pattern matching module is shown in Fig.3. The input data has to be first transferred to GPU where the actual matching takes place. The input data is then scanned for patterns in parallel by a large number of threads taking the advantage of the massive parallelism in the GPU. Batching many small transfers into large one is beneficial rather than transferring each portion separately [9]. Thus we copy the input data in batches. We make use of 2 buffers and when one is filled up it is transferred to the GPU where it is processed and in parallel the other buffer is filled up. Once the 2nd buffer is filled up, the result is retrieved from the GPU and the transfer for the second buffer is done and the whole procedure is repeated.

Since DFAs are far more efficient in terms of speed we make use of DFAs for our regular expression matching engine. However state space explosion may be caused when converting NFAs to DFAs [16] To prevent greedy memory consumption caused by some regular expressions, we will be using a hybrid approach and convert only the regular expressions that do not exceed 5000 states. The remaining expressions can be processed by the CPU using an NFA schema. The value of 5000 has been arrived since having a cap of 5000 states lets us cover 97% of all the rules in the Snort default rule set [9].

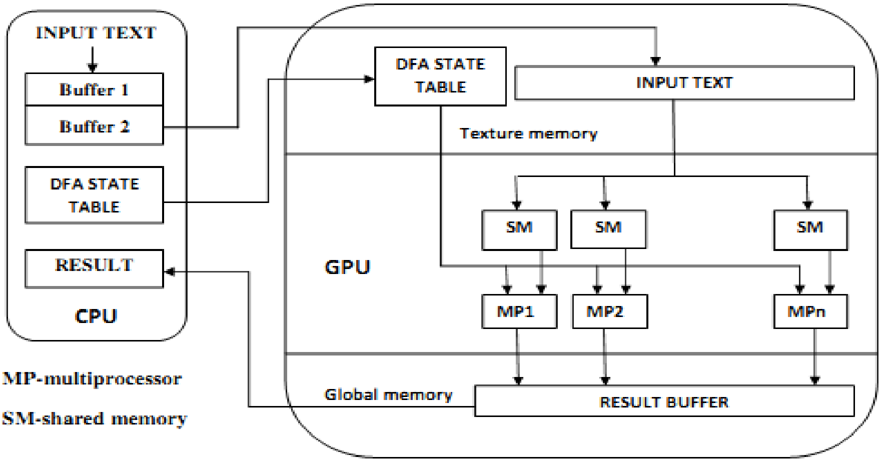


Fig. 3. Model for our pattern matching module

Once the DFA state table is obtained for the regular expression we transfer it to the GPU. The compilation of the DFA state table which involves the creation of the NFA for the regular expression using the Thompson's algorithm and then the creation of the DFA from the NFA using our modified Subset construction algorithm is done on the CPU at startup.

The DFA state table is accessed in a random manner and hence the DFA state table is saved as texture in the GPU. It gives a large improvement in performance when compared to global memory in which accesses need to be coalesced [17]. This is because the texture data accesses are cached which gives a large increase in performance when locality is preserved. Since no dynamic binding is supported for texture memory a large amount of linear memory is statically bound to the texture reference.

The input text is transferred from the main memory to the global memory in the GPU in batches. Each thread processes 256 bytes of the input text. So the buffer size should be (No of blocks* no of threads per block * 256) bytes. Now first thread will scan the first 256 bytes second thread the next 256 bytes and so on. On the GPU threads are executed on a multiprocessor in warps of 32 threads. Each of the thread in a warp executes the same instruction in a clock cycle. If there is thread divergence then the execution is serialized. Each thread in the warp will first fetch from the global memory a byte from the input text. But this would require 32 transactions since the accesses are for 32 different locations with each access of the global memory requiring around 800 clock cycles.

To improve performance on the accesses we have to coalesce the accesses to global memory. If all the threads in a warp access sequential addresses on 32 byte boundaries, only 1 transaction is needed to access the entire 32 bytes. To make use of this fact we will use the shared memory as a cache to store a part of the input data for each thread. The first 64 threads in a block will transfer the first 64 bytes of the data corresponding to the first thread in the block with each thread accessing one byte(the accesses in this case are coalesced) to the shared memory. This will continue with the data belonging to all threads in the block. Now each thread will access its own data from the shared memory which takes very few number of clock cycles. The same procedure is repeated on the next 64 bytes until the entire 256 bytes are processed for each thread.

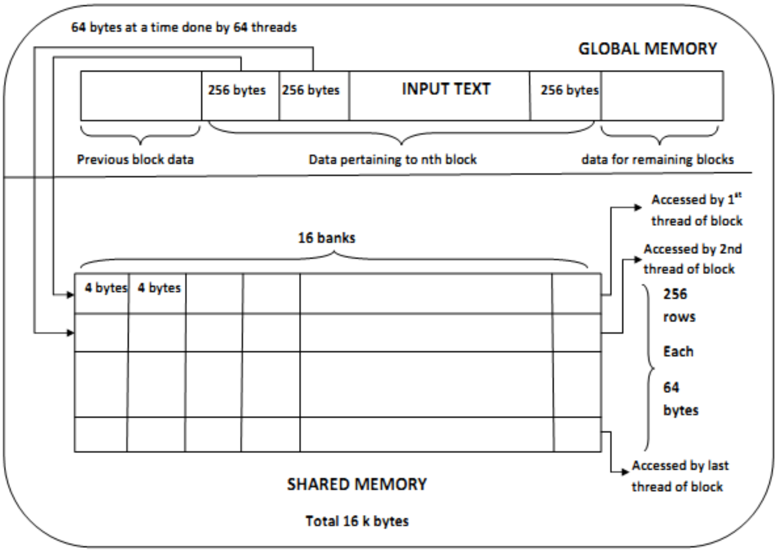

64 bytes at a time done by 64 threads
GLOBAL MEMORY
256 bytes
256 bytes
INPUT TEXT
256 bytes
Previous block data
Data pertaining to nth block
data for remaining blocks
Accessed by 1st thread of block
16 banks
Accessed by 2nd thread of block
4 bytes
4 bytes
256 rows Each 64 bytes
Accessed by last thread of block
SHARED MEMORY
Total 16 k bytes

Fig. 4. Access of the input text using shared memory

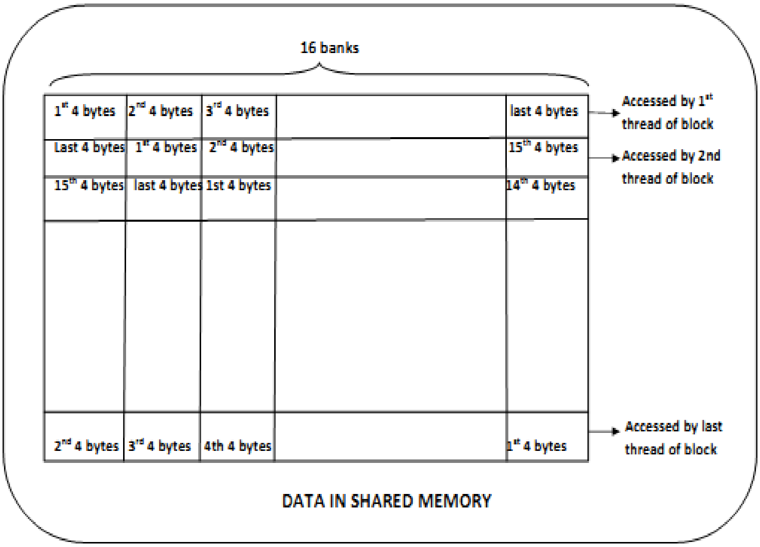


Fig. 5. Storage of part of the input text in shared memory

Only 16 threads can access the shared memory at a time with 1 thread accessing one bank at a time. If 2 threads in a half warp access the memory bank, bank conflicts occur. To prevent this we let the first thread in a block store the data starting from the first bank. The second thread will start storing the data from the second bank in a circular fashion with the last 4 bytes written in first bank. This continues for all the threads in the block. When threads in a half warp access from the shared memory there will be no bank conflicts since all of them will access data from different banks to obtain their corresponding inputs.

6 Experiments and Results

The comparisons of the texture, shared memory kernels and CPU implementation are shown below in fFig.3. For comparison we make use of cudaEvents. Texture memory build is the one proposed by Vasiliadis et al [9] whereas the shared memory approach is the one proposed by us. The performance comparisons in ms for different input sizes are given below in figure 3.

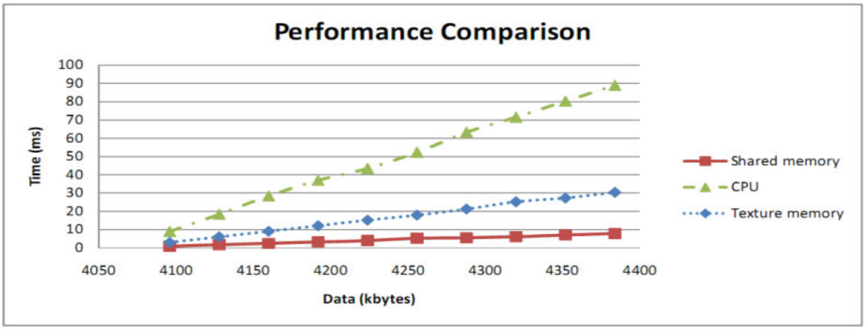


Fig. 6. Performance comparison between texture memory build, shared memory build and the CPU implementation

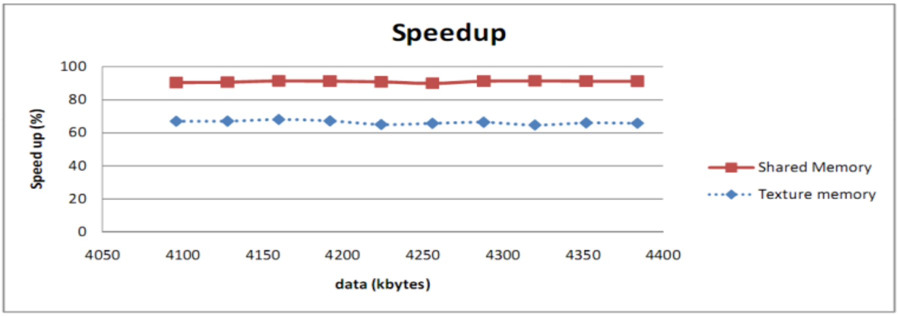


Fig. 7. Speedup of the texture and shared memory build over the CPU implementation

Looking at Fig. 6 we can see that the shared memory build is 4 times faster than the texture memory build being about 12 times faster than the CPU implementation for 1024 x 128 threads with each thread processing 256 bytes. We can also observe that as the no of threads increases the performance also improves and the best improvement over the CPU implementation is observed with 1024 X 128 threads.

The speedup of the shared memory build and the texture memory build is as shown in Fig 7.We get a speedup of about 90% with the shared memory build whereas the texture memory build gives a speedup of about 65% compared to the CPU implementation.

7 Conclusion and Future Work

In this paper we provided the design and implementation of a GPU based pattern matching solution. We proposed that using the shared memory as a temporary cache while making the accesses to the global memory coalesce has a large performance gain over the naïve implementation using the global memory or the implementation using the texture memory. Making use of the shared memory approach we got a speedup of 12x when compared with the CPU implementation and this approach is faster than the texture memory approaches implemented on the GPU by a factor of 4.We also observe that as the input sizes increase the performance advantages over the CPU implementation increase and the best speedup is obtained for 1024 X 128 threads.

For the future work we plan to run the implementation on multiple graphics cards since modern motherboards support up to 4 graphics cards in the same system to get a much better improvement in throughput.

References

1. Snort, `http://www.snort.org`
2. Paxson, V.: Bro: A system for detecting network intruders in real-time. In: Proceedings of the 7th conference on USENIX Security Symposium (SSYM 1998), pp. 3–3. USENIX Association, Berkeley (1998)
3. Sommer, R., Paxson, V.: Enhancing byte-level network intrusion detection signatures with context. In: Proceedings of the 10th ACM conference on Computer and communications security (CCS 2003), pp. 262–271. ACM Press, New York (2003)
4. Antonatos, S., Anagnostakis, K.G., Markatos, E.P.: Generating realistic workloads for network intrusion detection systems. ACM SIGSOFT Software Engineering Notes 29, 207–215 (2004)
5. Tuck, N., Sherwood, T., Calder, B., Varghese, G.: Deterministic memory-efficient string matching algorithms for intrusion detection. In: Proc. of INFOCOM, vol. 4, pp. 2628–2639 (2004)
6. Clark, C.R., Schimmel, D.E.: Scalable Pattern Matching for High Speed Networks. In: Proceedings of the 12th Annual IEEE Symposium on Field-Programmable Custom Computing Machines, pp. 249–257. IEEE Computer Society, Washington (2004)

7. Tan, L., Sherwood, T.: A High Throughput String Matching Architecture for Intrusion Detection and Prevention. In: Proceedings of the 32nd annual international symposium on Computer Architecture, vol. 4, pp. 112–122 (2005)
8. Yu, F., Katz, R.H., Lakshman, T.V.: Gigabit rate packet pattern-matching using TCAM. In: Proceedings of the 12th IEEE International Conference on Network Protocols, pp. 174–183 (2004)
9. Vasiliadis, G., Antonatos, S., Polychronakis, M., Markatos, E.P., Ioannidis, S.: Regular Expression Matching on Graphics Hardware for Intrusion Detection. In: Proceedings of the 12th International Symposium on Recent Advances in Intrusion Detection (RAID), pp. 265 – 283 (2009)
10. Jacob, N., Brodley, C.: Offloading IDS computation to the GPU. In: Proceedings of the 22nd Annual Computer Security Applications Conference on Annual Computer Security Applications Conference (ACSAC 2006), pp. 371–380. IEEE Computer Society, Los Alamitos (2006)
11. Marziale, L., Richard, G.G., Roussev, V.: Massive threading: Using GPUs to increase the performance of digital forensics tools. Digital Investigation 4, 73–81 (2007)
12. Vasiliadis, G., Antonatos, S., Polychronakis, M., Markatos, E.P., Ioannidis, S.: Gnort: High performance network intrusion detection using graphics processors. In: Lippmann, R., Kirda, E., Trachtenberg, A. (eds.) RAID 2008. LNCS, vol. 5230, pp. 116–134. Springer, Heidelberg (2008)
13. Hopcroft, J.E., Motwani, R., Ullman, J.D.: Finite Automata in Introduction to Automata Theory Languages and Computation, pp. 46–47. Pearson Eduaction, Delhi
14. Hopcroft, J.E., Motwani, R., Ullman, J.D.: Finite Automata in Introduction to Automata Theory Languages and Computation, pp. 56–57. Pearson Eduaction, Delhi
15. Thompson, K.: Programming techniques: Regular expression search algorithm. Commun. ACM, 419–422 (1968)
16. Berry, G., Sethi, R.: From regular expressions to deterministic automata. Theor.Comput. Sci. 48(1), 117–126 (1986)
17. NVIDIA CUDA Compute Unified Device Architecture Programming Guide, version 3.1 3.1.pdf, http://developer.download.nvidia.com/compute/cuda/3_1/NVIDIA_CUDA_Programming_Guide_

A New Blend of DE and PSO Algorithms for Global Optimization Problems

Pravesh Kumar Tomar and Millie Pant

Department of Paper Technology Indian Institute of Technology
Roorkee, India
praveshtomariitr@gmail.com,
milliefpt@iitr.ernet.in

Abstract. Differential Evolution (DE) and Particle Swarm Optimization (PSO) algorithms have gained a lot of popularity in the last few years for solving complex optimization problems. Several variants of both the algorithms are available in literature. One such variation is combining the two algorithms in a manner so as to develop an algorithm having positive features of both the algorithms. In the present study we propose a hybrid of DE and PSO algorithm called Mixed Particle Swarm Differential Evolution Algorithm (MPDE) for solving global optimization algorithms. The numerical and statistical results evaluated on a set of benchmark functions show the competence of the proposed algorithm. Further, the proposed algorithm is applied to a practical problem of determining the location of the earthquakes in the Northern Himalayan and Hindu Kush regions of India.

Keywords: Differential evolution, Particle swarm optimization, Hybridization, Global optimization.

1 Introduction

Differential evolution (DE) [1], [2], [3] and Particle swarm optimization (PSO) [4], [5], [6] are two simple and efficient, stochastic, population set based methods for global optimization over continuous space. DE and PSO both have been applied successfully to a wide range of problem as summarized in [3], [7]. Both the algorithms were developed in the year 1995, though both follow a different analogy. While PSO, proposed by Kennedy and Eberhart [4] is inspired by the social behaviour of birds flocking, fish schooling and so on; DE, a simple evolutionary algorithm like that of Genetic Algorithm (GA) was introduced by Storn and Price [1]. Both the algorithms have three main advantages; finding the true global minimum regardless of the initial parameter values, fast convergence, and use of few control parameters [1], [2]. However, both the algorithms have certain drawbacks associated with them which restrict their performance in certain cases. For example; DE has a good ability of search; however it may get trapped in local minima because there is no use of global information about the search space [12]. On the other hand the very fast convergence of PSO due to the rapid information flow among the particles may lead it to a

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 102–112, 2011.

suboptimal solution. In the present study an attempt is made to develop a novel algorithm called Mixed Particle Swarm Differential Evolution (MPDE), by hybridizing the concepts of DE and PSO so as to overcome the drawbacks of both the algorithms. The proposed MPDE is validated on a set of 8 standard benchmark problems and on a real life problem of determining the location of earthquake in a specific region.

The rest of paper is organized as follows: Section 2 provides a compact overview of DE and PSO. A brief literature survey on hybrid DE and PSO algorithms is given in Section 3. Section 4 presents the proposed MPDE algorithm. In Section 5 benchmark problems and the real life problem of earthquake location are given. Experimental settings are given in Section 6. Results and discussions are reported in Section 7, and finally the conclusions derived from the present study are drawn in Section 8

2 Differential Evolution and Particle Swarm Optimization

2.1 Differential Evolution

DE is a population-based approach to function optimization, in which the main strategy is to generate a new position for an individual by calculating vector differences between the randomly selected members of the population. The outline of the basic DE algorithm may be given as follows:

a. Initialize all the vector population randomly in the given upper and lower bound.

b. Evaluate the fitness *f* of each vector in the population.

c. Apply the operators Mutation, Crossover and Selection of DE as defined:

i. **Mutation***:* For each target vector $X_{i,G}$=($x_{i,1,G+1}$, $x_{i,2,G+1}$....,$x_{i,D,G+1}$), select three distinct vectors $X_{r1,G}$, $X_{r2,G}$ and $X_{r3,G}$ randomly from the current population other than vector $X_{i,G}$. Now generate a new population vector $M_{i,G+1}$=($m_{i,1G+1}$, $m_{i,2,G+1}$.... ,$m_{i,D,G+1}$) (called perturbed vector or mutated vector) as:

$$M_{i,G+1} = X_{r1,G} + F * \left(X_{r2,G} - X_{r3,G}\right) \tag{1}$$

Where $F\in$ *[0, 2]* is called scale factor which is use to control the amplification of the differential variation *($X_{r2,G}$ - $X_{r3,G}$).*

ii. **Crossover***:* Perform crossover operation to create a trial vector $U_{i,G+1}$ =($u_{i,1,G+1}$, $u_{i,2,G+1}$, ..., $u_{i,D,G+1}$) as:

$$u_j = \begin{cases} m_{j,G+1} \; if \; Cr < rand(0,1) \, \forall \, j = k \\ x_{j,G} \qquad\qquad\qquad\qquad else \end{cases} \tag{2}$$

Where *j* = 1, 2, ..., *D* (*D*=dimension of problem), *rand* ∈ [0, 1]; *Cr* is the crossover constant takes values in the range [0, 1] and *k* ∈ 1, 2,..., *D;* is the randomly chosen index.

iii. **Selection***:* Generate new population vector $X_{i,G+1}$ for next generation *G+1* by using equation given below:

$$X_{i,G+1} = \begin{cases} U_{i,G+1} \, if \; f\left(U_{i,G+1}\right) < f(X_{i,G}) \\ X_{i,G} \qquad\qquad\qquad\qquad else \end{cases} \tag{3}$$

d. Go to step ***c*** unless a termination criterion is met.

2.2 Particle Swarm Optimization (PSO)

PSO is a stochastic, population set based nature inspired optimization algorithm. In a PSO system, a swarm of individuals (called particles) fly through the search space. Each particle represents a candidate solution of the optimization problem. The position of a particle is influenced by the best position visited by itself and best position of a particle in its neighbourhood. Suppose that at G generation the position and velocity of the i^{th} particle are represented as $X_{i,G}$=($x_{i,1,G}$, $x_{i,2,G}$..., $x_{i,D,G}$) and $V_{i,G}$=($v_{i,1,G}$, $v_{i,2,G}$..., $v_{i,D,G}$) respectively.

Then the updating rule is as follows:

$$V_{i,G+1} = w * V_{i,G} + c_1 * rand_1(pbest_{i,G} - X_{i,G}) + c_2 * rand_2(gbest_{i,G} - X_{i,G}) \quad (4)$$

$$X_{i,G+1} = X_{i,G} + V_{i,G+1} \quad (5)$$

Here w denotes the inertia weight that is used to control the particle velocity. c_1 and c_2 are two positive numbers called acceleration constant and are usually set to 2.05, $pbest_{i,,G}$ and $gbest_{i,,G}$ are personal and global best position of i^{th} particle at generation G; $rand_1$ and $rand_2$ are two uniform random numbers between 0 and 1.

3 Related Work

A number of variations of both PSO and DE have been developed in the past decade to improve the performance of these algorithms. One class of variation include hybridization of DE and PSO [13], in which the two algorithms are merged together to form a new algorithm. Some of the work done in the field of hybrid DE and PSO algorithms is as follows: in 2001, Hendtlass [8] used the DE perturbation approach to adapt particle position. The DE reproduction process is applied to the particle in PSO swarm at specified intervals. In the specified interval PSO particles serve as population for DE and DE is executed for a number of generations. In 2003, Wen, et al [9] proposed a new hybrid variant of PSO and DE term as DEPSO, which provide the bell-shaped mutations with consensus on the population diversity along with the evolution, while keep the self organized particle swarm dynamics. In 2004, Kannan et al.[10] extended the work of Hendtlass [8] by applying DE to each particle for number of iteration, and replaced the particle with the best individual obtained from the DE process. In 2004 Talbi and Batouche [11] provided mutation by using DE operator. In 2007 Zhi-Feng Hao et al. [12] proposed a hybridization of PSO and DE that combines the differential information obtained by DE with the memory extracted by PSO to create promising solution. A hybrid version of DE and PSO was suggested by Pant et al. [15] where the two algorithms are applied alternatively. In 2009, Mahamed G.H. Omran et al. [13] presented barebones differential evolution, which combines the concepts of barebones PSO and recombination operator of DE. Later in 2009, Changsheng Zhang et al. [14] proposed a novel hybridization of PSO and DE called DE-PSO for unconstrained problem by updating particles not only by DE operator but also using mechanism of PSO.

4 MPDE: Mixed Particle Swarm Differential Evolution

The proposed MPDE is a judicious blend of DE and PSO algorithms. It starts as the global best version of PSO for updating the velocity and position of the particles. Mutation and crossover operations of DE are then applied after the position update. Before generating the next population we select the best candidate generated by PSO operations and the DE operations. After selection of the best position, we apply the usual selection process of DE. By doing this we merge the global information obtained by PSO into DE with the hope of maintaining a better balance between the exploration and exploitation factors. The main step in proposed MPDE are given below;

1. Initialize the population randomly
2. Find the velocity $V_{i,G+1}$ of any particle $X_{i,G}$ by using the following equation

$$V_{i,G+1}=w*V_{i,G} + c*rand*(gbest_{i,,G} - X_{i,G}) \tag{6}$$

3. Find new position $X'_{i,G+1}$ of $X_{i,G}$ as;

$$X'_{i,G+1} =X_{i,G} +V_{i,G+1} \tag{7}$$

4. Perform Mutation and crossover according to equ-1 and equ-2.
5. Select the fittest position between trail vector $U_{i,G+1}$ and $X'_{i,G+1}$
6. Generate the new population for next generation by taking the fittest particle from step 5 and target vector $X_{i,G}$.

4.1 Pseudo Code

Let P be population of size N_P and let $X_{i,G}$ be any individual of dimension D in population P and let $V_{i,G}$ be the position of each individual $X_{i,G}$ at any generation G.

```
Begin
Generate Uniformly Distribution random population $P=\{X_{1,G}, X_{2,G}, ..., X_{NP,G}\}$.
for each i ∈ (1, 2, ..., NP)
    $X_{i,G} = lower(X_{i,G})+rand[0,1]*(upper(X_{i,G})-lower(X_{i,G}))$
End for each.
Evaluate P.
While ($FV > VTR$ and $NFE<Max_{NFE}$) /* VTR=Value to Reach, FV= Fitness Value, NFE=Number of Function Evaluation */
forAll $i \le NP$
$V_{i,G+1} = w* V_{i,G} + c* rand[0,1]*(gbest_{i,G} - X_{i,G})$
$X'_{i,G+1}= V_{i,G} + X_{i,G}$
Select three random parents $X_{r1}$, $X_{r2}$, and $X_{r3}$, from the population where $i \neq r1 \neq r2 \neq r3$
$M_{i,G+1}= X_{r1,G} + F*(X_{r2,G} - X_{r3,G})$
Randomly select $j_{rand}$ from $j=(1,2,...,D)$
```

```
        forAll $j \le D$
            if $Cr<rand[0,1]$ or $j=j_{rand}$;
                $u_{j,G+1}=m_{j,G+1}$
            else
                $u_{j,G+1}=x'_{j,G+1}$
            End if
        End forAll
        Evaluate $U_{i,G+1}$
        if $f(U_{i,G+1})<f(X'_{i,G+1})$
            $S_{i,G+1}= U_{i,G+1}$
        else
            $S_{i,G+1}= X'_{i,G+1}$
        End if
        If $f(S_{i,G+1})<f(X_{i,G})$
            $Y_{i,G+1}=S_{i,G+1}$
        else
            $Y_{i,G+1}= X_{i,G}$
        End if
    End forAll
    forAll $i \le NP$
        $X_{i,G+1}=Y_{i,G+1}$;
    End forAll
End While
```

5 Application of MPDE

5.1 Application to Benchmark Problems

8 common benchmark functions with bound constraints are used for experiments. These problems are selected from [3], which contains many more problems of the same type. This test bed though narrow forms a launch pad for the validity of an optimization algorithm.

5.2 Practical Problem: Determination of the Earthquake Location

Locating earthquakes is one of the oldest and perhaps the most active problem in seismology. It can be modelled as a non linear optimization problem where the objective is to minimize the discrepancy between the observed and the calculated seismic travel time. The problem is complicated by the nonlinear dependence of seismic travel times on location, incomplete knowledge of three dimensional velocity structures along the source receiver path and difficulties associated with the inadequate station coverage and outliers in the observed travel time picks.

Conventionally, the mathematical model may be described in two parts; the *forward problem model* and the *inversion problem model*. The forward model, also called travel time model gives the travel time of the compressional wave in the different layers of the earth crust calculated for different depths. The travel time of seismic waves in each layer to get the total travel time for the waves is then added to reach at the observational stations on the surface of the earth from focus. Let the parameters of the preliminary hypocentre be (x_i, y_i, z_i), representing the coordinate values of its latitude, longitude and depth. Let (x_j, y_j) be the latitude and longitude of the stations and let v_l represent the average crustal velocity of the compressional waves in the l^{th} layer of the earth. The theoretical travel time and epicentral distances according to Xing et al. [18], are given as

$$t_{ij} = \frac{\sqrt{\Delta_{ij}^2+Z_i^2}}{v_l} \tag{8}$$

Where $\Delta_{ij}= 111.199\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 cos^2 \frac{(x_i-x_j)}{2}}$

In *inverse problem model* the values of hypocentral parameters are calculated inversely by minimizing a root mean square function of the calculated (C_k) and the observed (O_k) travel times. The objective function is then given as:

$$Minimize\ f = \{\textstyle\sum_{i=1}^{n}(C_k - O_k)^2\} \tag{9}$$

For a single earthquake C_k is:

$$C_k = \frac{\sqrt{\Delta_k^2+Z_k^2}}{v_l} \tag{10}$$

Where $\Delta_k= 111.199\sqrt{(x - x_i)^2 + (y - y_i)^2 cos^2 \frac{(x-x_i)}{2}}$

Here (x, y, z) represents the hypocentre of the earthquake.

This model is validated on real data from North Western (NW) Himalayan region, subject to the following restrictions:

Lower Latitude< x < upper Latitude; Lower Longitude < y < upper Longitude;

Lower Depth < z < upper Depth.

For the hypocenters in the NW Himalayan and Hindukush region, these restriction limits are taken as: $25^0 < x < 35^0$; $70^0 < y < 90^0$; $0 < z < 100$(Kms)

6 Experimental Settings

6.1 Experimental Setting for Benchmark Problems

MPDE is implemented in Dev-C++ and the experiments are conducted on a computer with 2.00 GHz Intel (R) core (TM) 2 duo CPU and 2- GB of RAM. The parameters used are given in Table-1.

Over all acceleration rate AR, used for the purpose of comparison is taken from [3]. In every case, a run was terminated when the best function value obtained is less than a threshold for the given function or when the maximum number of function evaluation (NFE=10^6) was reached. To find the statistical features of the best solutions obtained by the proposed algorithm, we have done the approximate two-sample t-tests [16], [17] between the classical DE and the proposed MPDE.

Table 1. Parameter Tuning

Parameter	Setting value
Pop size (*NP*)	100
Dimension (*D*)	30, 50
Scale Factor (*F*)	0.5
Crossover rate (*Cr*)	0.5
Max velocity (V_{max})	0.1*upper bound
Inertia weight (*w*)	0.5
Value to reach (*VTR*)	10^{-06}

6.2 Experimental Setting for Earthquake Problems

For earthquake problem, dimension (*D*) of the search space is 3. The decision variables are latitude, longitude and the depth of the hypocentre. All other parameter settings for earthquake problem are kept same as that of benchmark functions. The real data of earthquake is taken from an observation of Hindu Kush and NW Himalayan region where an earthquake of intensity 5.0 on Richter scale was recorded on September 25, 2008. Its observed location was 28.889°, 85.013° and depth of 111.6 Km. The location of hypocenter and 10 different stations is given in Table-2.

Table 2. Location of earthquake and the locations of observations stations

Hypocenter	Stations	
28.889°	30.00°	70.00°
85.013°	30.14°	79.20°
111.6 km	30.97°	77.86°
	30.71°	77.25°
	31.10°	79.61°
	30.54°	78.10°
	29.71°	78.43°
	30.49°	77.58°
	30.53°	77.73°
	30.33°	78.74°

7 Result and Discussion

We compare the convergence speed of DE and MPDE by measuring the *number of function Evolution* (NFE). A smaller NFE indicates higher convergence speed. The termination criterion is to find a value smaller than the value-to-reach (VTR=10^{-06}) before reaching the maximum NFE (=10^{6}). In order to minimize the effect of the stochastic nature of the algorithms, every result is taken as average of 30 different runs.

7.1 Result of Benchmark Problems

In Table-3, comparison of average NFE and acceleration rate (*AR*) of each algorithm for 10 benchmarks function is given. Table-4 provides average mean fitness values, standard deviation and t-value by each algorithm for dimension 30 and 50.

Table 3. Average number of function evaluation (NFE) and Acceleration Rate (AR)

Fun	Dim.	DE	MPDE	AR (%)
Ackley	30	98020	**83770**	14.53 %
	50	203180	**174940**	13.90 %
Axis	30	58300	**44590**	23.52 %
	50	127430	**101010**	20.73 %
Griewenk	30	74550	**63180**	15.25 %
	50	149250	**122710**	17.78 %
Schawefel-1	30	90850	**77270**	14.95 %
	50	195410	**165620**	15.24 %
Sphere	30	51060	**35890**	29.71 %
	50	109210	**80820**	26.00 %
Zakharov	30	73140	**59840**	18.18 %
	50	168800	**144640**	14.31 %
Step	30	27400	**10770**	60.69 %
	50	59650	**25500**	57.25 %
Schawefel-2	30	405870	**329350**	18.85 %
	50	1014675	**823350**	18.86 %
Total	D=30	879190	**704660**	**19.85 %**
	D=50	2027605	**1638590**	**19.18 %**

Table 4. Average mean fitness, standard deviation and t-value of functions in 30 runs for *D*=30/ 50

Fun	**D=30**				
	DE		MPDE		
	Mean	S.D.	Mean	S.D	t-value
Ackley	9.648e-006	2.691e-007	9.466e-006	7.188e-007	0.707
Axis	9.041e-006	5.157e-007	8.917e-006	1.232e-006	0.277
Griewenk	9.021e-006	6.528e-007	9.039e-006	7.325e-007	0.054
Schawefel-1	9.356e-006	3.621e-007	9.452e-006	4.329e-007	0.513
Sphere	9.051e-006	4.681e-007	8.713e-006	1.221e-006	0.774
Zakharov	9.067e-006	4.629e-007	8.986e-006	8.901e-007	0.241
Step	0	0	0	0	--
Schawefel-2	9.442e-006	4.661e-007	9.489e-006	2.956e-007	0.0448
	D=50				
Ackley	9.767e-006	1.664e-007	9.753e-006	2.101e-007	0.156
Axis	9.376e-006	5.926e-007	9.493e-006	3.534e-007	0.507
Griewenk	9.359e-006	4.375e-007	9.358e-006	3.489e-007	0.004
Schawefel-1	9.605e-006	3.403e-007	9.729e-006	3.251e-007	0.788
Sphere	9.319e-006	5.555e-007	9.178e-006	5.471e-007	0.539
Zakharov	9.093e-006	6.082e-007	9.425e-006	5.391e-007	1.227
Step	0	0	0	0	--
Schawefel-2	9.531e-006	4.761e-007	9.432e-006	4.861e-007	0.783

From Table-3, we can clearly see that the NFE, of proposed MPDE is less than classical DE for dimensions 30 and 50. For solving 8 problems the average NFE taken by MPDE are 704660 and 1638590 for dimensions 30 and 50 respectively. This implies that acceleration rate for MPDE in comparison to DE is 19.85 % and 19.18 %for dimensions 30 and 50 respectively. Also, from numerical results given in Table 4, we can see that the performance of proposed MPDE is better than the simple DE. Performance graphs of selected problems illustrated in Fig.1 and Fig. 2 also indicate the faster convergence of MPDE.

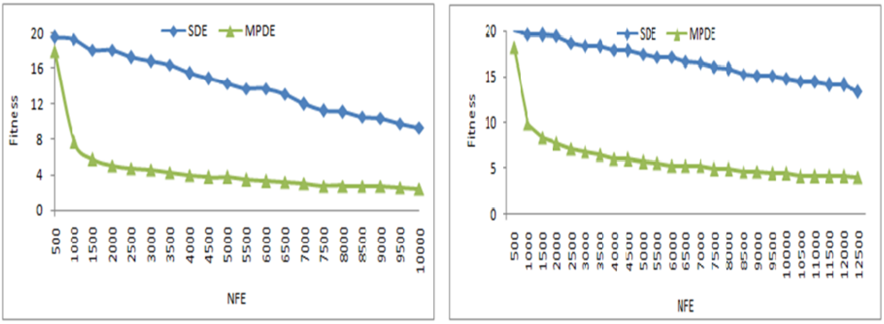


Fig. 1. Performance graph of Ackley function for D=30 and 50

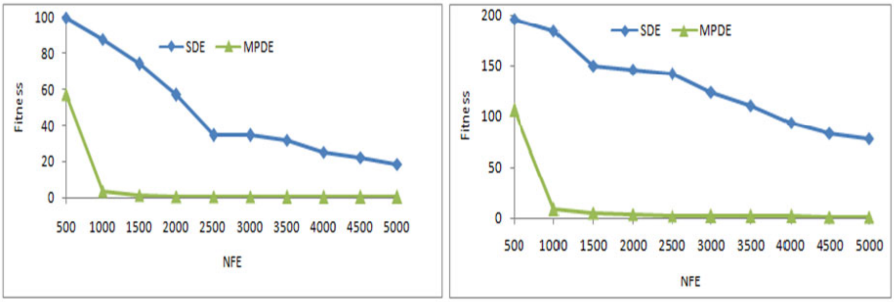


Fig. 2. Performance graph of Sphere function for D=30 and 50

7.2 Result of Earthquake Problem

According to Sushil Kumar and Sato [19], the velocity of the compressional waves within the depth 0- 15 Km is 5.2 Km/sec and 15-40 Km is 5.89 Km/sec. According to Kalia et al. [20] the velocity of compressional wave within the depth 40-70 Km is 8.14 Km/sec, 70-85 Km is 8.32 Km/sec and 85-100 is 8.29 Km/sec. On the basis of data given in Table-2, we calculated the travel time in each layer of the earth at different depths to the ten different stations by using equation (10). Table-5 gives the detailed calculation of travel times different layers. Using the travel time data we minimize the function f.

Table 5. Travel time of compressional waves at different depths of the earth crust

Travel time (Sec) in different depths (kms)					
0-15	15-40	40-75	75-85	85-100	Total
3.70972	4.71768	3.97535	2.31858	2.32697	17.04830
3.81867	4.78496	4.01723	2.38667	2.39531	17.40284
3.79631	4.77107	4.00857	2.37269	2.38128	17.32992
3.79271	4.76884	4.00718	2.37044	2.37902	17.31819
3.78329	4.76300	4.00354	2.36456	2.37311	17.28750
3.78739	4.76554	4.00512	2.36712	2.37568	17.30085
3.80601	4.77709	4.01232	2.37876	2.38736	17.36154
3.81897	4.78515	4.01734	2.38686	2.39550	17.40382
3.80686	4.77762	4.01265	2.37929	2.38790	17.36432

From Table-6, we can observe that both DE and MPDE are competent in determining the location as the values obtained by the algorithms matches the theoretical value. However, in terms of NFE MPDE is much faster than DE, giving an AR of more than 35% in all the cases.

Table 6. Location of hypocenter from 10 stations, mean fitness value and NFE in 30 runs

		DE					MPDE			
x^o	y^o	z(Km)	f	NFE	x^o	y^o	z(Km)	f	NFE	AR(%)
27.85	84.50	111.64	8.97e-006	8500	27.56	85.53	111.10	4.74e-006	5200	38.82%
28.49	83.71	111.67	4.53e-006	8600	29.83	84.00	111.67	2.01e-006	5100	40.69%
29.47	83.84	111.77	3.45e-006	8800	27.63	85.30	111.29	8.37e-006	5100	42.04%
27.29	84.88	111.17	5.25e-006	8600	28.47	85.93	111.17	2.31e-006	5500	36.04%
29.43	85.89	111.43	3.93e-006	7900	28.59	82.60	111.89	2.10e-006	5000	36.70%
29.20	85.01	111.44	7.43e-007	9400	29.57	84.42	111.59	1.13e-006	5400	42.55%
27.70	84.12	111.70	1.53e-005	8400	28.82	84.36	111.71	1.03e-005	5300	36.90%
29.16	85.58	111.30	4.09e-005	8200	28.34	83.60	111.73	6.29e-006	5400	34.14%
29.39	85.05	111.46	4.91e-005	8300	28.42	85.01	111.53	1.63e-005	5000	39.75%
28.77	84.16	111.65	5.68e-005	8600	28.67	83.29	111.79	9.34e-007	5200	39.53%

8 Conclusions

In the present study, we proposed a modified hybrid version of DE and PSO algorithm namely MPDE. In MPDE, the global information sharing mechanism of PSO is blended into the structure of DE. The proposed MPDE is evaluated on a set of 8 benchmark problems and a practical problem of determining the optimal earthquake location. Numerical results and statistical analysis validate the competence of the proposed algorithm. The proposed work can be extended in several directions. In future we plan to test our algorithm on more complex benchmark and real life problems and to compare MPDE with other hybrid DE-PSO versions.

References

1. Storn, R., Price, K.: Differential evolution—A simple and efficient adaptive scheme for global optimization over continuous spaces,Technical report TR-95-012, International Computer Science Insitute (1995)

2. Storn, R., Price, K.: DE-A simple evolution strategy for fast optimization. Dr. Dobb's Journal, 18–24, 78 (April 1997)
3. Ali, M., Pant, M., Abraham, A.: Simplex differential evolution. Acta Polytechnica Hungarica 6, 95–115 (2009)
4. Kennedy, J., Eberhart, R.C.: Particle swarm optimization. In: Proceedings of IEEE International Joint Conference on Neural network, pp. 1942–1948. IEEE Press, Los Alamitos (1995)
5. Pant, M., Thangaraj, R., Abraham, A.: A new PSO algorithm with crossover operator for global optimization problems. In: Corchado, E., et al. (eds.) Second International Symposium on Hybrid Artificial Intelligent Systems (HAIS 2007), Innovations in Hybrid Intelligent Systems. AISC, vol. 44, pp. 215–222. Springer, Germany (2007)
6. Pant, M., Thangaraj, R., Abraham, A.: A new quantum behaved particle swarm optimization. In: Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation, Atlanta, GA, USA, pp. 87–94 (2008) ISBN:978-1-60558-130-9
7. Engelbrecht, A.: Fundamental of computational swarm intelligence. Wiley & sons, Chichester (2005)
8. Hendtlass, T.: A combined swarm differential evolution algorithm for optimization problems. In: Monostori, L., Váncza, J., Ali, M. (eds.) IEA/AIE 2001. LNCS (LNAI), vol. 2070, pp. 11–18. Springer, Heidelberg (2001)
9. Zhang, W.J., Xie, X.F.: DEPSO: Hybrid particle swarm with differential evolution operator. In: IEEE International Conference on Systems, Man and Cybernetics, vol. 4, pp. 3816–3821 (2003)
10. Kannan, S., Slochanal, S., Subbaraj, P., Padhy, N.: Application of particle swarm optimization technique and its variants to generation expansion planning. Electrical Power System Research 70(3), 203–210 (2004)
11. Talbi, H., Batouche, M.: Hybrid particle swarm with differential evolution for multimodal image registration. In: Proceeding of the IEEE International Conference on Industrial Technology, vol. 3, pp. 1567–1573 (2004)
12. Hao, Z.F., Guo, G.H., Huang, H.: A particle swarm optimization algorithm with differential evolution. In: Proceeding of the Sixth International Conference on Machine Learning and Cybernetics, pp. 1031–1035. Hong Kong (August 2007)
13. Omran, M., Engelbrecht, A., Salman, A.: Bare bones differential evolution. European Journal of Operation Research 196, 128–139 (2008)
14. Zhang, C., Ning, J., Lu, S., Ouyang, D., Ding, T.: A novel hybrid differential evolution and particle swarm optimization algorithm for unconstrained optimization. Operation Research Letters 37, 117–122 (2009)
15. Pant, M., Thangraj, R., Grosan, C., Abraham, A.: Hybrid differential evolution – Particle swarm optimization algorithm for solving global optimization problems. In: ICDIM, pp. 18–24 (2008)
16. Zhu, R.: Statistical analysis methods. China Forestry Publishing House, Beijing (1989)
17. Zhang, M., Luo, W., Wang, X.: Differential evolution with dynamic stochastic selection for constrained optimization. Information Science: An International Journal 178, 3043–3074 (2008)
18. Xing, J., Yang, W.-d., Li, S.-y., Ma, Q.: A new seismic location method. Earthquake and Engineering Vibration 27, 20–25 (2007)
19. Kaliaa, K.L., Krishna, V.G., Narain, H.: Upper mantle velocity structurein the Hindu Kush region from travel time studies of deep Earthquakes using a new analytical method. Bull. Seismol. Soc. Am 59, 1949–1967 (1969)
20. Kumar, S., Sato, T.: Compressional & Shear waves velocities in the crust, beneath the Garhwal Himalaya,N-India. Journal of Himalayan Geology 24(2), 77–85 (2003)

Graph Isomorphism Detection Using Vertex Similarity Measure

Venkatesh Bandaru and S. Durga Bhavani

Department of Computer and Information Sciences,
University of Hyderabad, Hyderabad, India
sdbcs@uohyd.ernet.in

Abstract. Measures of vertex similarity have been incorporated in graph matching algorithms. Graph matching tries to retrieve a 1-1 correspondence between vertices of two given graphs. In this paper, the vertex similarity measure of Blondel et al. is studied for its usefulness in detecting graph isomorphism. Firstly, the applicability of this measure to distinguish similar pairs from dissimilar pairs is shown to be limited in scope even for small graphs. In a preliminary experiment, we show that Blondel's vertex similarity measure does not retrieve the isomorphism within a graph of 14 nodes. We propose a refinement of Blondel's measure. Zager et al. also refine Blondel's measure and further propose a graph matching algorithm. We propose a graph matching algorithm based on the lines of Zager et al. and test our algorithm against Zager's as well as Blondel's and show that the proposed refinement performs better than both the measures with regard to graph isomorphism problem. The performance is evaluated systematically on a large bench mark data set made available by Foggia et al. The proposed algorithm performs with 90.10% accuracy on all of the 18,200 pairs of isomorphic graphs available in the benchmark dataset.

Keywords: vertex similarity, graph isomorphism, graph matching, large graphs.

1 Introduction

The literature is abound with discussion on graph isomorphism and subgraph isomorphism problems. On the other hand, there are many situations where one needs to say if two graphs are 'similar' or not. There really may not be an exact matching between the two graphs. For example, in the problem of protein structure comparison, the 2D representations of proteins extracted as contact networks can be compared to see if there is an approximate graph matching or not. In many of such applications the local substructures need to be matched and not graphs as a whole. In this context studying graph similarity becomes important.

The concept of vertex similarity has been used in the context of networks with the idea of two vertices being similar if they share similar neighbourhoods.

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 113–122, 2011.

Leicht et al. [1] propose a vertex similarity measure based on matrix methods. One of the key ideas proposed in this area is by Kleinberg [2] based on whose work Blondel et al. [3] and later Zager et al. [5] propose vertex similarity measures. Literature on graph similarity involves proposing a vertex similarity measure which then is used to compute a matching and then derive an approximate graph matching between the input graphs. Kpodjedo et al. [6] propose vertex similarity measure which is based on perfect edge correspondences and then use it as a greedy criterion for tabu search to build an approximate graph matching.

Kleinberg [2] designs a novel method to assign scores for nodes in a web graph. This method led to many interesting papers in graph similarity literature. Blondel et al. [3] propose a vertex similarity measure based on Kleinberg's idea and apply it to the problem of synonym extraction and web-searching. Later Zager et al. [5] refine Blondel's measure by including edge similarity scores. Further, using the popular hungarian assignment algorithm [8] they propose an approximate graph matching algorithm. The authors test their algorithm on small randomly generated graphs.

In this work we first analyze Blondel's vertex similarity measure and evaluate its performance on self-similarity. It should be noted that the entire work is related to directed graphs though the measures are supposed to work for undirected graphs. We find that Blondel's measure does not work for large graphs and hence propose a refinement on the measure. In the second step we propose a graph matching algorithm based on our measure. One of the important contributions of this paper is that of testing all these algorithms on a huge bench mark data set which contains nearly 72,800 graphs having 18,200 pairs of isomorphic graphs, the largest graph being of size 1024 nodes. The implementation results show 100% accuracy in isomorphism detection in almost all the types of the isomorphic graph data sets which indicates that this measure may be used for the approximate graph matching problem. To the best of our knowledge this kind of testing has not been done earlier.

2 Analysis of Blondel's and Zager's Vertex Similarity Measures

Assume that there are two graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ with X, Y as their respective adjacency matrices.

2.1 Blondel's Similarity Measure

Blondel computes vertex similarity using an iterative approach and defines similarity of two directed graphs G_1 and G_2 of size N as

$$S(k+1) = \frac{YS(k)X^T + Y^T S(k)X}{\|YS(k)X^T + Y^T S(k)X\|}$$

at iteration $k \geq 1$ with $S_{(i,j)}(1) = 1, \text{for all}, i, j, 1 \leq i, j \leq N$. Note that at each step $S(k)$ is being normalized before carrying iteration forward. The normalization is carried out by dividing each of the column vectors in the matrix by the sum of the entries in the column. This score is based on the notion of hub and authority score defined by Kleinberg et al. [2]. Experimentation on vertex similarity scores using Blondel's measure shows that normalization at each step has a significant impact on the actual values that are obtained at the end of iterations. In fact, for large graphs, it is seen that the similarity values obtained become very small and thus not distinguishing the vertices of two graphs in any way. Also this leads to the algorithm not being very useful for matching two graphs as the vertices are indistinguishable.

We present here a preliminary test to evaluate the vertex similarity measure. We calculate vertex similarity measure between all pairs of vertices within a single graph and check if maximum similarity scores are obtained along the diagonal which we call as the self-similarity test. In the Figure 2, we present the results of applying this test on a sample graph of size 14 given in Figure 1.

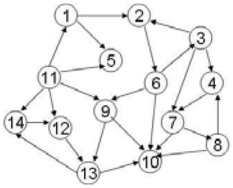
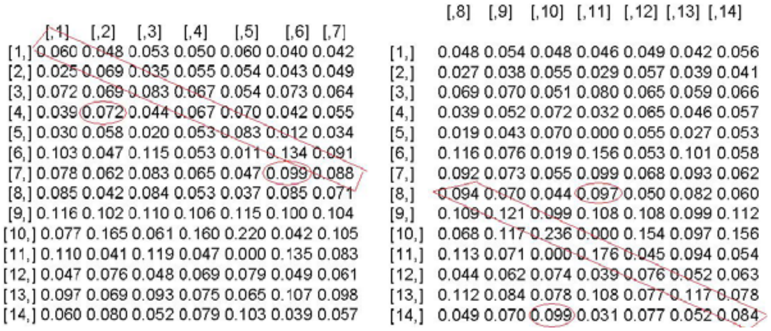


Fig. 1. Directed graph with 14 vertices



	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]	[,9]	[,10]	[,11]	[,12]	[,13]	[,14]
[1,]	0.060	0.048	0.053	0.050	0.060	0.040	0.042	0.048	0.054	0.048	0.046	0.049	0.042	0.056
[2,]	0.025	0.069	0.035	0.055	0.054	0.043	0.049	0.027	0.038	0.055	0.029	0.057	0.039	0.041
[3,]	0.072	0.069	0.083	0.067	0.054	0.073	0.064	0.069	0.070	0.051	0.080	0.065	0.059	0.066
[4,]	0.039	0.072	0.044	0.067	0.070	0.042	0.055	0.039	0.052	0.072	0.032	0.065	0.046	0.057
[5,]	0.030	0.058	0.020	0.053	0.083	0.012	0.034	0.019	0.043	0.070	0.000	0.055	0.027	0.053
[6,]	0.103	0.047	0.115	0.053	0.011	0.134	0.091	0.116	0.076	0.019	0.156	0.053	0.101	0.058
[7,]	0.078	0.062	0.083	0.065	0.047	0.099	0.088	0.092	0.073	0.055	0.099	0.068	0.093	0.062
[8,]	0.085	0.042	0.084	0.053	0.037	0.085	0.071	0.094	0.070	0.044	0.097	0.050	0.082	0.060
[9,]	0.116	0.102	0.110	0.106	0.115	0.100	0.104	0.109	0.121	0.099	0.108	0.108	0.099	0.112
[10,]	0.077	0.165	0.061	0.160	0.220	0.042	0.105	0.068	0.117	0.236	0.000	0.154	0.097	0.156
[11,]	0.110	0.041	0.119	0.047	0.000	0.135	0.083	0.113	0.071	0.000	0.176	0.045	0.094	0.054
[12,]	0.047	0.076	0.048	0.069	0.079	0.049	0.061	0.044	0.062	0.074	0.039	0.076	0.052	0.063
[13,]	0.097	0.069	0.093	0.075	0.065	0.107	0.098	0.112	0.084	0.078	0.108	0.077	0.117	0.078
[14,]	0.060	0.080	0.052	0.079	0.103	0.039	0.057	0.049	0.070	0.099	0.031	0.077	0.052	0.084

Fig. 2. Note the circled maximum entries which are not part of the main diagonal in the 14-node graph

It can be seen that at the positions 4, 7 and 14 the diagonal entries are not maximal but other node matches get high values which are circled in the Figure 2.

2.2 Zager's Similarity Measure

Zager et al. design a vertex similarity measure by incorporating edge similarity scores within Blondel's measure. Given two graphs $G_1 = (V_1, E_1)$ and

$G_2 = (V_2, E_2)$ with X, Y as their respective adjacency matrices. Zager et al. compute vertex similarity and edge similarity using an iterative approach and define similarity of two directed graphs G_1 and G_2 of size N as follows

$$E(k) = \frac{Y_S^T N(k-1) X_S + Y_T^T N(k-1) X_T}{\|Y_S^T N(k-1) X_S + Y_T^T N(k-1) X_T\|}$$

$$N(k) = \frac{Y_S E(k-1) X_S^T + Y_T E(k-1) X_T^T}{\|Y_S E(k-1) X_S^T + Y_T E(k-1) X_T^T\|}$$

$E(k)$, $N(k)$ are edge similarity and vertex similarity scores at iteration k respectively, where k is the number of iterations. X_S, Y_S are source edge matrices and X_T, Y_T are terminus edge matrices, as defined in [5]. The iteration is repeated until scores converge. In the above formulae after calculating denominator at each iteration column normalization is applied.

Problem with Blondel's and Zager's approaches arise from the fact that at each iteration they are normalizing scores. The emerging scores become smaller as the graph size increases, sometimes the scores may be very close to zero which makes it difficult to distinguish if the pair of vertices are similar or not. Hence these measures would not be useful in obtaining similarity or dissimilarity information.

Therefore, a possible solution would be either to remove normalization altogether or to limit the number of runs and not to iterate the measure to convergence.

3 Graph Matching Algorithm

Graph matching tries to retrieve a 1-1 correspondence between vertices of two given graphs. In the context of vertex similarity, this mapping maximizes the sum of similarity scores of the matched vertices. Zager et al. use Hungarian algorithm [8] to retrieve an optimal 1-1 correspondence between vertices of two graphs, with the optimization criterion based on maximizing vertex similarity measure. Hungarian algorithm addresses the assignment problem in which there are n jobs and n machines, where each job has to be assigned to only one machine subject to the objective function that can be either maximized or minimized.

Our aim in this section is to propose a refinement on Blondel's measure and exactly along the lines of Zager et al., apply Hungarian algorithm on the new measure to obtain an optimal graph matching. In order to evaluate the graph matching, we carry out experimentation on a bench mark data set whose details are given below.

3.1 Benchmark Data Set

The Data set [10] used in this experimentation is constructed by Foggia et al. [9]. The database consists of 72,800 graphs of different sizes and different types. It consists of bounded valence graphs, mesh graphs, randomly generated graphs of regular and irregular type [9]. The entire database has pairs of graphs with varying similarity that are 100% similar (isomorphic), 60%, 40%, 20% similar that are useful for performance evaluation of algorithms for isomorphism and subgraph isomorphism problems. The entire testing is done for the isomorphic pairs of graphs which are 18,200 in number available in the data set.

3.2 Propsed Refinement on Blondel's Vertex Similarity Measure

Our experimentation results on Blondel's measure show that applying normalization at every iteration has a negative impact on the final scores emerging in the similarity matrix. We note that without normalization the raw scores themselves emerge as winners in self-similarity test and further turn out to be useful for addressing the graph matching problem. Thus, we refine Blondel's approach by removing normalization altogether and calculate the vertex similarities for all pairs of vertices of two graphs feeding the raw scores themselves to the succeeding iterations. Hence the refined Blondel's vertex similarity measure is defined as follows: Let X and Y be adjacency matrices of two graphs of size N, G_1 and G_2 respectively.

$$S(k+1) = YS(k)X^T + Y^T S(k)X, S_{(i,j)}(0) = (1, 1, \ldots, 1) \forall 1 \leq i, j \leq N.$$

In Blondel's approach the algorithm is left to run till the matrix scores 'converge' whereas we notice that our algorithm need to be run only up to a small number of iterations to get significant distinguishing scores for vertex similarity. The significance of the scores is validated by the graph matching algorithm.

3.3 Graph Matching Algorithm

Aim: Matching graphs, given two graphs of equal size.
Input: X, Y are adjacency matrices of graphs G_1, G_2. If N is the number of vertices of G_1 and G_2 then S is a vertex similarity matrix of size $N \times N$
Otput: Vertex similarity matrix and a matching that gives 1-1 correspondence between vertices of graph G_2 to G_1.

Procedure:

```
$S_{ij}(0) \leftarrow 1, \forall 1 \leq i, j \leq N$
for $k = 1$ to 10 do
    $S(k) = YS(k-1)X^T + Y^T S(k-1)X$
    $k \leftarrow k + 1$
end for
Apply Hungarian assignment algorithm on similarity matrix $S$.
```

The usefulness of vertex similarity measure S for graph matching depends on the number of iterations considered for computing vertex similarity measure. In Blondel's approach the algorithm has to run until final scores of vertex similarity converge. But as we are removing normalization in the refined method there is no possibility for convergence, it is just an increasing function. To fix the number of iterations necessary for a good graph matching algorithm, we have done an experiment on 100 pairs of isomorphic graphs of different sizes M1000, M600, M200, S100, S40 from the benchmark data set [10] and calculated the accuracy of exact graph matching at all iterations. As shown in Figure3, it can be observed that by 10 iterations, the refined algorithm gives 100% graph matching for all pairs of isomorphic graphs from data set. Hence we fix the maximum value for k, maximum number of iterations as 10.

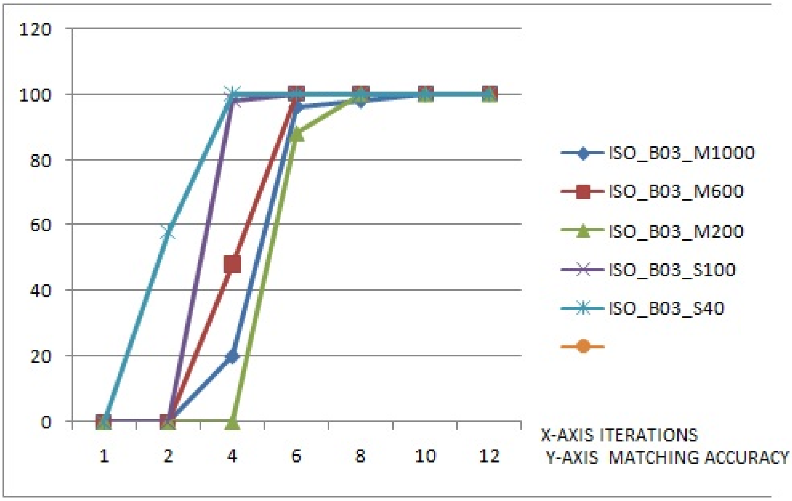


Fig. 3. The graph shows that the calculation of refined vertex similarity that is useful for graph matching has to run for only **10 iterations**

4 Experimentation on Benchmark Data Set

Firstly we intend to compare the performance of the refined algorithm with Blondel's algorithm. This testing is done on entire data base of isomorphic graphs and results are tabulated in the Section 4.1. Also we implemented Zager's algorithm on the entire data set but present a sample of our results comparing all the three algorithms: our algorithm and the graph matching algorithms using Blondel's similarity measure and also Zager's measure and the results are shown in the Table 5. The entire implementation is carried out using R-language [11].

4.1 Results of Implementation

The results of each type of graphs are given as a pair of tables, the first one containing large graphs whose sizes range from 200 to 1000 and the second table having small graphs ranging between 20 and 100.

Table 1. Bounded Valence Regular and irregular large and small graphs

Graph Type	M200	(%)	M400	(%)	M600	(%)	M800	(%)	M1000	(%)
	Blondel	Refined	Blondel	Refined	Blondel	Refined	Blondel	Refined	Blondel	Refined
ISO_B03	86	100	57	100	42	100	29	100	13	100
ISO_B03M	0	99	0	97					0	92
ISO_B06	100	100	100	100	100	100	100	100	100	100
ISO_B06M	0	100	0	100	0	100	0	100	0	100
ISO_B09	100	100	100	100	100	100	100	100	100	100
ISOB_09M	0	100	0	100	0	100	0	100	0	100
Graph Type	S20	(%)	S40	(%)	S60	(%)	S80	(%)	S100	(%)
	Blondel	Refined	Blondel	Refined	Blondel	Refined	Blondel	Refined	Blondel	Refined
ISO_B03	100	100	100	100	97	100	98	100	95	100
ISO_B03M	30	100	0	100	0	99	0	98	0	99
ISO_B06	100	100	100	100	100	100	100	100	100	100
ISO_B06M	54	100	7	100	0	100	0	100	0	100
ISO_B09	100	100	100	100	100	100	100	100	100	100
ISO_B09M	72	100	23	100	1	100	0	100	0	100

It should be noted that the refined algorithm performs significantly better than Blondel's algorithm. A few of the mesh graphs, especially M2D and M3D which are regular graphs pose to be a challenge to both the algorithms as seen in the Table 2.

Table 2. 2D-mesh graphs

Graph Type	M196	(%)	M400	(%)	M576	(%)	M784	(%)	M1024	(%)
	Blondel	Refined	Blondel	Refined	Blondel	Refined	Blondel	Refined	Blondel	Refined
ISO_M2D	0	0	0	0	0	0	0	0	0	0
ISO_M2Dr2	0	100	0	100	0	100	0	100	0	100
ISO_M2Dr4	0	100	0	100	0	100	0	100	0	100
ISO_M2Dr6	0	100	0	100	0	100	0	100	0	100
Graph Type	S16	(%)	S36	(%)	S64	(%)	S81	(%)	S100	(%)
	Blondel	Refined	Blondel	Refined	Blondel	Refined	Blondel	Refined	Blondel	Refined
ISO_M2D	4	0	0	0	0	0	0	0	0	0
ISO_M2Dr2	49	100	4	100	0	100	0	100	0	100
ISO_M2Dr4	41	100	2	100	1	100	0	100	0	100
ISO_M2Dr6	45	100	3	100	0	100	0	100	0	100

Similarly 3D-mesh graphs are also found to be challenging to both Blondel's and the refined algorithms as seen in Table 3. Refined method gives 100% matching for all isomorphic graphs except for bounded valence graphs of type ISO_B03M and mesh graphs of type ISO_M2D, ISO_M3D. In case of ISO_B03M graphs, the accuracy of matching is nearly 92-99% which is nearer to 100% matching. But in case of mesh graphs type ISO_M2D, ISO_M3D, as these are regular graphs where each node is having equal degree and it is not really possible to distinguish which nodes are similar in two input graphs.

Table 3. 3-Dimensional large and small mesh graphs

Graph Type	M216	(%)	M343	(%)	M512	(%)	M729	(%)	M1000	(%)
	Blondel	Refined	Blondel	Refined	Blondel	Refined	Blondel	Refined	Blondel	Refined
ISO_M3D	0	0	0	0	0	0	0	0	0	0
ISO_M3Dr2	0	100	0	100	0	100	0	100	0	100
ISO_M3Dr4	0	100	0	100	0	100	0	100	0	100
ISO_M3Dr6	0	100	0	100	0	100	0	100	0	100

Graph Type	S27	(%)	S64	(%)	S125	(%)
	Blondel	Refined	Blondel	Refined	Blondel	Refined
ISO_M3D	0	0	0	0	0	0
ISO_M3Dr2	0	100	0	100	0	100
ISO_M3Dr4	15	100	0	100	0	100
ISO_M3Dr6	27	100	0	100	0	100

Table 4. Randomly connected graphs

Graph Type	M200	(%)	M400	(%)	M600	(%)	M800	(%)	M1000	(%)
	Blondel	Refined	Blondel	Refined	Blondel	Refined	Blondel	Refined	Blondel	Refined
ISO_R001	0	100	0	100	0	100	0	100	0	100
ISO_R005	0	100	0	100	0	100	0	100	0	100
ISO_R01	0	100	0	100	0	100	0	100	0	100

Graph Type	S20	(%)	S40	(%)	S60	(%)	S80	(%)	S100	(%)
	Blondel	Refined	Blondel	Refined	Blondel	Refined	Blondel	Refined	Blondel	Refined
ISO_R001	1	92	0	94	0	92	0	97	0	100
ISO_R005	3	99	0	100	0	100	0	100	0	100
ISO_R01	6	100	0	100	0	100	0	100	0	100

Finally, the implementation results obtained on randomly connected graphs are given below.

Finally, a sample of the results obtained while comparing the three algorithms which includes that of Zager et al. is given in Table 5. It is clear that the refined Blondel's algorithm outperforms the other two and surprisingly, Zager's incorporation of edge similarity within Blondel's measure does not seem to help the final retrieval of isomorphism.

5 Analysis and Conclusion

In this paper we have presented a new vertex similarity measure based on Blondel's measure. Zager et al. also refine Blondel's measure and hence we compare our algorithm with Zager's algorithm for graph matching. We have computed and compared the results on isomorphic graphs of standard benchmark data set [10] for Blondel, Zager and refined Blondel methods. It is clear that the refined Blondel's algorithm outperforms the other two and surprisingly, Zager's incorporation of edge similarity within Blondel's measure does not seem to help the final retrieval of isomorphism. Blondel's and Zager's formulations are mathematically elegant with proofs for convergence. Only empirical investigation would have revealed the effect of normalization and the subsequent impact on the applicability

Table 5. Comparison of performance of Zager, Blondel and the proposed refined algorithm. Refined algorithm is seen to outperform the other two.

Graph Type	S20 (%)			S40 (%)			S60 (%)			S80 (%)			S100 (%)		
	Zager	Blondel	Refined	Zager	Blondel	Refined	Zager	Blondel	Refined	Zager	Blondel	Refined	Zager	Blond	Refined
ISO_B03M	9	30	100	0	0	100	0	0	99	0	0	98	0	0	99
ISO_B06M	13	54	100	1	7	100	0	0	100	0	0	100	0	0	100
ISO_B09M	22	72	100	1	23	100	0	1	100	0	0	100	0	0	100

of these measures. Hence, thorough implementation of all the algorithms with and without normalization of the measures is the most important contribution of this paper.

The ultimate goal of solving approximate graph matching algorithm remains to be tackled. The refined measure has to be tested for the other scenarios of approximate graph matching like subgraph isomorphism and common subgraph problem for its potential to be really tested. This evaluation and applications to application domains in bioinformatics forms part of the ongoing work.

Acknowledgment

We would like to thank reviewers for their valuable suggestions. The first author would like to thank his friend Mr.Saifuddin Kaijar for valuable discussions.

References

1. Leicht, E.A.: Vertex similarity in networks. Phys. Rev. E 73, 26120 (2006)
2. Kleinberg, J., Authoritative, M.: sources in a hyperlinked environment. J. ACM 46(5), 604–632 (1999)
3. Blondel, V.D.: A Measure of similarity between graph vertices: Applications to synonym extraction and web searching. Siam. Rev. 46(4), 647–666 (2004)
4. Melnik, S.: Similarity flooding: A versatile graph matching algorithm and its application to schema Matching. In: ICDE, pp. 117–128 (2002)
5. Zager, L.A.: Graph similarity scoring and matching. Applied mathematics letters 21, 86–94 (2008)
6. Kpodjedo, S., Galinier, P., Antoniol, G.: Enhancing a tabu algorithm for approximate graph matching by using similarity measures. In: Cowling, P., Merz, P. (eds.) EvoCOP 2010. LNCS, vol. 6022, pp. 119–130. Springer, Heidelberg (2010)
7. Raymond, J.W.: RASCAL: Calculation of graph similarity using maximum common edge sub graphs. The Computer Journal 45(6), 631–644 (2002)

8. Kuhn, H.: The Hungarian method for the assignment problem. Naval Research Logistic Quarterly 2, 83–97 (1955)
9. Foggia, P.: A Database of Graphs for Isomorphism and Sub-Graph Isomorphism Benchmarking,
http://amalfi.dis.unina.it/graph/
10. Foggia, P.: Benchmarking data set, http://amalfi.dis.unina.it/graph/
11. The R project for statistical computing www.r-project.org

Variable Length Virtual Output Queue Based Fuzzy Adaptive RED for Congestion Control at Routers

Pramod Kumar Singh and Santosh Kumar Gupta

Computational Intelligence and Data Mining Reasearch Laboratory
ABV-Indian Institute of Information Technology & Management Gwalior, India
pksingh@iiitm.ac.in, shanty.santosh@gmail.com

Abstract. Internet routers play an important role during the time of network congestion. All the Internet routers have some buffer at input and output ports, which hold the packets at the time of congestion. Many queue management algorithms have been proposed but they focus on fixed queue limit. Recognizing the fact that active queue management algorithms have fixed maximum queue limit, we direct our attention to variable length queue limit for Combined Input Output Queued (CIOQ) switches. We incorporated our proposed technique, which is a fuzzy logic control based generic variable length active queue management scheme in TCP/IP networks, to the drop-tail and the Adaptive RED (A-RED) algorithm. The empirical results show low packet loss and high queue utilization in modified algorithms (augmented with variable length active queue management scheme) in comparison to the original drop-tail, RED and A-RED algorithms.

Keywords: Congestion Control, Fuzzy Logic Controller (FLC), Active Queue Management (AQM), Virtual Output Queue (VOQ), Combined Input Output Queued (CIOQ) Switch, Adaptive RED (A-RED).

1 Introduction

Congestion is a critical issue as it reduces the overall throughput of the network and users experience greater delay. Though the routers play an active role in its resource allocation to effectively control/prevent congestion, it is still a major cause of concern because of ever growing Internet and ever growing number of users; they increase the amount of data to be carried over the Internet. Todays Internet routers have some buffer at input and output ports. The buffer size of the router should be large enough to accommodate the packets during the time of congestion but, at the same time, should also take care of the queuing delay. It demands for an optimum size and efficient management of buffers at the input/output ports. This is known as active queue management (AQM) [6].

Most of the Internet routers run drop-tail gateways. However, drop-tail buffer management introduces large queuing delays in bursty traffic [13]. The RED algorithm [4] [12] [14] manages the (buffer) queue more effectively and monitors the average queue length. The average queue length is compared with minimum threshold (min_{th}) and maximum threshold (max_{th}). If the queue length is less than min_{th} then all

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 123–134, 2011.

the incoming packets are accepted. If the queue length is in between min_{th} and max_{th} then packets are dropped with a probability that increases linearly up to maximum drop probability (max_p) and if the queue length exceeds the max_{th} then all the incoming packets are dropped. The most important advantage of RED is that it keeps the average queue length low to allow occasional burst of packets in queue. However, it has several shortcomings, e.g., a high degree of sensitivity towards its operating parameters, unfairness to flows with different round-trip times, the problem of global synchronization. A related weakness of RED is that throughput is also sensitive to the traffic load and the RED parameter. In particular RED does not perform well when the average queue length becomes larger than max_{th}. It results in a significant decrease in throughput and a significant increase in the drop rate [5] [17].

The A-RED [11] [13], proposed by one of the authors of RED, attempts to solve the problem of need for continuously (re)tuning RED parameters. In particular, A-RED adjusts the value of maximum drop probability (max_p) to keep average queue length within a target range half way between the min_{th} and max_{th}. It is shown in Figure 1. Though A-RED attempts to tune the RED parameters for a robust behavior, it fails to do so in various dynamic cases as A-RED retains RED's basic linear structure.

```
Every interval seconds:
    if (avg > target and $max_p \leq 0.5$)
        increase $max_p$;
        $max_p \leftarrow max_p + \alpha$ ;
    else if (avg < target and $max_p > 0.01$)
        decrease $max_p$:
        $max_p \leftarrow max_p * \beta$;
```

Variables:

avg: average queue length

Fixed parameters:

Interval: time; 0.5 seconds;

target : target for avg ;

[min_{th} +0.4*($max_{th} - min_{th}$), min_{th} + 0.6*(max_{th} - min_{th})].

α: increment; min (0.01, max_p / 4)

β: decrease factor; 0.9

Fig. 1. Adaptive RED algorithm

Many other active queue management schemes are also reported in the literature, e.g., Random Exponential Marking (REM) [2], fuzzy Proportional Integral (PI) [19], and Adaptive Virtual Queue (AVQ) [15]. All these existing active queue management schemes are based on the equation model. These equation models use various control parameters, which are dependent on different network parameters, e.g., number of flows, round trip time. However, it is very difficult to set these parameter as TCP/IP network is dynamic in nature. Many researchers, e.g., [3], [7], [8], [10], adopted fuzzy logic controller (FLC) to set these parameters dynamically in the congestion control algorithms because of its strength in controlling highly nonlinear, complex systems.

In this paper, we design a fuzzy logic controller (FLC) based on fuzzy logic set theory [23], [24] for computing the change in virtual output queue (VOQ) length [1],

[21], [18] according to its instantaneous queue length. It is a generic variable length active queue management scheme, which may be incorporated to any VOQ based active queue management congestion control mechanism to improve its performance while keeping the basic structure of the original algorithm same. In this paper, we incorporate our proposed generic method to the drop-tail and the A-RED and obtained encouraging results.

Rest of the paper is organized as follows. Section 2 discusses about variable length VOQ. In Section 3, we present our proposed generic variable length active queue management method and its application to drop-tail (Fuzzy drop-tail) and A-RED (Fuzzy A-RED). The rule base design of FLC is presented in Section 4. We discuss the performance of proposed method through a set of extensive simulation and compare the obtained results with well-known methods in Section 5. Finally in section 6, we present our conclusion.

2 Variable Length VOQ

The combined input and output queuing scheme uses buffers at both input and output modules of a switch, and a switch that employs this queuing scheme is called a CIOQ switch [9]. Every input port maintains a virtual queue, known as virtual output queue (VOQ), at its input buffer for each output port. In other words, for an N×N switch, each input port i ($1 \leq i \leq N$) maintains a separate set of FIFO queues for each output port j ($1 \leq j \leq N$), named as $\text{VOQ}_{\text{i,j}}$. Therefore, there are N sets of $\text{VOQ}_{\text{i,j}}$ queues at each input port. The incoming packets are stored in appropriate VOQ according to their destination address; the $\text{VOQ}_{i,j}$ buffers packet at input port i, which is destined to output port j. The buffer space of an input port is divided according to the number of VOQs and each VOQ has a fixed maximum queue limit to store the incoming packets. If there is no space for incoming packet at VOQ, the packet is dropped. However, it is desirable and a good strategy to vary (increase or decrease) the address space of VOQs at run time as per the requirement for efficient utilization of the queue while the buffer size at the input port is fixed.

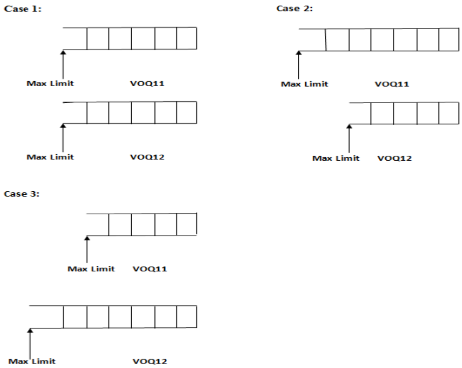


Fig. 2. In Case 1, the maximum queue limit of both VOQs is equal; in case 2, the maximum queue limit for VOQ_{11} is less than the VOQ_{12} and in case 3, the maximum queue limit for VOQ_{11} is greater than the VOQ_{12}

Our approach is based on the fact that we can vary the buffer size of the VOQ during the processing time as per the requirement. Figure 2 shows three possible cases for 2×2 switch. We can modify any VOQ based active queue management algorithm which uses a fixed queue limit. The amount of variation in the maximum queue limit of VOQs (while the total buffer size of an input port is fixed) is calculated by using the fuzzy logic controller.

3 Fuzzy Logic Controller

The fuzzy Controller is a controller which is based on the fuzzy logic based rules and often contains nonlinear mapping. The idea of FLC was initially introduced by Zadeh [24] and first applied by Mamdani [16] in an attempt to control systems that are difficult to model mathematically. FLC may be viewed as a way of designing feedback controllers where it is convenient and effective to build a control algorithm without relying on formal models of the system. The control algorithm is encapsulated as a set of commonsense rules. FLC has been applied successfully in control system for which analytical models are not easily obtained or the model itself, if available, is too complex and highly non-linear.

Our approach is to design a non-linear fuzzy logic controller, which operates at each input port of the router. For example, in a 2×2 switch at input port1 the fuzzy controller has two virtual queues, namely VOQ_{11} and VOQ_{12}, which may be in the low, average and high queue length states (refer, Figure 3). The low, average and high represent the status of input variables in linguistic form, which change dynamically over time. In order to determine the linguistic values of input and output, we partitioned the input and output space. The controller changes the maximum queue limit (varies the maximum buffer size of each VOQ) according to status of the inputs. Each of the input variables is represented by a fuzzy set.

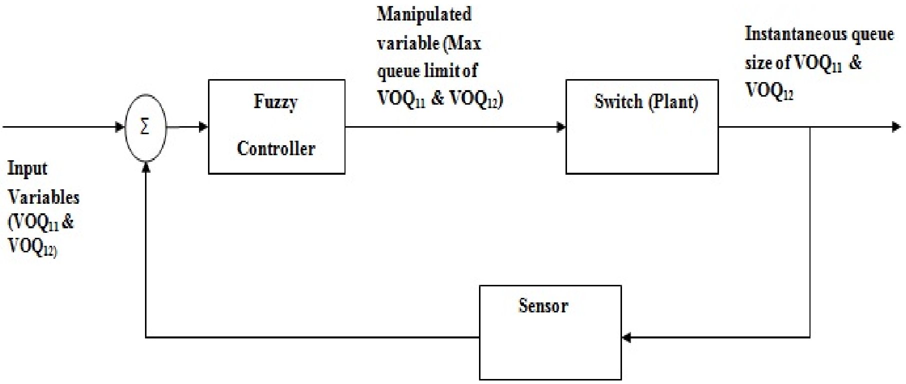

Manipulated variable (Max queue limit of VOQ_{11} & VOQ_{12})
Instantaneous queue size of VOQ_{11} & VOQ_{12}
Σ
Fuzzy Controller
Switch (Plant)
Input Variables (VOQ_{11} & VOQ_{12})
Sensor

Fig. 3. Fuzzy Logic Controller for Queue Management System Model

A notation convention for fuzzy sets [22], when the universe of discourse, X, is discrete and finite, is shown in Eq. 1. Here, A is a fuzzy set.

$$\mathrm{A}= \{\frac{\mu_{\mathrm{A}}(\mathrm{x}_1)}{x_1} + \frac{\mu_{\mathrm{A}}(x_2)}{x_2} + \;\dots\dots\}=\{\textstyle\sum_i \frac{\mu_{\mathrm{A}}(\mathrm{x}_\mathrm{i})}{x_i}\} \tag{1}$$

For our algorithm, the fuzzy sets are represented as follows (refer, Eq. 2):

$$\mathrm{VOQ}_{11}= \{ \;\frac{\mu_1}{low} + \frac{\mu_2}{avg} + \frac{\mu_3}{\mathrm{high}} \;\}$$

$$\mathrm{VOQ}_{12}= \{ \;\frac{\mu_1}{low} + \frac{\mu_2}{avg} + \frac{\mu_3}{\mathrm{high}} \;\} \tag{2}$$

Where VOQ_{11} and VOQ_{12} are fuzzy variables; low, average, and high are the possible values for fuzzy variables, and μ_1, μ_2, and μ_3 are the membership functions of the fuzzy variable.

3.1 Variable Length VOQ for Drop-Tail Algorithm

In modified drop-tail algorithm the maximum queue limit of each VOQ is not fixed. The FLC uses the instantaneous queue length as feedback, which is measured periodically, to compute new limit of the queue length. In other words, the controller varies the maximum queue length limit (refer, manipulated variable in Figure 3) as per sampled queue length of each VOQ. Variation in queue limit gives the low loss rate in comparison to simple drop-tail algorithm while the average queue length of both the algorithm is approximately same. It means modified drop-tail algorithm gives higher queue utilization in comparison to original drop-tail.

3.2 Fuzzy Adaptive RED

The overall guideline for fuzzy A-RED algorithm is same as A-RED algorithm except the variable *target*. In A-RED algorithm (refer, Section 1) the target value is fixed and is calculated as follows:

Target value:

$$[min_{th} + 0.4*(\; max_{th} - min_{th}\;),\; min_{th} + 0.6*(max_{th} - min_{th})]$$

In modified A-RED algorithm, which has the variable length VOQ, the value of max_{th} and min_{th} is not fixed. As a result the target value of the A-RED algorithm changes in each time interval with the varying values of min_{th} and max_{th} according to fuzzy logic controller. For n^{th} time interval the target value is calculated as follows:

Target value:

$$[min_{th}\,(n) + 0.4*(\; max_{th}\,(\mathrm{n}) - min_{th}\,(\mathrm{n})),\; min_{th}\,(\mathrm{n}) + 0.6*(max_{th}(\mathrm{n}) - min_{th}(\mathrm{n}))]$$

Here the min_{th}(n) and max_{th}(n) is the minimum and maximum threshold at n^{th} time interval respectively. Fuzzy A-RED removes A-RED's dependence on target value. Adapting the target value results in low packet loss and low average queue length in comparison to A-RED.

4 Rule Based Design

The FLC uses some linguistic rules for each input port. These linguistic rules control the system under different operating conditions. Usually multi-input FLC makes it easier to describe the system dynamics linguistically. We expect that we can tune the system and improve the behavior of active queue management algorithm, e.g., drop-tail, A-RED, by using variable length VOQs. The fuzzy rule base (IF-THEN rules) for the 2×2 CIOQ switch is presented linguistically in Figure 4 and same is represented in tabular form in Table. 1. The control surface defined by the FLC rules is shown in Figure 5.

Usually to define the linguistic rules of a fuzzy variable, Gaussian, triangular or trapezoidal shaped membership function are used. Since triangular and trapezoidal shaped function offer more computational simplicity, we have selected them for our rule base. The rule base is fine-tuned by observing the progress of simulation, such as packet loss occurrences and number of buffered packets at each VOQ.

/* linguistic rules for each Input port */
/* initially voq$_{11}$ and voq$_{12}$ queue limit is equal and set to maximum queue limit.*/
/* if voq$_{11}$ queue limit increases, then the voq$_{12}$ queue limit decreases to make input port buffer size fixed */
/* Set of linguistic rules defining the control surface of FLC */

if voq$_{11}$ is low and voq$_{12}$ is low then voq$_{11}$_modified_queue_ length is equal to max queue limit.
if voq$_{11}$ is average and voq$_{12}$ is low then voq$_{11}$_modified _queue_ length is greater than max queue limit.
if voq$_{11}$ is high and voq$_{12}$ is low then voq$_{11}$_modified _queue_ length is greater than max queue limit.

if voq$_{11}$ is low and voq$_{12}$ is average the voq$_{11}$_modified_queue_length is lesser than max queue limit.
if voq$_{11}$ is average and voq$_{12}$ is average then voq$_{11}$_modified_queue_length is equal to max queue limit.
if voq$_{11}$ is high and voq$_{12}$ is average then voq$_{11}$_modified_queue_length is greater than the max queue limit.
if voq$_{11}$ is low and voq$_{12}$ is high the voq$_{11}$_modified_queue_length is lesser than max queue limit.
if voq$_{11}$ is average and voq$_{12}$ is high then voq$_{11}$_modified_queue_length is lesser than max queue limit.
if voq$_{11}$ is high and voql$_{2}$ is high then voq$_{11}$_modified_queue_length is equal to max queue limit.

Fig. 4. Fuzzy rule base for 2×2 CIOQ switch

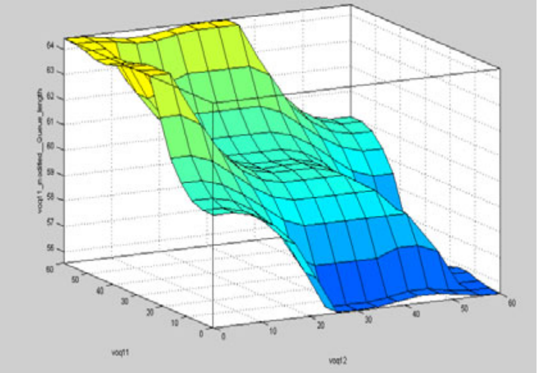


Fig. 5. Control decision surface of the Fuzzy Logic Controller shaped by rule base and linguistic variables

Table 1. Rule base of fuzzy controller

Input variable		Output (manipulated variable)	
VOQ_{11}	VOQ_{12}	VOQ_{11} Queue limit	VOQ_{12} Queue limit
low	low	equal to max queue limit	equal to max queue limit
average	low	greater than max queue limit	less than max queue limit
high	low	greater than max queue limit	less than max queue limit
low	average	less than max queue limit	greater than max queue limit
average	average	equal to max queue limit	equal to max queue limit
high	average	greater than max queue limit	less than max queue limit
low	high	less than max queue limit	greater than max queue limit
average	high	less than max queue limit	greater than max queue limit
high	high	equal to max queue limit	equal to max queue limit

5 Simulation Result

In this Section, We compare our scheme with the original drop-tail, RED and A-RED algorithm.

5.1 Experimental Setup

We perform our simulation on 2×2 CIOQ [21] switch as shown in Figure 6. The size of the buffer at input and the output port is 120 and 100 respectively. Buffer size at the input port is segmented according to the number of VOQs. Each input port carries multiplexed TCP Reno flows. The TCP flows are generated at separate source nodes and then multiplexed into the backbone before reaching at the input port of the switch. In this experiment, we use 4 source nodes at each input port, hence, total 8 source nodes generates TCP flows in the range of 50 to 500 sessions to the input ports of the switch. The size of the packets is 1000 bytes. The queue monitoring interval is set to 0.0001 sec. In A-RED algorithm α is set to 0.01 and β is set to 0.9.

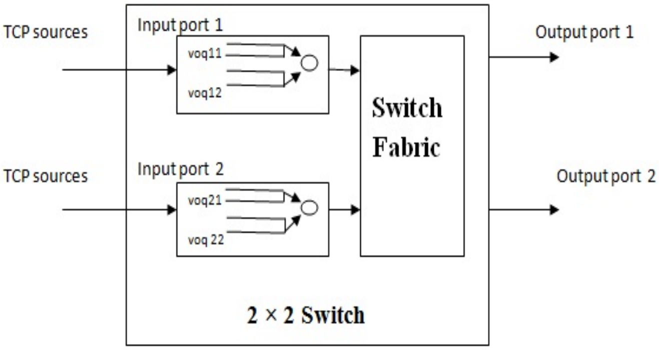


Fig. 6. 2×2 CIOQ switch

5.2 Input Queue Length

Figure 7 displays the average input queue length in the queue management unit when we use the Variable length VOQ based drop-tail and Fuzzy A-RED. The corresponding simulation results of original RED, drop-tail and A-RED are also shown for the comparison. For this simulation, the number of TCP sessions is 250 and speedup varies from 0.5 to 2.0. For RED, the minimum threshold is set to 19 packets and the maximum threshold is set to 59 packets for each VOQ. We use drop-tail algorithm at output ports for all the comparative algorithms.

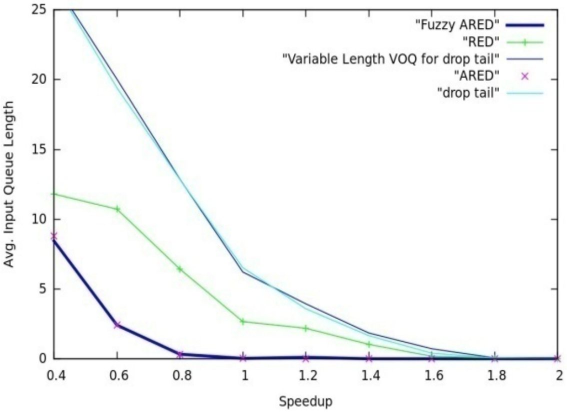


Fig. 7. Average input queue length v/s speedup for 2×2 Switch (load 200 TCP sessions)

On the input port, the drop-tail algorithm has the longest queue Length as it drops packets only when the buffer overflows. The suggested change in drop-tail algorithm, i.e., the variable length VOQs, has the same average queue length as original drop-tail algorithm whereas the RED algorithm has lower queue length as it drops packets even before buffer is overflow.

As A-RED keeps the average queue length away from max_{th}, the input queue length for A-RED is less than RED algorithm. As mentioned in section 3, the target value is not fixed in the Fuzzy A-RED algorithm; hence the result is even better than original A-RED algorithm.

5.3 Loss Rate

Loss rate is the ratio of the number of packets dropped and the number of packets sent. In this experiment, the speedup is fixed at 1.1whereas the load varies from 50 to 500 TCP sessions. Figure 8 shows that the loss rate of the fuzzy A-RED algorithm is lowest. Adjusting the max_p and target value of the Fuzzy A-RED algorithm avoids the higher packet loss as the average queue length oscillates near to the target value. We observe that the performance of fuzzy A-RED is comparatively better than the drop-tail, RED and the original A-RED because of the efficient management of the buffer space among the VOQs as per the requirement and the variation in the target value as well.

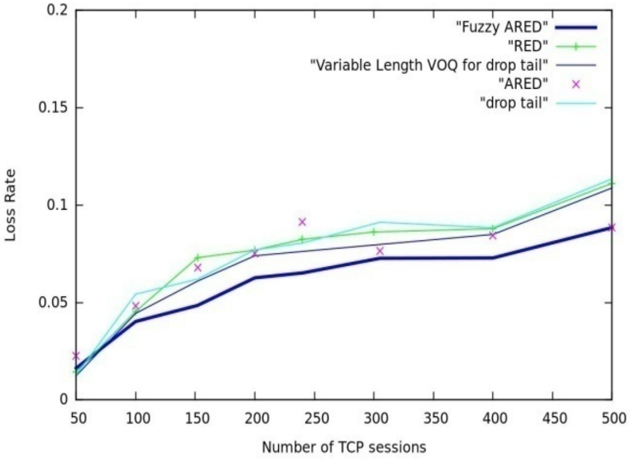


Fig. 8. Loss Rate of the 2×2 Switch at speedup 1.1

5.4 Buffer Size Variation

We investigate the buffer size of drop-tail with fixed length VOQ's (maximum queue limit for drop-tail is set to 60 packets) over time. Figure 9 shows the buffer size variation of VOQ_{11} and VOQ_{12} respectively. The buffer size of the both VOQs is not more than the 60 packets at any time instant.

Figure 10 shows the buffer size variation of proposed algorithm. The result shows that maximum queue limit of VOQ_{11} and VOQ_{12} is varying from 54-65 packets according to suggested approach, but at any instant of time the number of packets in

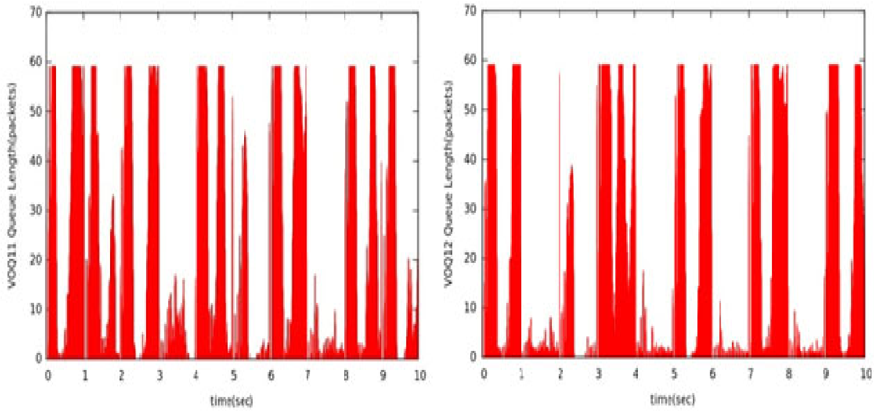


Fig. 9. Buffer size variation of VOQ_{11}& VOQ_{12} over time (for drop-tail algorithm)

both of the VOQs simultaneously is not more than 60**.** The reason is that at that instant of time the other VOQ had relatively less queue length hence it could relinquish a portion of its unused address space to share with other heavily loaded VOQ. Therefore, the proposed variable length VOQ algorithm minimizes the packet loss and provides efficient management of the buffer space at input ports.

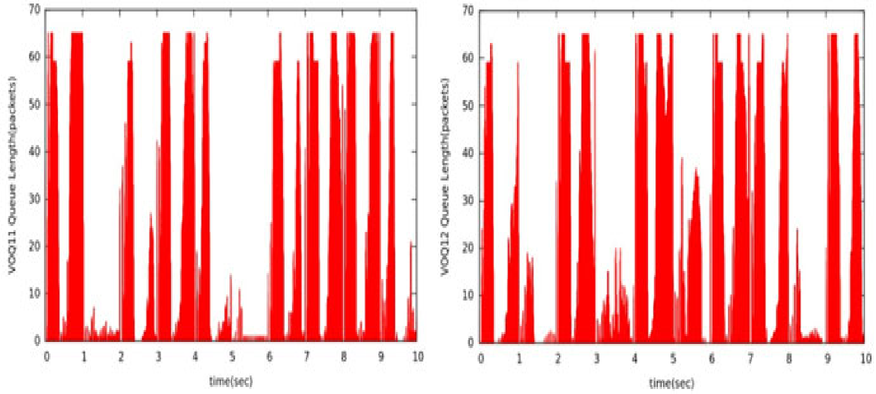


Fig. 10. Buffer size variation of VOQ_{11}&VOQ_{12} over time (variable length VOQ for drop-tail)

As fuzzy A-RED algorithm has the basic structure of the A-RED algorithm, the size of the buffer oscillates near to the target value. Figure 11 shows that the number of packets buffered is half way between maximum and minimum threshold.

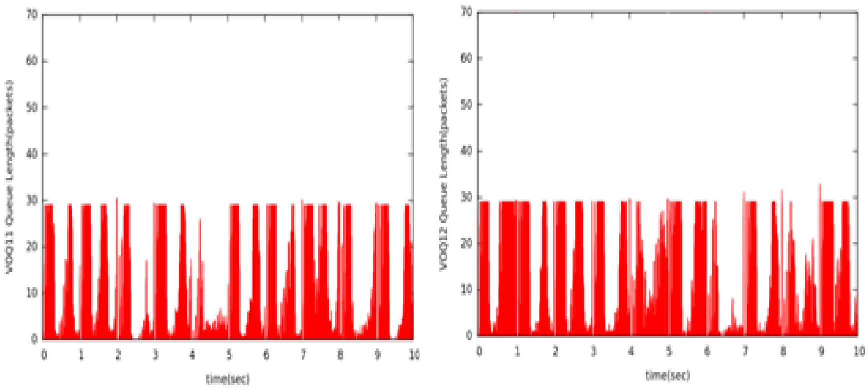


Fig. 11. Buffer size variation of VOQ_{11}&VOQ_{12} over time (Fuzzy A-RED)

6 Conclusion

Our proposed scheme is a generic concept which may be used to improve the performance of any active queue management scheme. In this paper, we have incorporated variable length VOQ with the active queue management algorithms, e.g., drop-tail, A-RED algorithm, which improves the performance in terms of queuing delay, average queue length and loss rate**.** Variable length VOQ for drop-tail is the improvement over simple drop-tail algorithm which utilizes the queue more efficiently. In fuzzy A-RED target value dynamically change with change in VOQ length which gives improvement over A-RED algorithm in dynamic network environment. We formulate an effective and efficient technique for queue management using the fuzzy logic control, to solve the problem of congestion in TCP/IP networks. We have demonstrated that in the real world for the non-linear and complex system the fuzzy logic control gives the acceptable solution with the help of linguistic models. Additionally, it does not require any change in the switch hardware.

In future work, we plan to explore the proposed scheme on other active queue management algorithms such as REM, AVQ and in other network scenario like Diff-Serv, and in different types of traffic condition. We also plan to check the stability and performance of the proposed method on the larger switches.

References

1. Anderson, T.E., et al.: High Speed Switch Scheduling for Local Area Networks. ACM Trans. on Computer System 11, 319–352 (1993)
2. Athuraliya, S., Li, V.H., Low, S.H., Yin, Q.: REM: active queue management. IEEE Network Magazine 15(3), 48–53 (2001)
3. Behrouz, S., Amir, M.R., Mahdipour, E.: A New Fuzzy Congestion Control Algorithm in Computer Networks. In: International Conference on Future Computer and Communication, pp. 314–318 (2010)

4. Bonald, T., May, M., Bolot, J.C.: Analytic Evaluation of RED Performance. In: proceedingof the 19th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM), vol. 3, pp. 1415–1424 (2000)
5. Brandauer, C., Iannaccone, G., Diot, C., et al.: Comparison of Tail Drop and Active Queue Management Performance for Bulk-data and Web-like Internet Traffic. In: Proceedings of6th IEEE Symposium on Computers and Communications(ISCC), pp. 122–129 (2001)
6. Braden, B., Clark, D., Crowcroft, J., Davie, B., Deering, S., Estrin, D., Floyed, S., Jacobson, V., Minshall, G., Partidge, C., Peterson, L., Ramakrishnan, K., Shenker, S., Wroclawski, J., Zhang, L.: Recommendation on Queue Management and Congestion Avoidance in the Internet. RFC 2309 (1998)
7. Chrysostomou, C., Pitsillides, A., Sekercioglu, Y.A.: Fuzzy Explicit Marking: A Unified Congestion Controller for Best-Effort and Diff-ServNetworks. Computer Networks 53(5), 650–667 (2009)
8. Chrysostomou, C., Pitsillides, A., Rossides, L., Sekercioglu, A.: Fuzzy Logic Controlled RED: Congestion Control in TCP/IP Differentiate Services Networks. Soft Computing 8(2), 79–92 (2003)
9. Chuang, S.T., Goel, A., McKeown, N., Prabhakar, B.: Matching Output Queuing with a Combined Input/Output-Queued Switch. IEEE Journal On Selected Areas in Communications 17(6), 1030–1039 (1999)
10. Douligeris, C., Develekos, G.: A fuzzy Logic Approach to Congestion Control in ATM Networks. In: Proceedings of IEEE ICC 1995, Seattle, vol. 3, pp. 1969–1973 (1995)
11. Feng, W.C., Kandlur, D.D., Saha, D., et al.: A Self-Configuring RED Gateway. In: proceedings of 18th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM), vol. 3, pp. 1320–1328 (1999)
12. Floyd, S., Jacobson, V.: RandomEarly Detection Gateways for Congestion Avoidance. IEEE/ACM Trans. on Networking 1(4), 397–413 (1993)
13. Floyd, S., Gummadi, R., Shenker, S.: Adaptive RED: An Algorithm for Increasing the Robustness of RED's Active Queue Management. Technical Report, ICSI (2001)
14. Hollot, C.V., Misra, V., Towsley, D., et al.: Analysis and Design of Controllers for AQM Routers Supporting TCP Flows. IEEE Trans. on Automatic Control 47, 945–959 (2002)
15. Kunniyur, S., Srikant, R.: An Adaptive Virtual Queue (AVQ) Algorithm for Active Queue Management. IEEE/ACM Trans. on Networking 12(2), 286–299 (2004)
16. Mamdani, E.H.: Application of Fuzzy Algorithm for Simple Dynamic Plant. Proceedings of IEEE 121(12), 1585–1588 (1974)
17. May, M., Bolot, J., Diot, C., Lyles, B.: Reasons Not to Deploy RED. In: 7th International Workshop on Quality of Service, IWQoS 1999, pp. 260–262 (1999)
18. McKeown, N.: The iSLIP Scheduling Algorithm for Input-Queued Switches. IEEE/ACM Trans. on Networking 7(2), 188–201 (1999)
19. Misir, D., Malki, A., Chen, G.: Design and Analysis of a Fuzzy Proportional-Integral Derivative Controller. Fuzzy Sets and System 79, 297–314 (1996)
20. Network Simulator, NS-2, `http://nsnam.isi.edu/nsnam/`
21. Sundararajan, J., Zhao, K., Pamela, F., Muriel, M.Y.,, M.: A Modification to RED AQM for CIOQ Switches. In: IEEE Global Communication Conference (Globecom), Dallas, vol. 3, pp. 1708–1712 (2004)
22. Timothy, J.R.: Fuzzy Logic with Engineering Application. John Wiley, Chichester (2004)
23. Zadeh, L.A.: Fuzzy sets. Information and Control 8, 338–353 (1965)
24. Zadeh, L.A.: Outline of a New Approach to the Analysis of Complex System and Decision Process. IEEE Trans. on Systems, Man, and Cybernetics 3(1), 28–44 (1973)

An Efficient EA with Multipoint Guided Crossover for Bi-objective Graph Coloring Problem

Soma Saha, Gyan Baboo, and Rajeev Kumar

Dept. of Computer Science & Engineering
Indian Institute of Technology Kharagpur
Kharagpur, WB 721302, India
{somasaha,gbaboo,rkumar}@cse.iitkgp.ernet.in

Abstract. Graph Coloring Problem is a well-studied classical NP-hard combinatorial problem. Several well-known heuristics and evolutionary approaches exist to solve single-objective graph coloring problem. We have considered a bi-objective variant of graph coloring problem, in which the number of colors used and the corresponding penalty which is incurred due to coloring the end-points of an edge with same color, are simultaneously minimized. In this paper, we have presented an evolutionary approach with Multipoint Guided Crossover (MPGX) to minimize both objectives simultaneously. On applying proposed evolutionary algorithm over standard graph coloring problem instances, a guaranteed solution to the single-objective graph coloring problem is achieved. We have adapted a few well-known heuristics which are evolved for single-objective graph coloring problem to generate set of solutions for bi-objective graph coloring problem and obtained Pareto fronts. Empirical results show that proposed evolutionary algorithm with simple Multipoint Guided Crossover generates superior or (near-) equal solutions in comparison with the adapted heuristic solutions as well as with evolutionary algorithm solutions using a few crossover (Penalty-based Color Partitioning Crossover (PCPX) and Degree Based Crossover (DBX)) operators across entire Pareto front for considered bi-objective variant of graph coloring problem.

Keywords: Optimization methods, multi-objective optimization, combinatorial optimization, genetic algorithm, evolutionary algorithm, heuristics, graph coloring, Pareto front.

1 Introduction

Graph Coloring Problem (a.k.a GCP) has many applications in real-life, particularly in time-table scheduling, register allocation, circuit board testing, sudoku problem solving [1,2,3]. Deciding whether a given graph G is k colorable or not, is an NP-complete problem [4]. Even, approximating the chromatic number of a given graph G is an NP-hard problem within range $n^{1-\in}$ for any $\in< 0$,

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 135–145, 2011.

where n is the cardinality of vertex set [5,6]. Many exact algorithms exist to solve graph coloring problem for small instances of graphs or specialized graphs [7,8,9]. There are number of heuristics like Largest Degree Ordering (LDO) [10], DSatur/Saturated Degree Ordering (SDO) [7], Smallest Last Ordering (SLO) [11], Iterated Greedy [12], Incidence Degree Ordering (IDO) [13]; these heuristics are evolved to approximate the chromatic number of general graphs. Genetic algorithms [14,15,16,17,18] have been widely considered for standard graph coloring problem. Competing goals and highly complex large search space of multi-objective optimization problems boosts the need of multiple compromised solution set, known as Pareto-optimal set, instead of single optimal solution for a single objective/goal; hence, evolutionary approaches overpower the rest of the approaches. Evolutionary algorithms deal with a set of possible solutions in a single run due to its population based approach. Kumar et al. [19] proposed two penalty adjusting heuristics and two crossover operator, Penalty based Color Partitioning Crossover (PCPX) and Degree Based Crossover (DBX) for bi-objective variant of graph coloring problem.

We have combined and recasted a few well-known single-objective graph coloring heuristics to generate feasible solution sets for bi-objective variant of graph-coloring problem and obtained Pareto fronts.

In this paper, we have presented a novel bi-objective evolutionary/genetic algorithm model which minimizes both number of colors and corresponding incurred penalty simultaneously on applying proposed multipoint guided crossover operator. We have used well-known integer representation scheme [20] to represent chromosomes and a new multi-point guided crossover (MPGX) operator to reduce the penalty. We have adapted Smallest Degree Last (SDL) [11], DSatur-LDO [19], DSatur-IDO-LDO [19], Penalty Adjusting Heuristics (PAH) [19] and obtained Pareto fronts to compare with MPGX operator generated solution-set across the entire Pareto front. We assess the performance of MPGX operator in comparison with PCPX and DBX [19] operators. Empirical solutions of genetic/evolutionary algorithm (GA/EA) with proposed crossover operator are comparable to the solutions generated using recasted graph coloring heuristics as well as to the existing crossover operators for considered bi-objective variant of graph coloring problem.

In Section 2, we present an overview of basic definitions and problem formulation. Section 3 contains a short description of heuristics which are adapted in our work. Next, Section 4 contains the brief description of evolutionary model. Section 5 depicts empirical solutions of GA along with a comparison with a few considered different approaches and we draw conclusion in Section 6.

2 Basic Definitions and Problem Formulation

Definition 1 Multi-objective Optimization Problem (MOP).
In an MOP, a number of objectives have to be minimized/maximized along with constraints (optional) to achieve goal vectors which can be written as:
Maximize/Minimize : $F(\mathbf{X}) = \{f_1(\mathbf{X}), f_2(\mathbf{X}), \ldots, f_m(\mathbf{X})\}$

subject to satisfaction of the constraints:
$C(\mathbf{X}) = \{c_1(\mathbf{X}), c_2(\mathbf{X}), \ldots, c_k(\mathbf{X})\} \leq (0, \ldots, 0)$

A set of objective values constitutes an *objective* space and the collection of decision variables forms a *decision* space.

Definition 2 Pareto-optimal set.
Without loss of generality, we assume an m-objective minimization problem. We say that a vector of decision variables $x \in X'$ includes in Pareto-optimal (P) set as a Pareto-optimal point if another x^* does not exist such that $f_i(x^*) \leq f_j(x)$ for all $i = 1, 2, 3, \ldots, m$ and $f_i(x^*) < f_j(x)$ for atleast one j. Here, X' denotes the feasible region of the problem (i.e. where the constraints are satisfied).

Definition 3 Pareto dominance.
A vector $\overrightarrow{u} = (u_1, ..., u_k)$ dominate $\overrightarrow{v} = (v_1, ..., v_k)$ (denoted by $\overrightarrow{u} \preccurlyeq \overrightarrow{v}$) iff u is partially less than v; it can be represented as follows,
$\imath \in 1, 2, ..., k,\ u_i \leq v_i \wedge \exists\ i \in 1, 2, ..., k : u_i < v_i$.

Definition 4 Graph Coloring Problem (GCP).
Coloring of an undirected graph $G = (V,\ E)$ is to color the vertex set V with k (a positive integer) number of colors such that no adjacent vertices are colored with same color. The GCP problem is to find the smallest value of k with feasible vertex coloring.

Definition 5 Bi-Objective Graph Coloring Problem.
The Bi-Objective variant of GCP is to color the vertices of graph G with k (a positive integer) colors; here, we allow same color to the adjacent vertices. The penalty which is incurred due to coloring adjacent vertices with same color is minimized [19].

3 Adaptation of Heuristics

Many heuristics exist to solve classical graph coloring problem. We have adapted a few heuristics for bi-objective variant of graph coloring problem to yield Pareto fronts.

3.1 Smallest Degree Last

Matula and Beck [11] proposed Smallest Degree Last (SDL) heuristics, in which a vertex having least degree is colored with least priority and removed from graph. Similar procedure is repeated to the remaining vertices of the graph and highest priority vertex is colored first.

3.2 DSatur-LDO

In DSatur heuristics, next vertex selection for coloring depends on vertex's highest number of differently colored neighbors. Largest degree vertex is selected for next vertex coloring in Largest Degree Ordering (LDO). LDO is used to break a DSatur heuristics generated saturation degree tie. This approach outperforms smallest degree last heuristics and individual DSatur, LDO, IDO heuristics for bi-objective variant of graph coloring [19].

3.3 DSatur-IDO-LDO

Here, DSatur heuristics generated saturation degree ties are resolved by Incidence Degree Ordering (IDO) with first priority. Remaining unresolved ties are broken by LDO. This combined approach outperforms Smallest Degree Last and DSatur-LDO heuristics [19].

3.4 Penalty Adjusting Heuristics

Kumar et al. [19] proposed two penalty adjusting heuristics for this bi-objective variant of graph coloring problem. In first penalty adjusting heuristics, vertex coloring is started using two colors and stopped when penalty zero is reached for a particular value of color. With the aim to reduce the penalty for a given particular coloring, entire vertex set of the graph is scanned and reassigned with a color which minimizes the penalty for that vertex. The second penalty adjusting heuristics chooses a color for replacement with a directed way; whereas first penalty adjusting heuristics chooses a color randomly.

4 Evolutionary Approach

Initially, Multi-Objective Evolutionary Optimization approach was developed with the direction of minimizing or maximizing different objectives simultaneously. Recent trend shows that the basic traditional splitting strategy which decomposes a multi-objective problem into a number of scalar optimization subproblems and then, optimizes them simultaneously, reduces computational complexity at each generation and outperforms or performs similarly to few steady-state well-known MOEA (e.g. MOGLS [21], NSGA-II [22]) [23]. In this paper, we have proposed an EA approach for bi-objective graph coloring with Multipoint Guided Crossover operator where we have decomposed the two objectives, total number of colors used for graph coloring and the penalty incurred for assigning same color to the end-points of an edge. In this paper, we have shown that proposed Multipoint Guided Crossover operator which reduces basically the penalty incurred due to coloring the end-points of an edge with same color, confirms the solution for graph coloring problem and empirical results show superior or near-equal solution in comparison with the adapting heuristic solutions as well as with EA solutions using PCPX and DBX crossover operator across the entire Pareto front.

4.1 Evolutionary Algorithm

```
1: Input: Initial population of size N is generated randomly
2: Output: A set of improved solutions
3: Algorithm::
Choose population size N depending on size of the graph $G$ and proper
crossover probability $P_c$ ; Generate N individuals as initial population
**for** each $i = 0$ to Maximum # Iteration **do**
   **for** each $j = 0$ to $P_c$ * N1 **do**
      { N1 - modified population size during immediate last generation}
      a. Select two parent using a random selection scheme
      b. Perform guided multi-point crossover operation and get offspring
   **end for**
   Perform selection on Immediate old generation population and current set
   of offspring according to selection criteria
**end for**
Output the evaluated color set and corresponding penalty as Pareto solution.
```

Algorithm 1. Evolutionary Algorithm (EA)

4.2 Encoding Scheme

There are several encoding scheme to represent individuals like integer representation [20], order based representation [14]. In this work, we have considered integer representation scheme which is also known as the assignment representation [15,16]. It is position-based; color assigned at a particular gene of an individual represents the corresponding vertex's position where numbering of vertices is done in asynchronous order.

4.3 Initial Population

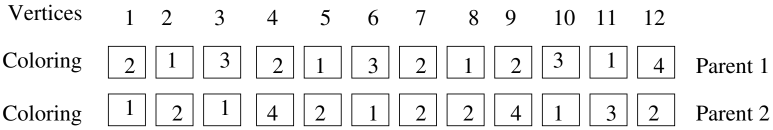
We have considered initial population size N = 1000. In the worst case, maximum number of colors needed to color a graph with zero penalty is equal to maximum degree of nodes of graph G + 1. Individuals for initial population are generated randomly for degree range 2 to maximum degree of graph G with equal distribution. For example, number of individuals generated with maximum color 2 is (N/maximum degree of G).

4.4 Crossover Operator

We have used multi-point guided crossover (MPGX) where number of crossover point depends on crossover probability (P_c) and total number of nodes ($|V|$) of given undirected graph G. Individuals are randomly selected from current population for recombination operation. A random point or node is chosen; depending on variation of assigned color of that node in both parent, we check whether swaping of color reduces penalty or not. Exchange of coloring takes place to either one parent or both, iff penalty reduces.

```
Count ← $P_c * |V|$
while Count $\neq \emptyset$ do
    $i$ ← a random node chosen
    if $i^{th}$ node of both parent have different color, say color $c1$ and $c2$ then
      if penalty reduces due to replacement of $c1$ color with $c2$ color in parent1 then
        color $i^{th}$ node of parent1 with $c2$ color
      end if
      if penalty reduces due to replacement of $c2$ color with $c1$ color in parent2 then
        color $i^{th}$ node of parent2 with $c1$ color
      end if
    end if
    Count ← Count - 1;
end while
```

Algorithm 2. Recombination operator



Vertices	1	2	3	4	5	6	7	8	9	10	11	12	
Coloring	2	1	3	2	1	3	2	1	2	3	1	4	Parent 1
Coloring	1	2	1	4	2	1	2	2	4	1	3	2	Parent 2

Fig. 1. Randomly selected parents for crossover operation

Figure 1 depicts a pair of randomly selected chromosome having $|V| = 12$. Let, vertex 4 is connected with vertices 1, 6, 7, 8, 9 and 12 in a given G and suppose, vertex 4 is randomly selected as a crossover point. In parent1 and parent2, vertex 4 is colored with two different color 2 and 4, respectively. Thus, penalty is calculated for vertex 4 with current color and probable-replacement color (color of vertex 4 in another parent) to ensure the reduction of overall penalty in offspring. Coloring of vertex 4 in parent1 adds penalty 3 to the overall penalty to the current coloring; replacing color 2 with color 4 in parent1 for vertex 4 reduces the overall penalty which motivates the generation of offspring. Replacement of color 4 with color 2 in parent2 for vertex 4 increases the overall penalty; thus a single offspring is generated using MPGX crossover operation for this particular example which is shown in Figure 2. If replaced color in offspring is not used for vertex coloring in parent, then replaced color is chosen as the not used color in parent within range 1 to maximum number color used in parent + 1.

Vertices	1	2	3	4	5	6	7	8	9	10	11	12	
Coloring	2	1	3	4	1	3	2	1	2	3	1	4	Offspring 1
Coloring	1	2	1	4	2	1	2	2	4	1	3	2	Parent 2

Fig. 2. Offspring after crossover operation

4.5 Selection Strategy

Individuals from current population and population generated from crossover operation are searched; we choose minimum penalty individuals for color 2 to the particular number of color where penalty is zero or minimum. The population size is reduced or remained same depending upon the number of minimum color where penalty is reached to zero value.

5 Experiments

5.1 Problem Instances

We have run our evolutionary algorithm on a set of standard benchmark graph coloring problem instances from DIMACS graph coloring challenge (http://mat.gsia.cmu.edu/COLOR/instances.html). We have considered various graphs with different densities (graph having same node size with various cardinality of edges) with 11, 120, 128, 184, 450 and 496 nodes. We have discussed here, the results obtained for fpsol2.i.1.col graph having 496 nodes and 11654 edges. We have seen similar nature of performance for our proposed GA on all standard benchmark instances. We set crossover probability (P_c) as 0.8 and observe the solution set after 3000 generations. We observe that EA generates optimal solutions within few generation for small graph instances. Thus, along with maximum 3000 generation, we have considered another stopping criteria which is based on histogram scheme. Constant population during consecutive 5 generation ends the EA execution which reduces the extra computational time for smaller graph instances.

5.2 Results and Discussions

In multi-objective problem domain, Pareto front is an easy and simple way to visualize and compare the nature of different solution-set across the entire front. Figure.3 depicts the obtained Pareto fronts for SDL, DSatur-LDO/SDO-LDO, DSatur-IDO-LDO/SDO-IDO-LDO, PAH (Penalty Adjusting Heuristics) heuristics and evolutionary algorithms using Penalty based Color Partitioning Crossover (PCPX), Degree Based Crossover (DBX) and Multi-Point Guided Crossover (MPGX) operator. PCPX outperforms over SDL, SDO-LDO and SDO-IDO-LDO. PAH and DBX shows near-equal solutions and outperform over PCPX. MPGX shows superior solutions for smaller number of colors and depicts near-equal solutions to DBX and PAH for rest of the front. In order to measure the performance of the multi-objective algorithms, the most challenging issues are maintaining the coverage of solutions across the entire Pareto front, sampling of the obtained solutions across the Pareto front and maintaining the convergence which is known as the distance of obtained solution front from the reference front. We have considered convergence metric [24], spread [25], and hyper volume/S-metric [26,27] to assess the performance of EA.

Avoiding Local Minima. It is a common concern that whether the obtained EA solution-set is near to the optimum Pareto front or not. Thus, we have considered multiple run of evolutionary algorithm for each crossover operator and combined the solution set to avoid trapping the search at local minima. Table 1 shows different metric values (Spread and Hypervolume) for different simulations and their combination for EA with our proposed recombination operation on fpsol2.i.1.col input instance.

Convergence. Deb and Jain [24] proposed convergence metric to evaluate the convergence of obtained solution-set towards a reference set; it measures the difference between obtained solution-set and reference set. Lower convergence value indicates the superiority of solution-set and the ideal value 0 indicates that obtained solution-set coincides with reference front. EA with MPGX gives better convergence over heuristics solutions.

Spread. Spread [25] measurement implies the diversity with respect to a reference front; it measures the distribution of the solutions in the obtained nondominated solution-set by computing a relative distance between consecutive solutions and takes care of the extent of the spread. Small spread value indicates a good distribution (when obtained solution set consists of extreme solutions and the distribution of intermediate solutions are uniform) and higher value indicates a bad distribution (when solutions get more and more closer from the ideal distribution and/or huge difference in extreme solutions) of solutions in the set.

Hypervolume. Hypervolume/S-metric [26,27] measures the multi-dimensional region which is enclosed by obtained Pareto fronts; it computes the volume of the search space dominated by a solution set. Hypervolume measurement depends on selection of a particular reference point [27]. Hence, in this work, reference point is considered depending on large set of solutions from all heuristics and EA outputs. The major advantage of this metric is that it can assess a solution-set independent of other solution-sets.

Reference front and reference point, both are considered depending on large set of solutions from all heuristics and EA outputs. Table 2 shows the convergence, spread and hypervolume metric calculation for Smallest Degree Last (SDL), DSatur-LDO, DSatur-IDO-LDO and Penalty Adjusting Heuristics (PAH) and PCPX, DBX and MPGX crossover operation on our proposed EA for fpsol2.i.1.col input instance. We have executed each heuristics and EA thrice to avoid trapping into local optimal and considered combined average metric values. The hypervolume and spread metrics show improved performance by combining simulations. DSatur-IDO-LDO heuristic outperforms Smallest Degree Last and DSatur-LDO heuristics. EA with MPGX outperforms over adapted bi-objective variant of single-objective graph coloring heuristics solutions and generates superior or near-equal solution set to Penalty Adjusting Heuristics and EA generated solution set using PCPX and DBX crossover across the complete Pareto front.

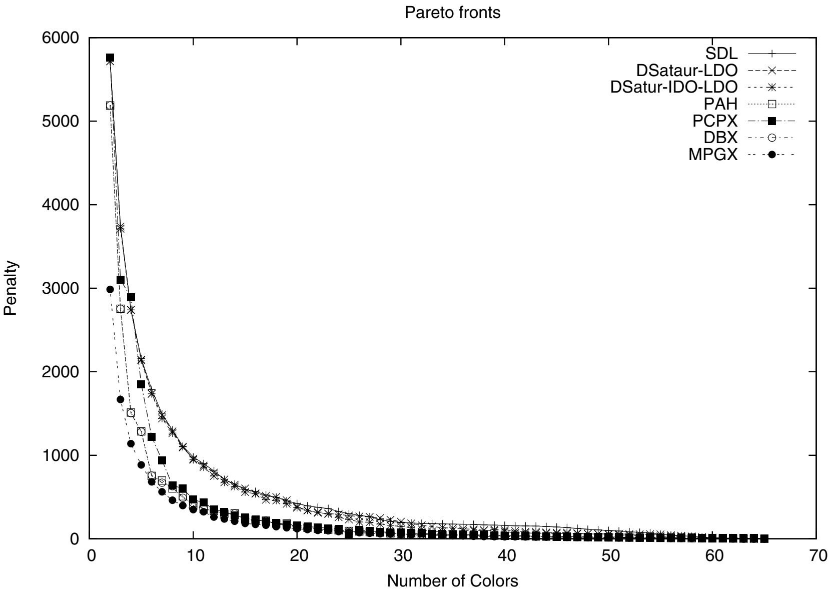


Fig. 3. Pareto fronts obtained from SDL, DSatur-LDO, DSatur-IDO-LDO, PAH heuristics and EAs using PCPX, DBX and MPGX crossover operators for standard benchmark graph instance having 496 nodes and 11,654 edges

Table 1. The Table shows different metric values (Spread and Hypervolume) for different simulations and their combination for EA with our proposed recombination operation on fpsol2.i.1.col input instance. The hypervolume and spread metrics show improved performance by combining simulations.

Simulation	**Spread**	**Hypervolume**
*Simulation*1	1.1537	353397
*Simulation*2	1.2257	345096
*Simulation*3	1.1537	353397
Combined	1.1777	350630

Table 2. Convergence, Spread and Hypervolume metric values for a standard benchmark graph of 496 nodes and 11654 edges

Metric	SDL	DSatur-LDO	DSatur-IDO-LDO	PAH	PCPX	DBX	MPGX
Convergence	0.0373	0.0309	0.0280	0.0005	0.0074	0.0004	0.0000
Spread	1.4160	1.0829	1.3970	1.6289	1.6303	1.5254	1.1777
Hypervolume	329220	330945	332031	345060	342821	345096	350630

6 Conclusions

Problem specific knowledge is a major driving criteria towards superior solution set for multi-objective evolutionary algorithms. We have designed a multi-point crossover operator with problem specific knowledge. Reducing population size with the gained zero penalty for a particular color, reduces number of colors used in next generations. Moreover, the stopping criteria of proposed EA minimizes the unnecessary computational effort for small graph instances. We monitor the performance of a few existing approaches and proposed EA with MPGX operator. Proposed EA with Multipoint Guided Crossover outperforms over adapted bi-objective variant of single-objective graph coloring heuristics solutions and generates superior or near-equal solution set to Penalty Adjusting Heuristics and EA generated solution set using PCPX and DBX crossover across the complete Pareto front.

References

1. Burke, E., Petrovic, S.: Recent research directions in automated timetabling. European J. operation research 140(2), 266–280 (2002)
2. Davis, T.: The mathematics of sudoku (2010), http://www.geometer.org/mathcircles/sudoku.pdf
3. Garey, M.R., Johnson, D.S., So, H.C.: An application of graph coloring to printed circuit testing. IEEE Transactions on Circuits and Systems 23(10), 591–599 (1976)
4. Karp, R.M.: Reducibility among combinatorial problems. In: Miller, R.E., Thatcher, J.W. (eds.) Complexity of Computer Computations, pp. 85–103. Plenum Press, NY (1972)
5. Feige, U., Kilian, J.: Zero knowledge and the chromatic number. J. Computer and System Sciences 57(2), 187–199 (1998)
6. Zuckerman, D.: Linear degree extractors and the inapproximability of max clique and chromatic number. Theory of Computing 3(6), 103–128 (2007)
7. Brelaz, D.: New methods to color the vertices of a graph. Commun. ACM 22(4), 251–256 (1979)
8. Caramia, M., Dell'Olmo, P.: Bounding vertex coloring by truncated multistage branch and bound. Networks 44(4), 231–242 (2004)
9. Mehrotra, A., Trick, M.A.: A column generation approach for graph coloring. J. Computing (JOC) 8(4), 344–354 (1996)
10. Zoellner, J., Beall, C.: A breakthrough in spectrum conserving frequency assignment technology. IEEE Trans. Electromagnetic Compatibility 19(3), 313–319 (1977)
11. Matula, D.W., Beck, L.L.: Smallest-last ordering and clustering and graph coloring algorithms. J. ACM 30(3), 417–427 (1983)

12. Marino, A., Prugel-Bennett, A., Glass, C.A.: Improving graph coloring with lin-ear programmng and genetic algorithms. In: Miettinen, K., Makela, M.M., Toivanen, J. (eds.) Proc. Evolutionary Algorithms in Engineering and Computer Science (EUROGEN), Jyvaskyla, Finland, pp. 113–118. John Wiley & Sons, Chichester (1999)
13. Al-Omari, H., Sabri, K.E.: New graph coloring algorithms. J. Mathematics and Statistics, 739–741 (2006)
14. Croitoru, C., Luchian, H., Gheorghies, O., Apetrei, A.: A new genetic graph coloring heuristic. In: Computational Symphosium on Graph Coloring and its Generalizations, pp. 63–74 (2002)
15. Drechsler, N., Gunther, W., Drechsler, R.: Efficient graph coloring by evolutionary algorithms. In: Reusch, B. (ed.) Proc. Int. Conf. Computational Intelligence, Theory and Applications, pp. 30–39. Springer, London (1999)
16. Galinier, P., Hao, J.K.: Hybrid evolutionary algorithms for graph coloring. J. Combinatorial Optimization 3(4), 379–397 (1999)
17. Han, L., Han, Z.: A novel bi-objective genetic algorithm for the graph coloring problem. In: Proc. Int. Conf. Computer Modeling and Simulation, Sanya, Chaina, pp. 3–6. IEEE Press, India (2010)
18. Huang, F., Chen, G.: A symmetry-breaking approach of the graph coloring problem with gas. In: Miettinen, K., Makela, M.M., Toivanen, J. (eds.) Proc. Int. Conf. Computer Supported Cooperative Work in Design, pp. 717–719. IEEE Press, Xiamen (2004)
19. Kumar, R., Tolay, P., Tiwary, S.: Enhancing solution quality of the biobjective graph coloring problem using hybridization of EA. In: Köppen, M., et al. (eds.) Proc. Genetic and Evolutionary Computation Conference (GECCO), pp. 547–554. ACM Press, Atlanta (2008)
20. Eiben, A.E., van der Hauw, J.K.: Graph coloring with adaptive genetic algorithms. Technical Report TR 96-11, Leiden University (1996)
21. Jazkiewicz, A.: Genetic local search for multiobjective combinatorial optimization. European J. Operational Research 137(1), 50–71 (2002)
22. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation 6(2), 182–197 (2002)
23. Zhang, Q., Li, H.: MOEA/D: a multiobjective evolutionary algorithm based on decomposition. IEEE Transactions on Evolutionary Computation 11(6), 712–731 (2007)
24. Deb, K., Jain, S.: Running performance metrics for evolutionary multiobjective optimization. In: Wang, L., Tan, K.C., Furuhashi, T., Kim, J.H., Yao, X. (eds.) Proc. Simulated Evolution And Learning (SEAL), Orchid Country Club, Singapore, pp. 13–20. ACM Press, New York (2002)
25. Deb, K.: Multi-Objective Optimization using Evolutionary Algorithms. John Wiley & Sons, Chichester (2001)
26. Zitzler, E.: Evolutionary algorithms for multiobjective optimization: methods and applications. PhD thesis, Swiss Federal Institute of Technology Zurich (1999)
27. Knowles, J.D., Corne, D.W.: On metrics for comparing nondominated sets. In: Proc. Congress Evolutionary Computation (CEC), Honolulu, Hawaii, pp. 711–716. IEEE Press, Los Alamitos (2002)

A Framework for Specification and Verification of Timeout Models of Real-Time Systems

Janardan Misra*

EMCSS India Private Limited
Bangalore 560103, India
janardanm@acm.org

Abstract. Timeout based models are an important class of design models for discrete event modeling and simulation of real-time systems and protocols. In this work, we define a framework to graphically represent timeout based models with synchronous communication. The formalism offers system designers an expressive graphical language with well defined semantics to model their system designs and reason about their behavior. For actual implementation, these models are expressed using GraphML standard with support for embedded ANSI C code. We further devise an automated translation technique (and develop corresponding prototype tool support) to translate the GraphML designs into SAL (Symbolic Analysis Laboratory) model specifications, which in turn, can be formally verified using the SAL verification engine.

Keywords: Timeout models, Timeout Transition Diagram, Graphical Specification, GraphML, SAL.

1 Introduction

Timeout based models are an important class of design models for discrete event modeling and simulation of real-time systems. These models are especially useful for modeling time triggered systems and protocols e.g., TTCAN (Time Triggered Controller Area Network), TTP (Time Triggered Protocol), FlexRay etc.

To overcome the complexity of verifying real-time systems with dense time dynamics, [3] proposed timeout and calender based transition models. In these behavioral models, each process in the system possesses a timeout that specifies the time when the next discrete transition of the process would happen, and there is a global data structure, called *calendar*, which stores future events (message delivery) and the time points at which these events are scheduled to occur. Although these transition models are expressive enough to capture a range of behaviors associated with time triggered systems including asynchronous communication delays, they however have specific design limitations: First, these models are not well suited for actual system design purpose since they describe the behavior of the combined system without (explicitly) specifying the design of

* Work done while author was associated with HTS Research, Bangalore 560026.

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 146–157, 2011.

the modular components. Secondly, absence of formally defined syntactic design models corresponding to these transitions systems would demand that additional correctness measures are put in place since for verification purposes actual design models need to be (manually) interpreted and translated into these transition models as per the underlying system dynamics and on discovering an error during verification, such errors need to comprehended by a designer, and subsequently, translated back into the design for a remedial action.

To overcome these design limitations, in [7] we adapted the specification framework of timed transition diagram introduced in [5] to define clockless-timeout and calender based models. In this work we extend it further to formalize timeout based models as timeout transition diagrams and their behavior in terms of transition system semantics. The benefits that we derive from using this formalization are many-fold. Our framework inherits most of the properties of the timed transition system including reachability properties. Most of the techniques, like digitization that can be applied to these timed transition systems are applicable to our formalization also.

Related Work: Because of the fact that the state space of real-time systems with dense time dynamics is uncountable, modeling and verification of them is rather difficult. Many formalisms have been proposed to model and verify real-time systems including timed automata (TA) [1], timed transition models [5], timed process algebras [2], and timed petri-nets [9]. Although timeout based models can be represented using clock based formalisms including timed automata and timed petri-nets, modeling asynchronous communication with delays is generally bit difficult. A natural approach in these formalisms is to model each channel as a separate process with delays as state variables. However with larger systems with high degree of inter-process connectivity, such modeling is difficult, error prone and quickly leads to state space explosion problem. These difficulties necessciate that a separate timeout based expressive specification formalisms should be evolved.

Existing design standards e.g., UML-RT (Real Time Unified Modeling Language) [8], though again can be used for representing timeout based models, absence of formal semantics is a big limitation while applying formal verification techniques on these UML models. For this reason there exist many works on automated translation from the UML designs to some of the existing verification platforms giving platform specific semantics to these UML designs. For example, [6] presents a prototype tool HUGO/RT, which translates Timed UML diagrams into timed automata of UPPAAL and thus verifies whether timed state machines in a UML model interact according to the scenarios specified by time-annotated UML collaboration diagrams, which are translated into observer TAs.

Rest of the paper is organized as following: In Section 2, we present the formalization of the timeout based models including parametric specifications. Section 3 presents a scheme for XML representation of these models. In Section 4, we discuss a scheme for translating the XML representation into a SAL specifications. Section 5 concludes the paper.

2 Formalization of Timeout Based Models

A timeout transition model [3] contains a finite set of timeouts and a global clock variable t. Timeouts define the time points when discrete transitions will be enabled in the future. The clock variable t keeps track of the current time. Interprocess communication delay during message transfers however cannot be modeled using timeout based modeling alone because delays are beyond the control of individual processes. Therefore they are modeled by using the calendar as a global data structure [4]. Transitions in this model are classified into two types: time progress transitions and discrete transitions. In time progress transition, t (time) advances to the minimum of the timeouts, or to the least time point at which a message will be delivered in future, whichever is less. Discrete transition occurs when t is equal to the this minimum value. If there are more than one processes, which have their timeouts equal to the minimum value, one of them is randomly chosen and corresponding discrete transition occurs updating the value of the timeout for the selected process.

In [5] an abstract model of timed transition diagram was proposed which could represent a wide variety of behaviors of the timed execution of concurrent processes. In this section we adapt and extend these timed transition diagrams to represent timeout based models. Further we describe their associated semantics in terms of state transition systems.

2.1 Timeout Based Timed Transition Model

A Timeout based Model (ToM) is represented as

$$P: \ \{\theta\}[P_1||P_2||\ldots||P_n]$$

Each process P_i is a sequential non-deterministic process having τ_i as its local timeout and $\mathcal{X}_i$ as a set of local timing variables. Local timing variables are used for determining the relative delay between events. A shared variable t represents the global clock. The operator "$||$" denotes parallel composition. The formula θ, called the *data pre-condition* of P restricts the initial values of variables in

$$\mathcal{U} = \{t\} \cup \mathcal{T} \cup \mathcal{X} \cup \mathit{Var}$$

where set of all timeouts is $\mathcal{T} = \{\tau_1, \tau_2, \ldots, \tau_n\}$, and $\mathcal{X} = \bigcup_i \mathcal{X}_i$. The set $\mathit{Var} = (G \cup L_1 \cup L_2 \cup \ldots \cup L_n)$ is the set of other state variables assuming values from finite domains. The variables in G are globally shared among all the processes while L_i contains variables local to process P_i. Let f^{Var} be the set of computable functions on Var.

Each process P_i is represented using a *timeout transition diagram* (TTD), which is a finite directed graph with a set of nodes $\mathit{Loc}_i = \{l_0^i, l_1^i, \ldots, l_{m_i}^i\}$, called *locations*. The *entry location* is l_0^i. There are two kinds of edges in the graph of a process P_i: *Timeout edges* and *Synchronous Communication edges*. Edge definitions involve an enabling condition or guard ρ, which is a boolean-valued function or a predicate.

Timeout Edges: A timeout edge $(l^i_j, \rho \ \Rightarrow \langle \tau_i := \mathit{update}_i, \gamma, f\rangle, l^i_k)$ in the graph of the process P_i is represented as

$$l^i_j \overset{\rho\ \Rightarrow\langle \tau_i:=\mathit{update}_i,\gamma,f\rangle}{\longrightarrow} l^i_k,$$

where update_i specifies the way timeout τ_i is to be updated on taking a transition on the edge when the guard ρ evaluates to `true`. $\gamma \subseteq \mathcal{X}_i$ specifies the local timing variables which capture value of the clock t while taking transition on the edge. $f \in f^{Var}$ manipulates the state variables in $G \cup L_i$. update_i is defined using the rule:

$$\mathit{update}_i = k_1 \mid k_2 \mid \infty \mid \max(\mathcal{M})$$

where $l + z \prec k_1 \prec' m + z'$ for $\prec, \prec' \in \{<, \leq\}$ and $k_2 \succ l + z$ for $\succ \in \{>, \geq\}$; $z, z' := t|w$ and $l, m \in \mathbb{N}$ are non negative integer constants specifying the lower and upper limits for a timeout increment interval, and $w \in \mathcal{X}_i$ is a local timing variable. The variable z makes such an interval relative to the occurrence of specific events. $\mathcal{M}$ is the set of all the integer constants that are used to define the upper limit of different timeouts for different processes in the system. $\max(\mathcal{M})$ returns the maximum of all the integers in $\mathcal{M}$.

Constraints on k_1, k_2 specify how new value of timeout τ_i should be determined based upon the current value of the clock t and/or w, which would have captured the value of t in some earlier transition. Setting a time out to ∞ is used to capture the requirement of indefinite waiting for an external signal/event. $\max(\mathcal{M})$ is used to capture the situation where the next discrete transition of a process may happen at any time in the future, for example, the process can be in a sleeping mode and may wake up at any future point of time.

Synchronous Communication Edges: Rendezvous communication between a pair of processes (P_s, P_r) is represented by having an edge pair (e_s, e_r) s.t. $e_s \in P_s$ and $e_r \in P_r$ such that

$$e_s : l^s_j \overset{\rho\ \Rightarrow\langle ch!m,\tau_s:=\mathit{update}_s,\gamma,g\rangle}{\longrightarrow} l^s_k$$

$$e_r : l^r_j \overset{True\ \Rightarrow\langle ch?\bar{m},\tau_i:=\mathit{update}_r,\gamma',h\rangle}{\longrightarrow} l^r_k$$

where ch is the channel name, $m \in L_i$ is the message sent, and $\bar{m} \in L_r$ the message received, and $g, h \in f^{Var}$ are the computable functions for manipulating state variables.

Semantics. With a given ToM P, we associate the following transition system $S_P = (\mathcal{V}, \Sigma, \Sigma_0, \Gamma)$, referred to as *timeout based clocked transition system* (TCTS) where,

1. $\mathcal{V} = \mathcal{U} \cup \{\pi_1, \ldots, \pi_n\}$. Each *control variable* π_i ranges over the set $Loc_i \cup \{\perp\}$. The value of π_i indicates the location of the control for the process P_i and it is $\perp$ (undefined) before the start of the process.
2. Σ is the set of *states*. Every state $\sigma \in \Sigma$ is an interpretation of $\mathcal{V}$, that is, it assigns values to clock variable t, every timeout variable in $\mathcal{T}$, timing variables in $\mathcal{X}$, state variables in Var, and control variables $\pi_1, \ldots, \pi_n$, in their respective domains. For $x \in \mathcal{V}$, let $\sigma(x)$ denotes its value in state σ.

3. $\Sigma_0 \subseteq \Sigma$ is the set of *initial states* such that for every $\sigma_0 \in \Sigma_0$, θ is true in σ_0 and $\sigma_0(\pi_i) =\perp$ for each process P_i.
4. $\Gamma = \Gamma_e \cup \Gamma_+ \cup \Gamma_0 \cup \Gamma_{syn\_comm}$ is the set of *transitions*. Every transition $\nu \in \Gamma$ is a binary relation on Σ defined further as follows:

Entry Transitions: Γ_e, the set of entry transitions contains a transition ν_e^i for every process P_i. In particular, $\forall \sigma_0 \in \Sigma_0$,

$$\nu_e^i \equiv (\sigma_0, \sigma') \in \Gamma_e \Leftrightarrow \begin{cases} 1.\ \forall x \in \mathcal{U} :\ \sigma'(x) = \sigma_0(x) \\ 2.\ \forall \tau \in \mathcal{T} :\ \sigma'(t) \leq \sigma'(\tau) \\ 3.\ \sigma_0(\pi_i) = \perp \ \text{ and } \sigma'(\pi_i) = l_0^i \end{cases}$$

Time Progress Transition: The first kind of edges $\nu_+ \in \Gamma_+$ are those where the global clock is increased to the minimum of all timeouts. In particular,

$$\nu_+ \equiv (\sigma, \sigma') \in \Gamma_+ \Leftrightarrow \begin{cases} 1.\ \sigma(t) < \min\{\sigma(\mathcal{T})\} \\ 2.\ \forall \tau \in \mathcal{T} :\ \sigma'(\tau) = \sigma(\tau) \\ 3.\ \forall x \in \mathcal{X} :\ \sigma'(x) = \sigma(x) \\ 4.\ \forall i :\ \sigma'(\pi_i) =\ \sigma(\pi_i) \\ 5.\ \sigma'(t) = \min\{\sigma(\mathcal{T})\} \end{cases}$$

Timeout Increment Transition: For the second kind of edges $\nu_0^i \in \Gamma_0$ the global clock equals the minimum of timeouts. If there is a timeout edge in the TTD of process P_i, $\nu_0^i \equiv (\sigma, \sigma') \in \Gamma_0$

$$\Leftrightarrow \begin{cases} 1.\ \rho \text{ holds in } \sigma \qquad 2.\ \sigma'(t) = \sigma(t) \\ 3.\ \textbf{If}\, \sigma(\tau_i) = \sigma(t) \\ \qquad\quad \textbf{then}\, \sigma'(\tau_i) = \mathit{update}_i > \sigma(\tau_i) \\ \qquad \textbf{else}\, \sigma'(\tau_i) = \sigma(\tau_i) \\ 4.\ \forall x \in \gamma :\ \sigma'(x) = \sigma(t) \ \text{ and} \\ \quad \forall x \in \mathcal{X} \setminus \gamma :\ \sigma'(x) = \sigma(x) \\ 5.\ \forall v \in G \cup L_i : \sigma'(v) = f(\sigma(v)) \text{ and} \\ \quad \forall v \in \mathit{Var} \setminus (G \cup L_i) :\ \sigma'(v) = \sigma(v) \\ 6.\ \sigma(\pi_i) =\ l_j^i \text{ and } \sigma'(\pi_i) = l_k^i \end{cases}$$

If $\mathit{update}_i = k_1$ s.t. $l + z \prec k_1 \prec m + z'$, then update_i arbitrarily selects a value δ such that $[l + \sigma(z) \prec \delta \prec m + \sigma(z')] \wedge [\delta > \sigma(\tau_i)]$ and returns δ. If $\mathit{update}_i = k_2$ s.t. $k_2 \succ l + z$, then update_i arbitrarily selects a value δ such that $[\delta \succ l + \sigma(z)] \wedge [\delta > \sigma(\tau_i)]$ and returns δ. If $\mathit{update}_i = \infty$, update_i returns the largest possible constant defined as per the design of the system. If $\mathit{update}_i = \max(\mathcal{M})$, update_i nondeterministically selects any integer δ in $[0, M + 1]$, where M is the maximum of all the integers in $\mathcal{M}$ returned by $\max(\mathcal{M})$. The local timing variables $\gamma \subseteq \mathcal{X}_i$ for process P_i are assigned the current value of global clock, however the remaining local timing variables in the system retain their old values before the transition.

Synchronous Communication: For a pair of processes P_s, P_r having edges (e_s, e_r) as defined before, $\nu^{sr}_{syn\_comm} \in \Gamma_{syn\_comm}$ exists such that: $\nu^{sr}_{syn\_comm} \equiv (\sigma, \sigma')$

$$\Leftrightarrow \begin{cases} 1.\ \rho \text{ holds in } \sigma \qquad 2.\ \sigma'(t) = \sigma(t) \\ 3.\ \sigma'(\tau_s) = \mathit{update}_s > \sigma(\tau_s) \text{ and} \\ \quad\ \sigma'(\tau_r) = \mathit{update}_r > \sigma(\tau_r) \\ 4.\ \forall x \in (\gamma \cup \gamma') :\ \sigma'(x) = \sigma(t) \text{ and} \\ \quad\ \forall x \in \mathcal{X} \setminus (\gamma \cup \gamma') :\ \sigma'(x) = \sigma(x) \\ 5.\ \sigma'(\bar{m}) = \sigma(m) \\ 6.\ \forall v \in G \cup L_s : \sigma'(v) = g(\sigma(v)) \\ \quad\ \forall v \in G \cup L_r : \sigma'(v) = h(\sigma(v)) \\ \quad\ \forall v \in \mathit{Var} \setminus (G \cup L_s \cup L_r) :\ \sigma'(v) = \sigma(v) \\ 7.\ \sigma(\pi_s) = l^s_j, \sigma(\pi_r) = l^r_j \text{ and} \\ \quad\ \sigma'(\pi_s) = l^s_k, \sigma'(\pi_r) = l^r_k \end{cases}$$

This semantic model defines the set of possible computations of the ToM P as a (possibly infinite) set of state sequences $\xi :\ \sigma_0 \rightarrow \sigma_1 \rightarrow \ldots$, which start with some initial state σ_0 in Σ_0 and follow consecutive edges in Γ

Models for Time: There are two natural choices for time, the set of non-negative integers $\mathbb{N}$ (discrete time) and the set of non-negative reals $\mathbb{R}$ (dense time). When we consider that the underlying model of time is dense, we need to add the following non-zenoness condition to ensure effective time progress in the model. *There must not be infinitely many time progress (or timeout increment) transitions effective within a finite interval.*

2.2 Modeling Parametric Processes

A completely parametric process family would be specified as $\{\theta\}[\{P(i)\}_{i=1}^{i=N}]$, where $N \geq 1$ is some finite positive integer and $\theta = \theta_1 \wedge \ldots \wedge \theta_N$ such that θ_i $(1 \leq i \leq N)$ initializes the variables for the i^{th} copy of the process. Each process $P(i)$ is a TTD. The semantic interpretation of such parametrically specified process family is given by first flattening the specification as $\{\theta\}[P(1)||\ldots||P(N)]$ and then applying the semantics presented before.

Such parametric specification can be generalized to a homogeneous set of process families as $\{\theta\}[\{P(i_1)\}_{i_1=1}^{i_1=N_1}||\ldots||\{P(i_l)\}_{i_l=1}^{i_l=N_l}]$, where $N_1, \ldots N_l$ are some finite positive integers and $\theta = \theta_{\mathbf{1}} \wedge \ldots \wedge \theta_{\mathbf{l}}$ such that $\theta_{\mathbf{i}} = \theta_{i1} \wedge \ldots \wedge \theta_{iN_i}$ initializes the variables for the i^{th} process family. The term homogeneous arises because processes in all the process families should uniformly be either TTDs or calender based TTDs. We do not consider the case of hetrogeneous set of process families, where processes across different process families might be different. Similar to the case of a single parametric process family, the generalized process family can be interpreted by flattening the process specification.

The basic framework of TTD can be extended with advanced modeling concepts like calendar based asynchronous communication, inter-process scheduling, priorities, and interrupts [7,5].

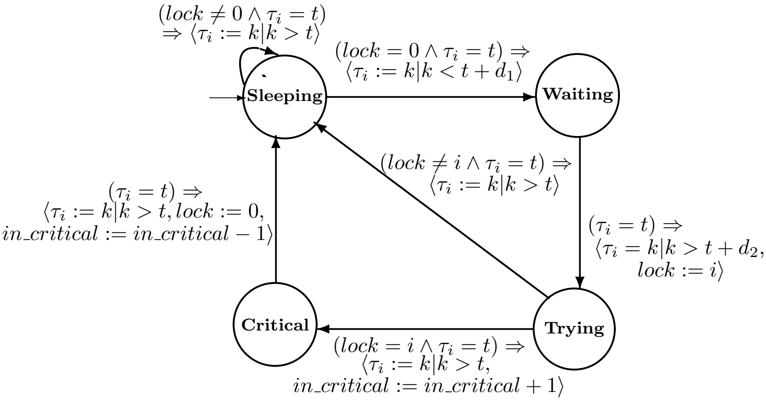

$(lock \neq 0 \wedge \tau_i = t)$
$\Rightarrow \langle \tau_i := k | k > t \rangle$
$(lock = 0 \wedge \tau_i = t) \Rightarrow$
$\langle \tau_i := k | k < t + d_1 \rangle$
Sleeping
Waiting
$(lock \neq i \wedge \tau_i = t) \Rightarrow$
$\langle \tau_i := k | k > t \rangle$
$(\tau_i = t) \Rightarrow$
$\langle \tau_i := k | k > t, lock := 0,$
$in\_critical := in\_critical - 1 \rangle$
$(\tau_i = t) \Rightarrow$
$\langle \tau_i = k | k > t + d_2,$
$lock := i \rangle$
Critical
Trying
$(lock = i \wedge \tau_i = t) \Rightarrow$
$\langle \tau_i := k | k > t,$
$in\_critical := in\_critical + 1 \rangle$

Fig. 1. TTD for the i^{th} process in the Fischer's Protocol

Example: Fisher's Mutual Exclusion Protocol: Fischer's mutual exclusion (mutex) protocol is a well studied protocol to ensure mutual exclusion among concurrent processes $P_1, ...P_n$ trying to access shared resources in a real-time. A process P_i is initially idle (*Sleeping* state), but at any time, may begin executing the protocol provided the value of a global variable *lock* is 0 and moves to *Waiting* state. There it can wait up to maximum of d_1 time units before assigning the value i to *lock* and moving to *Trying* state. It may enter the *Critical* section after a delay of at least of d_2 time units provided the value of *lock* is still i. Otherwise it has to move to *Sleeping* state. Upon leaving the critical section, it re-initializes *lock* to 0. There is another global variable, *in_critical*, used to keep count of the number of processes in the critical section. The auto-increment (auto-decrement) of the variable is done before a process enters the critical section (leaves the critical section).The TTD of the i^{th} process P_i executing Fischer's protocol is shown in Fig. 1.

3 XML Representation for Timeout Based Models

In order to support system designers, the timeout and calender based models of real-time systems need to have standard graphical representation. We choose GraphML, a XML (Extensible Markup Language) based standard for representing the process graphs. GraphML is a widely accepted standard to represent graphs in XML format. Such a representation gives the practical power to the designs since any formalism devoid of graphical representations is of little practical value in industrial settings.

Since certain components in the definition of a ToM representing a complete system are not entirely graphical in nature (e.g., global variable initializations using θ, composition operation $||$ etc.), we need to devise suitable schemes for representing both graphical as well as non graphical components in XML format.

Also the GraphML schema places certain restrictions on the way elements and attributes can be represented. These are discussed next:

- Each TTD for individual processes is assigned a unique name as a value for `graph id` attribute in the GraphML representation.
- Unlike other edges, there is no direct representation for the (pendent) initialization edges (i.e., edges with target as entry locations l_0^i) in GraphML. Therefore we represent these edges in GraphML by introducing dummy nodes as the source nodes for these edges. These dummy nodes are special starting nodes for each process and will be identified by the presence of **$** symbol in their name attribute.
- An edge attribute $\rho \Rightarrow \langle \ldots \rangle$, is represented as a String type attribute for the edge elements in GraphML. The decision was primarily influenced by the fact that no XML standard currently supports algorithmic information, so we can best hope to parse such attribute as a string and process it accordingly.
- While defining an edge attribute resulting form $\rho \Rightarrow \langle \ldots \rangle$, if either of $ch[!/?]msg$ or *update* or f or η are absent, then that is indicated using **@** in the XML representation.
- Predicate ρ can be `TRUE/FALSE` or an ANSI C conditional expression.
- *Update* can be expressed using standard XML syntax: e.g., `τ_i := k | k &lt; t+d_1` for representing $\tau_i := k | k < t + d_1$. $\tau_i = \infty$ has *special representation* as `infinity = 10000, τ_i := infinity`, which allows designer to specify the maximum waiting delay a process should have under this transition for an external signal.
- Edge ids should be of the form source_node::target_node.
- All variables - local vars, global vars, timing vars, timeout vars, input/output vars - must be declared at the starting edges in each process by adding keywords `local::, global::, input::, output::, TIME::, TIMEOUT::` as prefix to the variable declaration. e.g. `input:: int x = 3`. Where input/output type for state variables need to be determined by the designer a priori. Input variables are considered to be observed variables and their values cannot be changed during transitions. On the other hand, output variables are the controlled variables and so their values may change.
- For the starting edges, in the edge attribute: $\rho \Rightarrow \langle \ldots \rangle$, all but the attribute f should to be empty, where the value of f would be the variable declarations and initializations.
- **Parameterized processes** need to be specified using special constructs on the starting edges. A parametric specification would have the following components in it: i) the starting dummy node would be represented as `node id = “$p(i)”`, ii) `paratype:: i:[1..N];` specifies that there are N copies of this process in the system, iii) `process_vars:: type1: var1, ..., typeM: varM` provides the declaration for the *state independent process variables*, and iv) `parainit::` $\texttt{init}_1$ `^` $\texttt{init}_2$ `^ ... ^` $\texttt{init}_N$ specifies the initializations of these process variables for each of the process copies.
- Random assignment to a variable is specified as `type var_name=random(e1, e2);` where `e1, e2` are arithmetic expressions. E.g., `int d=random(2+x,6);`

specifies `d` as a random integer taking values in the range `[2+x, 6]`. Similarly `double d = random(0,infinity);` specifies `d` to be any value greater than 0. Function `random()` can be used in f or in fundef.vgm file.

- Special keywords (reserved): "rho"(ρ), "ch"(channel), "update" (update function), "f"(state variables manipulation function), "eta"(η). These keywords are used to define `key ids` for edges and take string type values.
- ANSI C Syntax with SAL specific extensions will be used in:
  1. A special file *fundef.vgm* consisting of variable declarations with constant initialization (like global vars in C), user defined complex data type declarations, and function definitions.
  2. Starting edges: Variable declarations. E.g. input::int x = 3; local::double y = g(x); Everything defined in the file *fundef.vgm* is treated global at this step. Variables declared at the starting edges are considered global for the rest of the edges in the TTD.
  3. Other edges in TTD: Usual assignment statements of the form l.h.s = r.h.s (e.g. x = g(x)) and statements like x++. Expressions using variables and functions defined in 1. and 2..
  4. Defining ρ: Conditional C expression.

4 Translating Timeout Based Models into SAL

Given a GraphML representation of a ToM, we next discuss how it can be translated into a SAL modeling language specification. The SAL models can in turn be used for automated formal verification for uncovering design flaws or providing a proof for formal correctness of the design.

Rules To Convert GraphML Elements into SAL

R1. *XML Filename* will be used as Context name in SAL. E.g., in Fisher's Mutex example it would be `mutex:CONTEXT`

R2. Process name appearing as a value of the attribute `graph id` will be used to define the *module* (name) in SAL. E.g., `Train: MODULE =`

R3. Collect all the states of a process as values for the node attribute `node id` except the dummy node and enumerate them to a variable `PROCESS_STATE = {...}`.

R4. By traversing all the edges, collect the messages sent/received and channel names. Enumerate the messages to a global variable `SIGNAL = {...}`.

R5. Define each communication *channel* for a process as a variable of type `SIGNAL` and associate global flag: `channel_flag` of type `BOOLEAN` with initialization as `FALSE`.

R6. Define `process_state` of type `PROCESS_STATE` as a local variable in each process to store the current state. Initialize it with a state where starting edge is pointing.

R7. The information on all *input, output,* and *local* variables types and initialization of a process is extracted from the starting edge in each process's graph. E.g., from the i^{th} process graph in the Fisher's mutex protocol: `INPUT` time: REAL, OUTPUT T_i: REAL, LOCAL p_state: p_STATE, INITIALIZATION p_state = sleeping; lock = 0; T_i IN {z: REAL | z > time }.

R8. All *global* variable (except flags) type and initialization are extracted from the special file ***fundef.vgm***.

R9. A random variable specified as type var_name = random(e1,e2); is correspondingly translated into SAL. E.g., d = random(2+x, 6); is translated into d :{k:INTEGER | k > x+2 AND k < 6};

R10. Each edge in the GraphML file is a transition and will be used to name the end node of that edge. E.g., n::m will be give rise to a transition in the SAL model as n_m.

R11. For each message to be sent, enabling conditions to be extracted from the definition of *rho*.

R12. For each message to be received, enabling conditions become of the type: process_state = l_j AND channel = msg AND channel_flag = true.

R13. Things to be updated for a transition: E.g., transition $(l_j,\ \rho \Rightarrow \langle\ ch!m,\text{ă}timeout := k\ |t \leq k \leq t+5,\ \eta, f\rangle,\ l_k)$ would give rise to:

```
process_state = l_j AND Tr(rho) -> process_state’ = l_k  ch’ = m  ch_flag’ = true
timeout’ IN {x:TIME|(clock<x)AND(x≤clock+5)}  FORALL x in eta x’ = t  Tr(f)
```

Where Tr(rho) and Tr(f) represent the translation of rho and f as per the rules discussed next.

R14. ***Parameterized processes:*** For each parameterized process, we define specification of corresponding SAL module as follows:

```
PARA_VALUE: INTEGER = N; PARATYPE: TYPE = [1.. PARA_VALUE];
p[i: PARATYPE, var1: type1, ..., varM: typeM]: MODULE =
```

where N is the parameter appearing in paratype:: i:[1..N]; for the process specified by node id = “$p(i)" and variable declarations come from process_vars:: type1: var1, type2: var2, ..., typeM: varM. Later during module composition these variables are initialized as per the rules appearing in parainit:: init_1 ^ init_2 ^ ... ^ init_N.

Rules To Convert ANSI C Expressions into SAL: In addition to translating the GraphML elements into SAL, we need to translate the algorithmic information on the edges represented using the C syntax. For some representative elements of the ANSI C, SAL correspondence are given next. However some elements of ANSI C language are not supported including pointers, void, static, and extern quantifiers.

- int c ⇒ c: INTEGER. float c ⇒ c: REAL.
- struct stu {int age; int height;}; ⇒ stu: TYPE = [# age: INTEGER, height: INTEGER #]
- typedef int number; ⇒ number : TYPE = INTEGER
- int x[5] ⇒ x : ARRAY [0..4] OF INTEGER
- for(j = 0; j < 10; j++){size[j] = j;} ⇒ (FORALL (j: [0..9]): size WITH [j]:= j);
- Local variable declarations: int c = a+b; ⇒ LET c: INTEGER = a+b
- Macro Definition: # define n 8 ⇒ n: {8}
- Function: double Max(int s1, int s2) {...} ⇒ Max(s1:INTEGER, s2:INTEGER): REAL= ...

A function definition consisting of a sequence of expressions needs to be collapsed into single compound expression. [8] discusses a technique of reducing arbitrary imperative programs into functional programs using their denotational semantics. Applying this technique a function definition could be translated into an equivalent functional program consisting of only one compound statement, which can then be translated into SAL. However the only formal parameter or global variable, which could be manipulated, are the ones which would be returned by the function.

ToM to SAL Translation Algorithm: Finally based upon the above rules for translating the GraphML elements and ANSI C expressions into SAL specifications, we define the following overall translation algorithm:

Step1. Apply R1. Find out all the messages used in all the processes and use R4.

Step2. Find out the number of processes (excluding the special process) and define the following:

```
N_TIMEOUT: NATURAL= [number of processes];
TIMEOUT_INDEX: TYPE = [0..N_TIMEOUT];
TIMEOUT_ARRAY: TYPE = ARRAY TIMEOUT_INDEX OF type of time;
```

Step3. Translate all the user defined type definitions, function definitions, and other global variables defined in the file `fundef.vgm` by applying the translation rules given later.

Step4. Print the function to calculate minimum of the timeouts:

```
recur_min(x:TIMEOUT_ARRAY, min_sofar: REAL, idx: [0..N_TIMEOUT] ): REAL =
   IF idx = 0 THEN  min_sofar
   ELSE  recur_min(x,min(min_sofar, x[idx]), idx-1) ENDIF;
min(x : TIMEOUT_ARRAY): REAL = recur_min(x, x[N_TIMEOUT], N_TIMEOUT-1);
```

Step5. Print the clock module, which advances the clock to the next minimum timeout. `% Assuming that there are` n `different channels for message communication`:

```
clock : MODULE =
BEGIN
INPUT timeout:TIMEOUT_ARRAY  OUTPUT time:REAL  GLOBAL ch1_flag, ..., chn_flag:BOOLEAN
INITIALIZATION time = 0
TRANSITION
[time_elapses:time < min(timeout) AND (ch1_flag) ... AND (chn_flag)-> time' =
 min(timeout)]
END;
```

For each individual process repeat **Step6 - Step9**:

Step6. Apply R2 for the case of non-parametric process and R14 for parametric process to define SAL module corresponding to the process.

Step7. Extract all the type definitions mentioned on the starting edge of the process and use R3.

Step8. Extract input, output, global and local variables type and initialize them using R5 - R8.

Step9. Translate transition for each edge using R10 - R13.

Step10. Print the module (`composition: MODULE`) with output timeout of type `TIMEOUT_ARRAY` and do the asynchronous composition of all the processes by renaming the local timeout variables as elements of the timeout array.

```
([] (i: PARATYPE_1): (RENAME T_i TO timeout[i] IN p[i]));
```

OR (with initializations)

```
(RENAME T_1 TO timeout[1] IN p[1,init_1]) ...[](RENAME T_N TO timeout[N] IN p[N,init_N]);
```

Step11. Finally define the main module: `system: MODULE = clock [] composition;`

5 Concluding Remarks

This work defines a framework to represent graphical designs of timeout models of real-time systems with synchronous and asynchronous communication with delays. We further discussed a scheme for automated translation from XML based representation of the ToM (both parameterized and non parameterized models) into SAL specifications. The scheme uses ANSI C syntax to express computable operations on state variables (with some exceptions including pointers) involving user defined data types and functions.

Some relevant pointers for future work include extending the translation for models with asynchronous communication, multiprogramming, priorities, and interrupts and reverse translation of the SAL outputs (error traces) into input TTD models, thus enabling, iterative design refinement and verification support.

References

1. Alur, R., Dill, D.: A Theory of Timed Automata. Theoretical Computer Science 126(2), 183–235 (1994)
2. Baeten, J., Bergstra, J.: Real Time Process Algebra. Formal Aspects of Computing 3(2), 142–188 (1991)
3. Dutertre, B., Sorea, M.: Timed systems in SAL. Technical report, Computer Science Laboratory, SRI International, Menlo Park, CA (2004)
4. Dutertre, B., Sorea, M.: Modeling and verification of a fault-tolerant real-time startup protocol using calendar automata. In: Lakhnech, Y., Yovine, S. (eds.) FORMATS 2004 and FTRTFT 2004. LNCS, vol. 3253, pp. 199–214. Springer, Heidelberg (2004)
5. Henzinger, T.A., Manna, Z., Pnueli, A.: Timed transition systems. In: Huizing, C., de Bakker, J.W., Rozenberg, G., de Roever, W.-P. (eds.) REX 1991. LNCS, vol. 600, pp. 226–251. Springer, Heidelberg (1992)
6. Knapp, A., Merz, S., Rauh, C.: Model checking - timed UML state machines and collaborations. In: Damm, W., Olderog, E.-R. (eds.) FTRTFT 2002. LNCS, vol. 2469, pp. 395–416. Springer, Heidelberg (2002)
7. Saha, I., Misra, J., Roy, S.: Timeout and calendar based finite state modeling and verification of real-time systems. In: Namjoshi, K.S., Yoneda, T., Higashino, T., Okamura, Y. (eds.) ATVA 2007. LNCS, vol. 4762, pp. 284–299. Springer, Heidelberg (2007)
8. Selic, B.: Using UML for modeling complex real-time systems. In: Müller, F., Bestavros, A. (eds.) LCTES 1998. LNCS, vol. 1474, pp. 250–260. Springer, Heidelberg (1998)
9. Wang, J.: Timed Petri nets: Theory and application. Kluwer Academic Publishers, Dordrecht (1998)

Enhancing the Local Exploration Capabilities of Artificial Bee Colony Using Low Discrepancy Sobol Sequence

Tentu Monica[1], Anguluri Rajasekhar[2], Millie Pant[3], and Ajith Abraham[4]

[1] Dept of Information Technology, Sri Sai Jhyothi Engineering College,
Hyderabad, India
mounika.ssj@gmail.com

[2] Dept of Electrical and Electronics Engineering, National Institute of Technology,
Warangal, India
rajasekhar.anguluri@ieee.org

[3] Dept of Paper and Pulp Technology, Indian Institute of Technology,
Roorkee, India
millifpt@iitr.ernet.in

[4] Director of Machine Intelligence Research Laboratories (MIR Labs), USA
ajith.abraham@ieee.org

Abstract. In this paper we propose a mechanism for enhancing the performance of the Artificial Bee Colony Algorithm (ABCA) by making use low discrepancy Sobol sequence. The performance of the proposed Sobol sequence guided ABC (S-ABC) is analyzed over several benchmark functions and also compared to that of basic ABC. The empirical results show that the presence of low discrepancy sequence like that of Sobol, significantly improves the performance of the basic ABCA.

Keywords: artificial bees; quasi random sequences; continuous functions, global optimization.

1 Introduction

With the growing complexities of real world application problems, researchers are concentrating on general purpose algorithms that can be applied to a wide and diverse range of problems efficiently. Some of the algorithms that have gained popularity in the last few decades because of their wide applicability and simple structure include Genetic Algorithms (GA) [1], Particle Swarm Optimization [2], Differential Evolution (DE) [3] etc. A latest addition to the family of such algorithms is Artificial Bee Colony (ABC) algorithm, which is a swarm intelligence algorithm based on the foraging behavior of honey bees.

ABC was proposed by Karaboga and Bastruk for optimizing various numerical functions [4, 5] in 2005. Due to its simplicity and robustness it has been successfully applied to various practical optimization problems like Clustering [6]; IIR filter design [7]; extraction of MESFET [8] and some of the modifications for improvising the performance are provided in [9, 10, 11, 12, 13];

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 158–168, 2011.

In basic ABC, each virtual bee updates its trajectory towards the nectar, based on the information provided by onlooker bees via *waggle dance*. This is done by roulette wheel selection criterion. Finally, a '*scout*' is employed so that there is no scope of local optimum. In literature it had been proved through various instances that ABC provides promising solutions in less computational time when compared to traditional methods like GAs and other soft computing tools, but the quality of solution don't improve at the higher iteration count and remains stagnated causing premature convergence leading to an intermediate optimal solution. This drawback is mainly due to the lack of a proper path which drives the colony of bees to converge to a solution found (for a pre-determined termination criterion) which may not be a global solution. Thus a mechanism is needed to enhance the local exploration capabilities of ABC.

In this work we propose the use of a quasi-random, Sobol sequence to guide the movement of scout bees. Although, low discrepancy sequences like that of Sobol have been applied to various general purpose algorithms [14, 15], where it was observed that the use of these sequences improve the performance of algorithms quite appreciably. However, to the best of our knowledge, the effect of applying low discrepancy sequences to ABC is still unexploited.

The remaining orientation of the paper is as follows; Section 2 gives a brief overview of Quasi Random Sequences (QRS) and Sobol Sequence. Section 3 describes the basic structure of Artificial Bee Colony Algorithm. Section 4 describes the proposed algorithm. Section 5 gives the experimental settings and numerical results of some standard unconstrained benchmark problems. Finally we provided some conclusions towards end.

2 Quasi Random Sequences (QRS)

QRS or low discrepancy sequences are less random than pseudo random number sequences but are more useful for computational methods, which depend on the production of random numbers. Some of these tasks involve approximation of integrals in higher dimension, simulation and global optimization. Some well-known QRS are Vander Corput, Sobol, Halton and Faure. Any of these sequences can be used to generate random numbers.

The 3rd and the 4th author have applied Sobol sequence in PSO algorithm [14] and observed that its inclusion helps in improving the performance of basic PSO. Encouraged by its success, in the present study, the authors have used it in the basic ABC algorithm.

Most of the programming languages have an inbuilt function code for generating random number (for example in C++ the inbuilt function is rand()). Computer generated random numbers follow uniform distribution. QRS on the other hand, are said to be better than pseudo random sequences, because they follow a particular pattern and hence are able to cover the search space more evenly in comparison to pseudo random sequences. This can be seen easily from Fig 1(a) and Fig 1(b), which shows the distribution of random numbers using computer generated random numbers and random numbers generated using QR Sobol sequence.

Before proceeding further on to the main algorithm we will discuss some definitions related to QSR in order to get their better understanding.

2.1 Discrepancy of a Sequence

Mathematically, discrepancy of a sequence is the measure of its uniformity, It is computed by comparing the actual number of sample points in a given volume of a multi-dimensional space with the number of sample points that should be there assuming a uniform distribution defined.

For a defined set of points $x^1, x^2, ..., x^N \in I^S$ and a subset of $G \subset I^S$, define a counting function $S_N(G)$ as the number of points $x^i \in \mathrm{G}$. For each $x = (x_1, x_2, ..., x_S) \in I^S$ let G_x be the rectangular **S** dimensional region, such that $G_x = [0, x_1) \text{ x } [0, x_2) \text{ x....x } [0, x_S)$, with volume $\mathrm{x}_1\mathrm{x}_2...\mathrm{x}_\mathrm{N}$. Then the discrepancy of points is given by $D_N^*(x^1, x^2, x^3.....x^N) = \mathrm{Sup} \mid \mathrm{S_N(G_x)} - \mathrm{N}x_1x_2....x_S \mid \; x \in I^S$

2.2 Construction of Low-Discrepancy Sobol Sequence

The Sobol sequence is the most widely deployed low-discrepancy sequence, and is used for calculating multi-dimensional integrals and in quasi-Monte Carlo simulation. The construction of the Sobol sequence [16] uses linear recurrence relations over the finite field, F2, where F2= {0, 1}. Let the binary expansion of the non-negative integer n be given by $n = n_1 2^0 + n_2 2^1 + ... + n_w 2^{w-1.}$. Then the n^{th} element of the j^{th} dimension of the Sobol sequence $X_n^{(j)}$ can be generated by:

$$X_n^{(j)} = n_1 v_1^{(j)} \oplus n_2 v_2^{(j)} .....n_w v_w^{(j)}$$

Where $v_1^{(j)}$ is a binary fraction called the i^{th} direction number in the j^{th} dimension. These direction numbers are generated by the following q-term recurrence relation.

$$v_i^{(j)} = a_1 v_{i-1}^{(j)} \oplus a_2 v_{i-2}^{(j)} \oplus ... \oplus a_q v_{i-q+1}^{(j)} \oplus v_{i-q}^{(j)} \oplus (v_{i-q}^{(j)} / 2^q)$$

We have $i > q$ and the bit a_i comes from the coefficients degree-q primitive polynomial over F2. The distribution of sample points in space using computer generated pseudo random numbers and sample points generated using quasi random Sobol sequence are shown in Fig 1(a) and Fig (b) respectively. From the figures it is very clear that the sample points generated using the Sobol sequence are able to cover the search space more evenly in comparison to the computer generated random numbers.

3 Artificial Bee Colony Algorithm (ABCA)

ABCA is a swarm intelligent optimization algorithm inspired by honey bee foraging behavior. It classifies the foraging artificial bees into three groups namely *employed bees, onlooker bees* and *scouts*. The first half colony consists of the *employed bees* and second half consists of *onlooker bees*. A bee that is currently searching for food or exploiting a food source is called an *employed bee*. A bee waiting in the hive for making decision to choose a food source is named as an *onlooker*. For every food source, there is only one employed bee and the employed bee of abandoned food source becomes scout. In ABC algorithm, each solution to the problem is considered as *food source* and represented by a D-dimensional real-valued vector, where the fitness of the solution corresponds to the *nectar amount* of associated food resource. ABCA is an iterative process like most of its contemporary search algorithms.

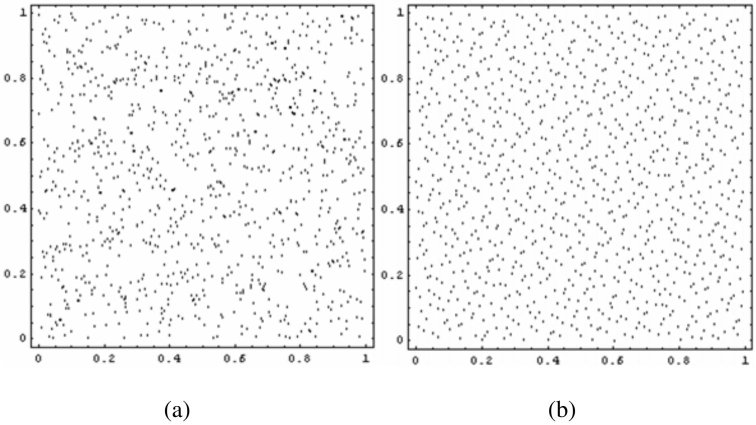


Fig. 1. (a) and Fig 1(b). Sample points of pseudo Random and Quasi Random Sequences respectively (reproduced from [14])

The algorithm starts by initializing all *employed bees* with randomly generated food sources (solutions). For each iteration every employed bee finds a food source in the neighborhood of its current food source and evaluates its nectar amount i.e., (*fitness*). In general the position of i_{th} food source is represented as $X_i = (x_{i1}, x_{i2}, ..., x_{iD})$. After returning to the hive information is shared by the employed bees, *onlooker bees* go to the region of food source explored by employed bees at X_i based on the probability p_i defined as

$$p_i = \frac{fit_i}{\sum_{k=1}^{FS} fit_k} \tag{1}$$

Where FS is total number of food sources. Fitness value fit_i is calculated by using following equation.

$$fit_i = \frac{1}{1+f(X_i)} \tag{2}$$

Here $f(X_i)$ is the objective function considered. The onlooker finds its food source in the region of X_i by using following equation

$$x_{new} = x_{ij} + r*(x_{ij} - x_{kj}) \tag{3}$$

Where x_{new} is the new food source exploited by onlooker and *k* is a solution in the neighborhood of *i, r* is a random number in the range [-1, 1] and *j* is the dimension of the problem considered.

If the obtained new fitness value is better than the fitness value achieved so far, than the bee moves to this new food source leaving this old one otherwise it retains its old food source. When all employed bees have completed this process, the information is shared with onlookers. Each of the onlookers selects food source according to probability given above. By this scheme good sources are well accommodated with onlookers. Each bee will search for a better food source for a certain number of cycles (*limit)*, and if the fitness value doesn't improve then that particular bee becomes scout bee. The food source is initialized to that *scout bee* randomly.

Pseudo code for ABC Algorithm

1. Initialization
2. Move the employed bees onto their food sources and evaluate their nectar amounts.
3. Place the onlookers depending upon the nectar amounts obtained by employed bees
4. Send the scouts for exploring new food sources.
5. Memorize the best food sources obtained so far.
6. If a termination criterion is not satisfied, go to step 2; otherwise stop the procedure and display the best food source obtained so far

4 Sobol Sequence Guided Artificial Bee Colony Algorithm (S-ABC)

The proposed S-ABC algorithm is an extension to the original ABCA by including the concept of Sobol sequence in it. The main point which differentiates between the basic ABCA and S-ABCA is that in ABCA, the scout is assigned a random food location (please note that scout bee is the one which does not show any improvement in fitness value for a certain number of cycles), while in S-ABCA the initialization of food source to the scout is done with the help of Sobol sequence.

The quasi random numbers generated by using Sobol Sequence are used in the SOBOL SEQUENCE GENERATOR (SSG) operator. Now, instead of moving randomly, the scout moves systematically and is able to explore the search space more efficiently. This exploration helps ABC in detecting the food sources (indicating the fitness solutions) effectively.

The SSG operator is defined as [14]:
$SSG = R_1 + (R_2 / \ln R_1)$
Where R_1 and R_2 are random numbers generated by sobol sequences.

Pseudo code for ABC Algorithm

```
Step1.  Initialize the population of solution $x_{ij}, i = 1,2,...FS$,
    $j = 1,2,...D$, $trail_i = 0$ ; trail is the non-improvement number of the
    solution abandonment of food source via trial (limit).
Step2.  Evaluate the population
Step3.  Cycle=1
Step4.  REPEAT
        {----Produce new food source population for employed bee-----}
Step5.  For $i=1$ to $FS$ do
    i.    Produce a new food source $v_i$ for the employed bee of the food
          source $x_i$ by using (3)
    ii.   Apply a greedy selection process between $v_i$ and $x_i$ and then select
          the better one
    iii.  If solution $x_i$ doesn't improve $trail_i = trail_i + 1$, otherwise $trail_i = 0$;
          End for
Step6.  Calculate the probability values $p_i$ by Eqn (1) for the solutions
    using fitness values.
          {----Produce new food source population for onlooker bee-----}
          t=0;
          i=1;
Step7.  REPEAT
    If $rand < P_i$ then
    i.    Produce a new food source $v_i$ for the employed bee of the food
          source $x_i$ by using (3)
    ii.   Apply a greedy selection process between $v_i$ and $x_i$ then select the
          better one
    iii.  If solution $x_i$ doesn't improve $trail_i = trail_i + 1$, otherwise
          $trail_i = 0$;
    iv.   t=t+1
    End if
```

```
        UNTIL (t=FS)
    {--------Determine Scout---------}
Step8.  If $max(trail)>limit$ then
    i.  Replace $x_i$ with a new solution produced using quasi random sequence
        For d = 1 to dimension D
                $temp_d$ = Sobolrand( )
        //Sobolrand() is a random number generated by Sobol sequence//
        End For
        $x_{ij} = temp_d$
        End If
    ii. Memorize the best solution achieved so far
          $Cycle=Cycle+1;$
        UNTIL (Cycle=Maximum Cycle Number)
```

5 Standard Benchmarks and Results

In the present study we had taken 6 benchmark problems (Table 1). And all the test problems are scalable in nature and except for f_1 (sphere function), each function is multimodal in nature.

Each function is tested for dimension 10, 50, 100 and the size of bee colony is set to 20 for all the dimensions. The maximum number of generations is set as 500, 2000 and 3000 corresponding to the dimension 10, 50 and 100 respectively. Maximum of 25 best runs for each experimental setting are conducted and the average fitness of the best solutions throughout the run is recorded.

The numerical results for different benchmark problems are recorded in Tables 3 – 5 for dimensions 10, 50 and 100 respectively. If we observe these Tables we can clearly see that by using Sobol sequence to guide the scout bee, we can improve the performance of basic ABCA algorithm, for dimensions 10, 50 and 100. In fact, the performance of S-ABC is much better than basic ABCA for high dimension like that of 100. This shows that S-ABC is able to solve problems of higher dimensions (like 100) as easily as it solves the problems of lower dimension (like 10).

Table 1. Parameter Settings of ABC

Parameter	**Value**
No of bees (NB)	20
Food Sources (FS)	NB/2
Employed bees (n_e)	*50 % of total bees*
Onlooker bees	*50 % of total bees*
Scout	1
Limit	$(n_e * D)$
Bee-scanner	10
Scale parameter γ	0.1

Table 2. Description of Benchmark Functions [17]

Function	**Range of Search**	**Optimum**
Sphere	(-100, 100)	$f_1(0)=0$
Rosenbrock	(-100, 100)	$f_2(0)=0$
Rastrigin	(-5.12, 5.12)	$f_3(0)=0$
Griewank	(-600, 600)	$f_4(0)=0$
Ackley	(-32, 32)	$f_5(0)=0$
Schwefel	(-500, 500)	$f_6(0)=0$

Table 3. Comparison of ABC and S-ABC in terms of Error, Standard and fitness (Dim=10)

Fun	Fitness (Dim=10)		Error and (standard deviation)	
	ABC	S-ABC	ABC	S-ABC
f_1	6.96613E-016	**7.87847E-017**	2.77454E-016 (1.16624E-016)	**1.4081E-016** **(6.51012E-017)**
f_2	6.86374E+00	**1.07752E+00**	1.51599E+001 (8.93638E+00)	**7.37742E+00** **(2.27918E-001)**
f_3	2.84217E-014	**0**	8.12595E-008 (2.41958E-007)	**1.23609E-010** **(3.90886E-010)**
f_4	3.82397E-010	**0**	1.13837E-002 (9.03497E-003)	**4.44089E-017** **(5.73317E-017)**
f_5	8.53568E-010	**1.74791E-013**	7.08422E-009 (7.11042E-009)	**1.91331E-012** **(2.17882E-012)**
f_6	1.59379E-003	**1.27275E-004**	8.35989E+001 (9.84111E+001)	**2.36906E+001** **(4.99365E+001)**

Table 4. Comparison of ABC and S-ABC in terms of Error, Standard and fitness (Dim=50)

Fun	Fitness (Dim=50)		Error and (standard deviation)	
	ABC	S-ABC	ABC	S-ABC
f_1	3.0736E-014	**9.38831E-016**	5.66491E-012 (7.22051E-012)	**1.39918E-015** **(3.41194E-016)**
f_2	4.65730E+001	**1.99201E+00**	4.71152E+001 (2.87453E+01)	**2.77842E+001** **(2.79576E-001)**
f_3	1.12777E-003	**1.64845E-011**	1.16177E+00 (6.64121E-01)	**1.07070E-001** **(3.12879E-001)**
f_4	3.03109E-011	**2.22044E-016**	4.16361E-003 (8.77758E-003)	**5.66214E-016** **(2.47977E-016)**
f_5	1.15782E-006	**1.03473E-009**	5.8194E-006 (5.55291E-06)	**3.01690E-009** **(1.38824E-009)**
f_6	1.97656E+003	**3.55300E+002**	1.43142E+003 (7.05152E+02)	**7.71971E+002** **(2.84908E+002)**

Table 5. Comparison of ABC and S-ABC in terms of Error, Standard and fitness (Dim=100)

Fun	Fitness (Dim=100)		Error and (standard deviation)	
	ABC	S-ABC	ABC	S-ABC
f_1	4.7862E-009	**3.76130E-012**	7.59245E-008 (7.0909E-008)	**1.63246E-011** **(1.04497E-011)**
f_2	5.4879E+001	**9.58502E+00**	1.0069E+002 (2.9026E+001)	**9.68073E+01** **(3.83279E-001)**
f_3	6.28160E+00	**2.21558E-006**	9.79577E+00 (2.74344E+001)	**1.59070E+00** **(1.34787E+001)**
f_4	1.00884E-007	**5.35132E-014**	1.24815E-003 (3.89934E-003)	**2.68802E-012** **(2.47904E-012)**
f_5	1.38373E-004	**7.71599E-007**	7.95947E-004 (6.37673E-004)	**2.52222E-006** **(1.66672E-006)**
f_6	2.38494E+003	**1.97656E+003**	3.97203E+003 (3.38619E+003)	**2.55821E+003** **(3.81167E+002)**

5.1 Convergence of Standard Benchmark Functions towards Optimum

The performance of the proposed S-ABC vis-à-vis basic ABCA is further shown with the help of convergence graphs illustrated in Fig 2 – Fig 7. These graphs clearly show the superior performance of S-ABC in comparison to basic ABCA. For all the test functions the proposed S-ABCA converged faster than the basic ABCA.

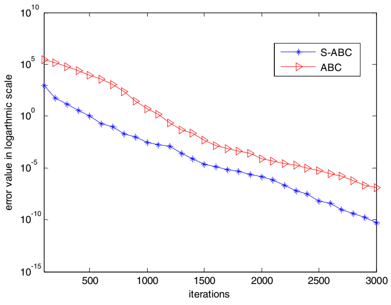


Fig. 2. Sphere function (Dim=100)

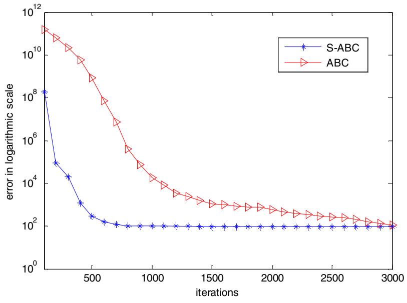


Fig. 3. Rosenbrock function (Dim=100)

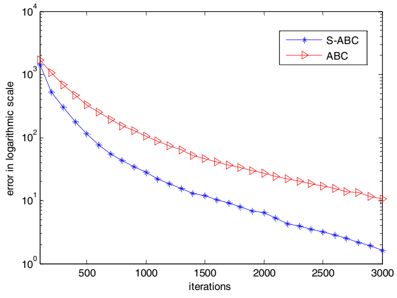


Fig. 4. Rastrigin Function (Dim=100)

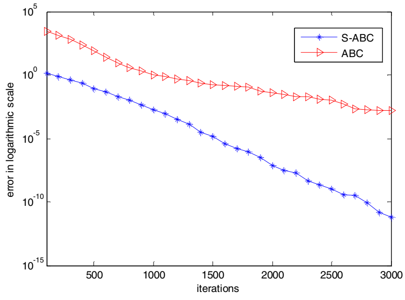


Fig. 5. Griewank function (Dim=100)

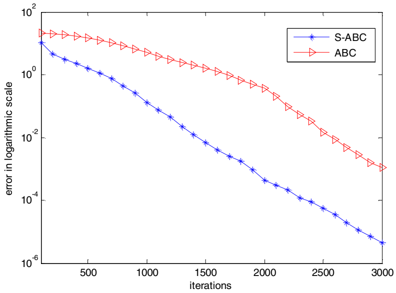


Fig. 6. Ackley Function (Dim=100)

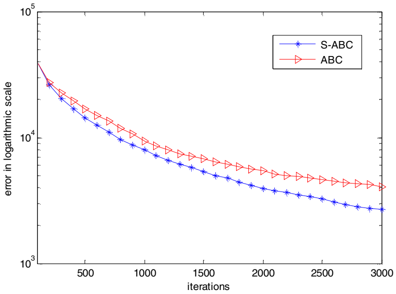


Fig. 7. Schwefel function (Dim=100)

6 Conclusions

In the present study we proposed the application of a low discrepancy Sobol sequence to guide the movement of scout bee in an ABCA. Use of a QRS or low discrepancy sequence to population based search algorithms like that of ABCA is particularly interesting because the working of these algorithms depend largely on random number numbers. Empirical analysis of the proposed S-ABC and its comparison with the basic ABCA indicates that the use of low discrepancy Sobol sequence improves the convergence rate besides maintaining the solution quality. The present work can be extended in several directions. In future, we plan to apply other low discrepancy sequences and compare it with the Sobol sequence. Also we are working on extending it for constrained optimization problems as well.

References

1. Goldberg, D.: Genetic Algorithms in Search Optimization and Machine Learning. Addison Wesley Publishing Company, Reading (1986)
2. Kennedy, J., Eberhart, R.C.: Particle Swarm Optimization. In: Proc IEEE International Conference on Neural Networks, Perth, Australia, pp. 1942–1948. IEEE Service Center, Piscataway (1995)
3. Price, K., Storn, R.: Differential Evolution – A Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Spaces. Technical Report, International computer Science Institute, Berkley (1995)
4. Karaboga, D., Basturk, B.: A powerful and efficient Algorithm for Numerical Optimization: Artificial bee Colony (ABC) Algorithm. J. Global. Optim. 39, 459–472 (2007)
5. Karaboga, D., Basturk, B.: On the Performance of Artificial bee Colony (ABC) Algorithm. Applied Soft Computing 8, 687–697 (2008)
6. Karaboga, D., Basturk, B.: A Novel Clustering Approach: Artificial bee Colony (ABC) Algorithm. Applied Soft Computing 11, 652–657 (2011)
7. Karaboga, N.: A new design method based on artificial bee colony algorithm for digital IIR filters. J. Franklin Inst. 346, 328–348

8. Sabat, S., Udgata, S.K., Abraham, A.: Artificial bee Colony Algorithm for Small Signal Model Parameter Extraction of MESFET. Engineering Applications of Artificial Intelligence (2010)
9. Zhu, G., Kwong, S.: Gbest-guided Artificial Bee Colony Algorithm for Numerical Function Optimization, pp. 3166–3173
10. Drias, H., Sadeg, S., Yahi, S.: Cooperative bees swarm for solving the maximum weighted satisfiability problem. In: Cabestany, J., Prieto, A.G., Sandoval, F. (eds.) IWANN 2005. LNCS, vol. 3512, pp. 318–325. Springer, Heidelberg (2005)
11. Sundarewsaran, K., Sreedevi, V.T.: Development of novel optimization procedure based on honey bee foraging behavior. In: IEEE International Conference on Systems, Man and Cybernatics (2008)
12. Akay, B., Karaboga, D.: A modified Artificial Bee Colony Algorithm for Real-Parameter Optimization. Information Sciences (2010), doi:10.1016/j.ins.2010.07.015
13. Abraham, A., Jatoth, R.K., Rajasekhar, A.: Hybrid Differential Artificial Bee Colony Algorithm. Journal of computational and Theoretical Nano Science, USA (2011) (accepted)
14. Pant, M., Thangaraj, R., Grosan, C., Abraham, A.: Improved Particle Swarm Optimization with Low-Discrepancy Sequences. In: Proc. IEEE Congress on Evolutionary Computation (2008)
15. Nguyen, X.H., Nguyen, Q., Uy., M.R.I., Tuan, P.M.: Initializing PSO with Randomized Low-Discrepancy Sequences: The Comparative Results. In: Proc IEEE Congress on Evolutionary Algorithms, pp. 1985–1992 (2007)
16. Chi, H., Beerli, P., Evans, D.W., Mascagni, M.: On the scrambled soboĺ sequence. In: Sunderam, V.S., van Albada, G.D., Sloot, P.M.A., Dongarra, J. (eds.) ICCS 2005. LNCS, vol. 3516, pp. 775–782. Springer, Heidelberg (2005)
17. Yao, X., Liu, Y., Lin, G.: Evolutionary Programming Made Faster. IEEE Trans. Evol. Compt. 3, 82–102 (1999)

Property Analysis and Enhancement in Recombination Operator of Edge-Set Encoding for Spanning Tree

P.K. Singh and Abhishek Vaid

Computational Intelligence and Data Mining Research Laboratory
ABV-Indian Institute of Information Technology and Management Gwalior, India
pksingh@iiitm.ac.in, vaid.abhi@gmail.com

Abstract. The spanning tree problem is a well-studied problem and Evolutionary Algorithms (EAs) have been successfully applied to a large variants of the spanning tree problem. The behavior of an evolutionary algorithm depends on the interaction between the encoding and the genetic operators that act on that encoding. Various encodings and operators have been proposed for the spanning tree problems in the literature. The edge-set encoding has been shown very effective for such problems as it shows high locality and high heritability. However, it requires effective genetic operators to exploit favorable characteristics of an encoding to guide the search and obtain high quality results. In this work, we consider bounded-diameter minimum spanning tree (BDMST) problem and improve upon the crossover operator for edge-set encoding. The empirical results show the effectiveness of our approach. Finally, based on the simulation results, we highlight interesting properties of the new recombination operator which helps it find better trees compared to the previous one.

Keywords: Bounded-diameter spanning tree problem; spanning tree problem; evolutionary algorithms; edge-set encoding; crossover operators.

1 Introduction

A spanning tree T (V, E) is a connected graph with n = |V| vertices and |E| = n-1 edges such that T contains no cycles. The bounded-diameter minimum spanning tree (BDMST) problem seeks to find spanning tree of an undirected, connected, weighted graph, where the maximum distance between any two vertices does not exceed a given bound *D*. It is a well-studied combinatorial optimization problem, which finds its application in many real-world problems, e.g., design of wire-based communication networks [1], distributed mutual exclusion [2], and bit compression for information retrieval [3] (refer, Abdalla [4] for a detailed view of applications).

The BDMST problem is formally defined as follows. Given $G = (V, E)$ an undirected connected graph with positive edge weights $w(e)$ associated with each edge and a diameter bound D, it seeks to obtain the least cost tree where no path between any two vertices has more than D edges. Thus, the problem is formulated as:

Find a spanning tree T = (V, E) of G that minimizes $W(T) = \sum w(e)$ subject to $diam(T) \leq D$, where $e \in T$.

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 169–179, 2011.

The problem can also be formulated in terms of eccentricity and center of the tree. Let *T* be a spanning tree, the *eccentricity* of vertex v is the maximum number of edges on the path from v to any other vertex in *T* and the *diameter* of *T* is the maximum eccentricity of its vertices. Thus the largest number of edges on any path in *T* is known as its diameter *diam(T)*. The *center* of T is one vertex (if the diameter is *even*) or two adjacent (connected) vertices (if the diameter is *odd*) of minimum eccentricity. Suppose the diameter of the tree is defined by the path $v_1, v_2, \ldots, v_{\lfloor n/2 \rfloor}, v_{\lfloor n/2 \rfloor+1}, \ldots, v_n$. If n is even then $v_{[n/2]}$ is center of the tree and if n is odd then $v_{\lfloor n/2 \rfloor}$ and $v_{\lfloor n/2 \rfloor+1}$ are centers of the tree. In latter case, the edge ($v_{\lfloor n/2 \rfloor}$, $v_{\lfloor n/2 \rfloor+1}$) is called the center edge.

Polynomial time algorithms are known for the special cases having $D = 2$, $D = 3$, $D = n-1$, and when all edge weights are equal. For all other cases, i.e., $4 \leq D < n-1$, the problem has been proved to be NP-Hard [5] and cannot be approximated unless $P = NP$ [6]. Thus no polynomial time approximation algorithm can be guaranteed to find a solution whose cost is within *log(n)* of the optimum.

In this paper, we study the recombination operator for edge-set encoding suggested by Raidl and Julstrom [7], referred hereinafter as RJ-ESRO, and propose three incremental greedy improvements, referred as RJ-ESRO1, RJ-ESRO12 and RJ-ESRO123, to obtain more effective results. However, instead of being hysterical to show better results in comparison to some of the previous results, we aim to analyze the behavior of RJ-ESRO and our recombination operator(s) in extent of improvements they provide. Further, we use another genetic algorithm, referred as JR-PEA, suggested by Julstrom and Raidl [8], which works with permutation encoding to further highlight some interesting properties of our recombination operators.

Rest of the paper is organized as follows. In the next section, we provide a brief overview of the previous work to solve BDMST problem. A review of RJ-ESEA is summarized in Section 3. The improvements proposed in RJ-ESEA, i.e., RJ-ESRO1, RJ-ESRO12, and RJ-ESRO123 are explained in Section 4. Section 5 presents the results of simulations and comparison performed over two different datasets – Euclidean graphs and Random graphs. The paper ends with concluding remarks in Section 6.

2 Previous Work

Many researchers have proposed algorithms to solve BDMST problem which may be classified into two broad categories – exact methods and inexact (heuristic) methods including meta-heuristics. Achuthan et al. [9] proposed a branch-and-bound exact algorithm for BDMST problem. Similarly, Kortsarz and Peleg [6] presented an algorithm for the BDMST that combines greedy heuristic and exhaustive search. Recently, Gruber and Raidl [10] suggested a branch and cut algorithm based on compact 0-1 integer linear programming. However, such exact methods, being deterministic and exhaustive in nature, may be used to solve only small instances of the problem. They will eventually break down on large problem instance owing to the huge search space of the problem.

Deo and Abdalla [11] presented heuristics namely One Time Tree Construction (OTTC) and Iterative Refinement (IR) to solve BDMST problem. OTTC is a modification to the Prim's algorithm [12] to obtain a MST for a pre-specified diameter. It begins with a randomly selected vertex and repeatedly extends the growing tree with the least cost edge between a (partial) tree vertex and an unconnected vertex whose inclusion does not violate the diameter constraint. The IR begins with the unconstraint MST and then, iteratively refines it by replacing one tree edge, starting with the edges that are closer to the center of the tree, with a non-tree edge until it meets diameter constraint or fails to obtain the required BDMST. However, it is computationally expensive and may not produce BDMST for very small values of *D*. Raidl and Julstrom [7] proposed an algorithm Random Greedy Heuristic (RGH) which avoids complete greedy approach of OTTC. It is a center based algorithm which begins constructing the tree with a randomly selected center and grows it iteratively. Rather than always extending the tree with the least cost edge, in each iteration, it chooses an unconnected vertex at random and connects it to the (partial) tree vertex with least cost edge that maintains the diameter constraint. The authors also presented an edge-set-coded steady-state evolutionary algorithm (RJ-ESEA) for BDMST problem. Apart from this, Julstrom and Raidl [8] also proposed a permutation-coded steady-state evolutionary algorithm (JR-PEA), where a permutation of vertices encoded a spanning tree. A decoder, which was based on RGH, was use to decode permutation to make a BDMST. By analyzing RJ-ESEA against JR-PEA on similar problem instances and EA parameters, it was shown that JR-PEA obtained better trees than RJ-ESEA but suffers with a poor worst time complexity of $O(n^2)$ because of the decoder used in its recombination operator. Further, Raidl [13] reported that an EA based on an indirect encoding - random-keys - was almost comparable to JR-PEA. The inherent indirection in random-keys did not affect its performance adversely to obtain as good solutions as JR-PEA on a range of Euclidean instances. Latter, Julstrom [14] proposed an improvement of RGH called CBTC (Center Based Tree Construction). By comparing OTTC, RGH and CBTC it was concluded that RGH performed better than OTTC and CBTC on Euclidean instances, however on non-Euclidean instances the situation gets reversed.

Gruber and Raidl [15] proposed four different types of neighborhoods as local improvement strategies within a Variable Neighborhood Search (VNS) approach and showed that their approach outperformed JR-PEA, RJ-ESEA and an EA based on Random-keys [13]. Latter, Gruber et al. [16] embedded these neighborhoods as local search strategies in an EA and an Ant Colony Optimization (ACO). Considering Euclidean instances, they showed that the EA and the ACO outperformed VNS whereas ACO was better in long-term runs and EA was better in time restricted computation. Recently, Singh and Gupta [17] proposed RGH-I and CBTC-I, which are improved versions of two well-known heuristics RGH and CBTC respectively. It was shown that these improved heuristics produce better results with reduced computational efforts; comparatively, results obtained by RGH-I were the best. They further proposed PEA-I, an improved version of JR-PEA, having a decoder based on RGH-I and other changes like crossover operator. The results obtained by PEA-I were better than RGH-I and CBTC-I. More recently, Binh et al. [18] proposed a new heuristic Center-Based Recursive Clustering (CBRC) and its improved version CBRC-I, which may be seen as an extension of the RGH in a way that both extend the concept of center of the tree to each level of the partially constructed tree. They further discuss two new GAs – one, Hybrid GA (HGA) which consists of multiple populations with migration and the other, MHGA which additionally employ multi-parent crossover. It is shown that the new heuristics and EAs are comparatively better than the earlier selected and well-known heuristics and EAs, respectively.

3 Summary of JR-PEA and RJ-ESEA

As highlighted in previous section, much work after the introduction of RJ-ESEA [7] has been focused on improving initialization heuristics. One key motivation may be that any improvement in these heuristics will also improve the decoder used in permutation-coded encoding, since working principal for latter can be derived from former. However, we aim to highlight interesting properties of edge-set recombination operator and its improvements vis-à-vis their performance to JR-PEA [8].

3.1 Permutation-Coded Evolutionary Algorithm (JR-PEA)

JR-PEA [8] encodes an individual (chromosome) as a permutation on vertices. Every permutation represents a valid tree, i.e., trees whose diameter does not exceed the bound D, because of the decoder, CBTC [14], used to build the tree from the permutation. In this representation, the first vertex (if D is *even*) or the first two vertices (if D is *odd*) constitute center of the tree; in latter case, center of the tree is an edge. Then CBTC is used to complete the tree by iteratively appending remaining vertices, using the (possible) least cost edges, in the listed order in chromosome. Hence, the permutation of vertices represents the genotype and the decoded tree represents the phenotype. The JR-PEA is a hybrid algorithm that uses a greedy heuristic to decode an individual and an EA to explore the search space.

The better performance of the JR-PEA is attributed to the greediness of the heuristic that decodes it chromosomes. However, it comes at a price as the computational complexity of the decoder is $O(n^2)$.

3.2 Edge-Set-Coded Algorithm (RJ-ESEA)

RJ-ESEA [7] uses edge-set representation, which encodes an individual directly as sets of their edges augmented with its center vertex (if D is *even*) or two vertices (if D is *odd*). The fitness of a chromosome is the total weight of the represented tree. The genetic operators are designed in such a way that they always generate a valid tree.

The recombination operator, RJ-ESRO, is based on RGH and builds one offspring from two parents. It is reproduced in fig. 1. It begins with selecting one or two center vertices from the parents' vertices. It maintains a set U of unconnected vertices and a set C of (partial) tree vertices of depth less than $\lfloor D/2 \rfloor$. Edges are appended to the vertices in C. To maintain strong heritability between parents and offspring, it maintains two sets of edges – A_1 and A_2. A_1 contains edges that are common to both the parents, and A_2 contains uncommon edges of the parents. While extending a partial tree, it chooses a random edge from A_1 or a random edge from A_2 if A_1 is

```
$F_1$ ← edges appearing in both the parents;
$F_2$ ← edges appearing in only one parent;
T ← Φ;

// determine the center:
if D is even then
    $v_0$ ← the first parent's $v_0$;
    U ← V - {$v_0$};
    C ← {$v_0$};
    depth[$v_0$] ← 0;
else (D is odd)
    ($v_0$, $v_1$) ← two random vertices from the union of the
    parents' centers;
    T ← {($v_0$, $v_1$)}
    U ← V - {$v_0$, $v_1$};
    C ← {$v_0$, $v_1$};
    depth[$v_0$] ← 0;
    depth[$v_1$] ← 0;
$A_1$ ← all edges from $F_1$ incident to the center;
$A_2$ ← all edges from $F_2$ incident to the center;

// add other nodes iteratively:
while U ≠ Φ do
    if $A_1$ ≠ Φ then
        pick an edge (u, v) ∈ $A_1$ at random;  ◄──── P2
        $A_1$ ← $A_1$ - {(u, v)};
    else if $A_2$ ≠ Φ then
        pick an edge (u, v) ∈ $A_2$ at random;  ◄──── P3
        $A_2$ ← $A_2$ - {(u, v)};
    else
        pick u ∈ C at random;
        pick v ∈ U at random;                   ◄──── P1
    if v ∈ U then
        T ← T ∪ {(u, v)};
        U ← U - {v};
        depth[v] ← depth[u] + 1;
        if depth[v] < $\lfloor D/2 \rfloor$ then
            $A_1$ ← $A_1$ ∪ all edges from $F_1$ incident to v;
            $A_2$ ← $A_2$ ∪ all edges from $F_2$ incident to v;
            C ← C ∪ {v};
Return T;
```

Fig. 1. Recombination operator, RJ-ESRO, in RJ-ESEA

empty. If $\boldsymbol{A_2}$ is also empty, it makes an edge by joining a random vertex from C to a random vertex in U. The computational complexity of the operator is $\boldsymbol{O(n)}$.

4 Improvement in Recombination Operator RJ-ESRO

We run RJ-ESEA [7] and JR-PEA [8] as proposed and with same (algorithm specific and problem specific) input parameters as suggested by the respective authors but without mutation operators. The population size is 400, probability of crossover (P_c) is 0.6, and selection operator is tournament with size 3. Individuals in the initial population in RJ-ESEA are generated using RGH whereas for the JR-PEA they are generated as suggested in [8]. We consider OR-library [20] datasets for Euclidean graphs and generate our own dataset in the range of [0.01, 0.99] for random graphs. Though the initial population is very similar in both the EAs as RGH is a generalization of decoder used to decode an individual in JR-PEA, RJ-ESEA starts with comparatively very high average fitness value since first generation. This trend is highlighted in Fig. 2, which presents average fitness values of the populations for first 100 generations of RJ-ESEA and JR-PEA for problem instances of 250 vertices on OR-library dataset. For further investigation, we ran RJ-ESEA and JR-PEA with same initial population; it validated this trend as we obtained similar results once again.

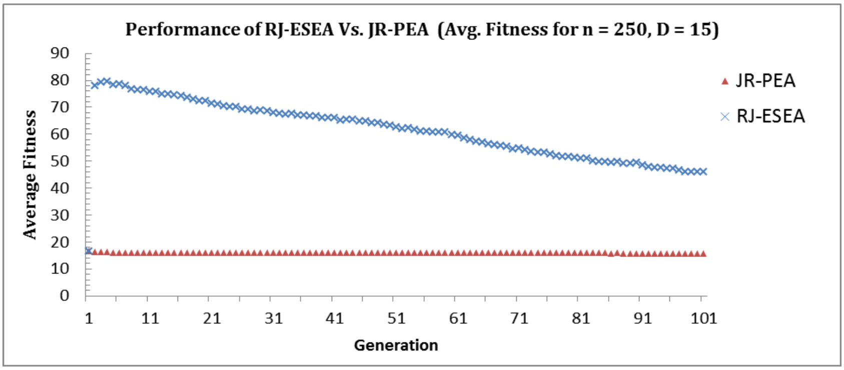


Fig. 2. The average fitness plot of RJ-ESEA and JR-PEA for 250 vertices Euclidean graph with D = 15

This behavior of RJ-ESEA is attributed to its recombination operator as it does not contain any greedy step. Probably, the authors chose so for low computational complexity. However, due to its non-greedy nature, it is not able to handle the initial greediness imparted on the solutions by RGH. The randomness at all the stages of an edge selection nullifies the low average fitness obtained by the RGH. In fact, it will exhibit this behavior with any greedy initialization heuristic. Nevertheless, the strong heritability characteristic of the operator still leads it to maintain and converge to good solutions by standard GA mechanism. The phenomenon of fitness reallocation will require RJ-ESEA to consume more iterations than JR-PEA, but probably an overall less time due to its less complexity compared to JR-PEA.

In order to address the problem, we propose three greedy improvements into the recombination operator and analyze the effects by comparing them with original recombination operator, RJ-ESRO, of the RJ-ESEA. For comparison and to measure the effectiveness of induced greediness we include the results of JR-PEA also. We identify three possible places for greedy improvements; refer three marked places, P1, P2, and P3 in Fig. 1. We refer the original recombination operator (Fig. 1) defined in RJ-ESEA as RJ-ESRO and analogously the modified operator(s), the one incorporating greediness at P1, the other incorporating greediness at P1 and P2 both, and the third incorporating greediness at all the places P1, P2 and P3 are referred as RJ-ESRO1, RJ-ESRO12 and RJ-ESRO123 respectively. The greedy improvements are as follows:

- P1- instead of taking a vertex $v \in U$ randomly, consider a vertex v that connects u with lowest edge.
- P2- instead of taking an edge $(u, v) \in A_1$ randomly, consider the smallest edge in A_1.
- P3 – instead of taking an edge $(u, v) \in A_2$ randomly, consider the smallest edge in $A_{2.}$

We realize that the three greedy improvements will increase the running time of our recombination operator in comparison to RJ-ESRO, but with better data structures, like heaps and priority queues the greedy improvements can be achieved in $O(nlog(n))$ time as well. Moreover, we expect that our changes will not possibly degrade the performance of new recombination operator much as is the experience of other researchers, e.g., Singh and Gupta [17], who observed that their permutation coded EA, PEA-I, always takes less time to reach to the best solution in comparison to edge-set coded EA, whereas worst-case time complexity to decode an individual in permutation coded EA and edge-set coded EA is $O(n^2)$ and $O(n)$ respectively.

5 Simulation and Results

We run RJ-ESEA [7] and JR-PEA [8] with standard C++ in windows platform on a Pentium 4 based machine with input parameters as suggested in [7] and [8] and described in Section 4. As discussed, we consider OR-library [20] datasets for Euclidean graph instances and self-generated dataset where weights are in the range of [0.01, 0.99] as Random graph instances having 50, 100, 250, and 500 vertices. The diameter bound is also chosen in tune with [7] and [8]; when the number of vertices n = 50, the diameter bound is D = 5; when n = 100, D = 10; when n = 250, D = 15; and when n = 500, D = 20. In all, we experiment with four different edge-set based EAs having different recombination operators: (i) RJ-ESRO, (ii) RJ-ESRO1, (iii) RJ-ESRO12, and (iv) RJ-ESRO123 and one permutation based EA, JR-PEA, with an aim to observe the level of improvement achieved, if any, by new greedy operators over the original RJ-ESRO and JR-PEA.

The results obtained after 100 generations for Euclidean graph instances and random graph instances are presented in Table 1 and Table 2 respectively. It is clear that there is a remarkable improvement in the performance of RJ-ESRO123 over the RJ-ESRO for the both the instances and there is a gradual improvement in obtained results with each greedy step induced in the RJ-ESRO, i.e., RJ-ESRO123 < RJ-ESRO12 < RJ-ESRO1 < RJ-ESRO (kindly remember, it is a minimization problem).

Table 1. Results obtained after 100 generations for Euclidean graph instances

Parameters		JR-PEA		RJ-ESRO		RJ-ESRO1		RJ-ESRO12		RJ-ESRO123	
Size	D	Best	Avg.	Best	Avg.	Best	Avg.	Best	Avg.	Best	Avg.
50	5	7.845	8.524	11.339	12.061	9.426	10.162	8.802	9.468	10.364	10.890
100	10	8.856	9.682	14.991	15.249	13.028	14.727	13.302	14.815	10.949	12.022
250	15	14.731	15.466	43.280	46.080	36.371	40.677	36.416	40.890	24.269	25.787
500	20	24.247	21.120	96.908	104.140	56.718	61.598	57.785	62.066	32.308	36.907

Table 2. Results obtained after 100 generations for random graph instances

Parameters		JR-PEA		RJ-ESRO		RJ-ESRO1		RJ-ESRO12		RJ-ESRO123	
Size	D	Best	Avg.	Best	Avg.	Best	Avg.	Best	Avg.	Best	Avg.
50	5	2.544	3.099	5.361	5.602	4.199	5.127	5.161	5.559	3.176	3.3551
100	10	3.834	4.250	7.380	7.879	4.412	4.718	4.188	4.425	3.957	4.176
250	15	7.743	8.186	25328	29.396	10.547	13.257	9.928	13.257	9.215	9.708
500	20	12.430	13.184	60.100	63.221	16.460	17.554	14.446	15.469	13.220	14.091

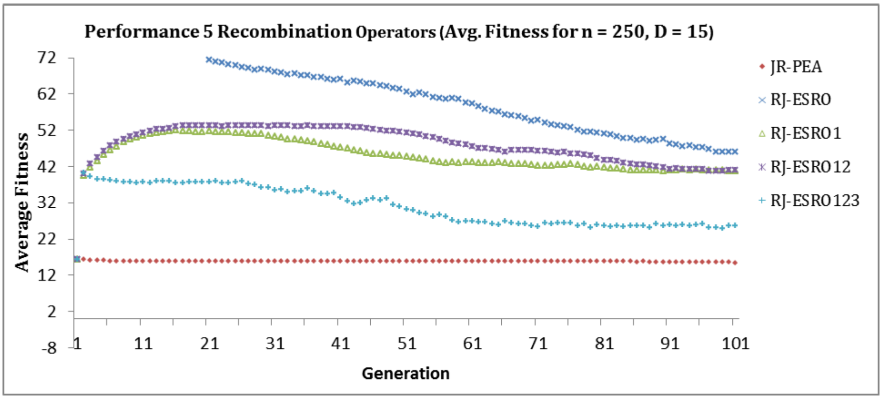


Fig. 3. The average fitness plot of RJ-ESEA with RJ-ESRO, RJ-ESRO1, RJ-ESRO12, RJ-ESRO123 and JR-PEA for 250 vertices Euclidean graph with D = 15.

A plot of average fitness values for the five EAs on Euclidean graph instance of 250 vertices for first 100 generations is shown in Fig. 3. Though the performance of RJ-ESRO123 is not a match for JR-PEA (it is evident from Table 1), there is a noticeable performance improvement in RJ-ESRO123 in comparison to RJ-ESRO. As expected, because of the induced greediness, RJ-ESRO123 also brings down the nullification effect of the recombination operator on initial low fitness values obtained by the greedy initialization heuristic to a minimum level (refer, plots of RJ-ESRO and RJ-ESRO123 in Fig. 3). Accordingly, properties of each of the recombination operator are discussed below.

5.1 PMX in JR-PEA

JR-PEA uses partially matched crossover (PMX) [19] as recombination operator and is able to preserve the initial low average fitness values because of its decoder - to represent an individual - which is based on RGH. The heritability of this operator is also strong but convergence is slow (refer Fig. 3, convergence is not very steep). It is probably because of the good quality solutions in initial populations themselves due to RGH based decoder. Although the theoretical worst case bound of JR-PEA is $O(n^2)$ due to its decoder, we observed that the execution time of JR-PEA was consistently less than RJ-ESEA based EAs. This is in support of what was previously reported in [17].

5.2 RJ-ESRO in RJ-ESEA

As mentioned in Section 3, RJ-ESRO is not able to handle the initial greediness imparted on the solutions by RGH because of its non-greedy nature. The randomness at all the stages of an edge selection nullifies the low average fitness obtained by the RGH on initial population. However, it maintains a strong heritability behavior, which leads it to have a good convergence rate (refer Fig. 3; it has a strong convergence descent pattern). Nevertheless, the phenomenon of fitness reallocation will require RJ-ESEA to consume more iterations than JR-PEA.

5.3 RJ-ESRO1 in RJ-ESEA

This operator adds a greedy step to RJ-ESRO; after picking a random vertex $\mathrm{u} \in \mathrm{C}$, it chooses a vertex $\mathrm{v} \in \mathrm{U}$ which connects it to u with the least cost edge available. It shows good improvement over RJ-ESRO but this greedy step is not executed frequently in the initial generations. This is primarily because in initial generations the population is fairly random and as a consequence the number of uncommon edges between two parents is very large, due to which most edges taken in the offspring are chosen from set A_2. But still, in all those cases, where set A_1 and A_2 are exhausted, this operator optimizes the edge creation in comparison to original operator by inducing a greedy strategy. In effect, on contrary to RJ-ESRO, it maintains good edges in an individual because of induced greediness. However, it exhibits a strange behavior of showing a bump in its convergence pattern in the initial generations. It can be attributed to the fact that since most edges in an offspring (in initial generations) are from A_1 and A_2 and are still randomly chosen, the problem of fitness reallocation inherent in original RJ-ESRO is not entirely mitigated by this operator. Nonetheless, it clearly marks an improvement over the RJ-ESRO.

5.4 RJ-ESRO12 in RJ-ESEA

This operator adds another greedy step to RJ-ESRO1; here, it chooses the least cost edge $(\mathrm{u}, \mathrm{v}) \in \mathrm{A}_1$ instead of choosing a random edge. Set A_1 maintains the permissible edges which are common to both the parents. This operator does not show much improvement over RJ-ESRO1 and its convergence pattern is almost coinciding with it. This behavior can be explained by observing that in early generations individuals in population have very few common edges and as a consequence set A_1 is of relatively small cardinality, Hence greediness induced at this stage proves to be less effective than expected.

5.5 RJ-ESRO123 in RJ-ESEA

This operator adds another greedy step to RJ-ESRO12; here, it chooses the least cost edge $(u, v) \in A_2$ instead of choosing a random edge. Addition of this greedy step makes it completely greedy for selection of an edge. It shows maximum improvement, as it introduces greediness in selection criterion for set A_2, which contains uncommon edges. This set is usually large among two parents chosen for crossover in initial generations and as such greedy step is executed frequently in initial generations. Also, as the EA progresses the individuals in the population starts sharing many edges. Hence, in later generations the greediness retained by this operator in set A_1, starts proving effective. The net effect is that, for all possible cases RJ-ESRO123 maintains good greediness and strives to improvise at all times compared to RJ-ESRO. A further consequence of RJ-ESRO123 is that, unlike other operators, it does not suffer with fitness reallocation, and converges smoothly and consistently through subsequent generations. Moreover, none of the induced greedy step disturbs the heritability of the operator. Hence, RJ-ERO123 shows a good heritability and performs much better due to greediness in comparison to RJ-ESRO.

6 Conclusion and Future Work

In this paper, we proposed a new recombination operator (RJ-ESRO123) which is an improvement over recombination operator (RJ-ESRO) defined in RJ-ESEA framework. The RJ-ESRO123 has essentially three greedy improvements over RJ-ESRO and is shown to be comparatively more effective. We also show some of the interesting properties of new recombination operator. Moreover, this paper sets the general tone regarding the scope of quality improvement which can be achieved by improving recombination operator.

This paper projects scope of a more comprehensive analysis of various strategies – encodings, genetic operators, and EAs - proposed in the BDMST optimization domain. As mentioned in Section 2, several improvements have been suggested in initialization heuristics and some researchers have also tried new EA frameworks to solve BDMST problem, a study taking in consideration all of these improvements and comparing them on standard benchmark datasets will be next line of work to find the best combination for solving BDMST problem. Moreover, such an effort will also help to investigate some of the abstruse behavior shown by various operators among several EA frameworks. The authors of the paper are currently investigating the likelihood of making such a research design to put forward more thorough analysis of various encodings and heuristics for BDMST problem.

References

1. Bala, K., Petropoulos, K., Stern, T.E.: Multicasting in a Linear Lightwave Network. In: IEEE INFOCOM 1993, pp. 1350–1358. IEEE Computer Society Press, Los Alamitos (1993)
2. Raymond, K.: A Tree-based Algorithm for Distributed Mutual Exclusion. ACM Transactions on Computer Systems 7(1), 61–67 (1989)

3. Bookstein, A., Klein, S.T.: Compression of Correlated Bit-Vectors. Information Systems 16(4), 387–400 (1996)
4. Abdalla, A.: Computing a Diameter-constrained Minimum Spanning Tree. PhD Dissertation, The School of Electrical Engineering and Computer Science, University of Central Florida (2001)
5. Garey, M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-Completeness. W. H. Freeman, San Francisco (1979)
6. Kortsarz, G., Peleg, D.: Approximating Shallow-light Trees. In: Symposium on Discrete Algorithms, pp. 103–110 (1997)
7. Raidl, G.R., Julstrom, B.A.: Greedy Heuristics and an Evolutionary Algorithm for the Bounded-Diameter Minimum Spanning Tree Problem. In: ACM Symposium on Applied Computing, pp. 747–752 (2003)
8. Raidl, G.R., Julstrom, B.A.: A Permutation Coded Evolutionary for the Bounded Diameter Minimum Spanning Tree Problem. In: Genetic and Evolutionary Computation Conference, pp. 2–7 (2003)
9. Achuthan, N.R., Caccetta, L., Cacetta, P., Geelen, J.F.: Algorithms for the Minimum Weight Spanning Tree with Bounded Diameter Problem Optimization. Australian Journal of Combinatorics 10, 51–57 (1994)
10. Gruber. M., Raidl, G.R.: A New 0-1 ILP Approach for the Bounded Diameter Minimum Spanning Tree Problem. In: International Network Optimization Conference (2005)
11. Deo, N., Abdalla, A.: Computing a diameter-constrained minimum spanning tree in parallel. In: Bongiovanni, G., Petreschi, R., Gambosi, G. (eds.) CIAC 2000. LNCS, vol. 1767, pp. 17–31. Springer, Heidelberg (2000)
12. Prim, R.: Shortest Connection Networks and Some Generalization. Bell System Technical Journal 36, 1389–1401 (1957)
13. Julstrom, B.A.: Encoding bounded-diameter spanning trees with permutations and with random keys. In: Deb, K., et al. (eds.) GECCO 2004. LNCS, vol. 3102, pp. 1272–1281. Springer, Heidelberg (2004)
14. Julstrom, B.A.: Greedy Heuristics for the Bounded-Diameter Minimum Spanning Tree Problem. ACM J. Exp. Algorithmics (2004)
15. Gruber, M., Raidl, G.R.: Variable Neighbourhood Search for the Bounded Diameter Minimum Spanning Tree Problem. In: 18th Mini Euro Conference on Variable Neighborhood Search, Spain (2006)
16. Gruber, M., Hemert, J., Raidl, G.R.: Neighborhood Searches for the Bounded Diameter Minimum Spanning Tree Problem Embedded in a VNS, EA and ACO. In: Genetic and Evolutionary Computational Conference, pp. 1187–1194 (2006)
17. Singh, A., Gupta, A.K.: An Improved Heuristic for the Bounded Diameter Minimum Spanning Tree Problem. Journal of Soft Computing 11, 911–921 (2007)
18. Binh, H.T.T., Hoai, N.X., McKay, R.I., Nghia, N.D.: New Heuristic and Hybrid Genetic Algorithm for Solving the Bounded Diameter Minimum Spanning Tree Problem. In: Genetic and Evolutionary Computational Conference. pp. 373–380 (2009)
19. Goldberg, D.E., Lingle, J.R.: Alleles, Loci, and the Travelling Salesman Problem. In: International Conference on Genetic Algorithms, pp. 154–159 (1985)
20. Beasley, J.E.: OR-library: Distributing Test Problems by Electronic Mail. Journal of the Operational Research Society 41, 1069–1072 (1990)

An Analysis of Security Related Issues in Cloud Computing

L.D. Dhinesh Babu[1], P. Venkata Krishna[2], A. Mohammed Zayan[1],
and Vijayant Panda[1]

[1] School of Information Technology and Engineering, VIT University, Vellore,
632 014, Tamil Nadu, India
[2] School of Computing Science Engineering, VIT University, Vellore,
632 014, Tamil Nadu, India
{lddhineshbabu,pvenkatakrishna}@vit.ac.in,
{zayan.vit,pandavijaya89}@gmail.com

Abstract. Over the past two decades, the scenario in the computing world has evolved from client-server to distributed systems and then to central virtualization called as cloud computing. Computing world is moving towards Cloud Computing and it remains as buzzword of the current era. Earlier, users had complete control over their processes and data stored in personal computer where as in cloud, cloud vendor provides services and data storage in remote location over which the client has no control or information. As application and data processing takes place in public domain outside the designated firewall, several security concerns and issues arise. The main objective of the paper is to provide an overall security perspective in cloud Computing and highlight the security concerns and other issues. The paper also highlights few technical security issues in cloud computing.

Keywords: Cloud Computing, Cloud Security, Software as a Service, Platform as a Service, Infrastructure as a Service, Public Private and Hybrid Clouds.

1 Introduction

The history of cloud goes way back to 1960.In 1960, John McCarthy proposed a computation model as a kind of public utility [1]. Cloud computing is one of the biggest evolving technologies. It basically originates from the idea that client's side work, data; applications can be moved to some remote unseen cluster of resources on the internet. All the required resources would be stored in one remote location and user's can connect and use the resources whenever required. The first public usage of cloud appeared in MIT published paper in 1996 [2]. Amazon introduced cloud based services consisting of data storage and information processing via Amazon Mechanical Trunk in 2002 [3].

IBM adopted the cloud computing methods and introduced the automated, pervasive, Grid computing and utility computing methods [4]. Amazon started a web application named as Elastic Compute Cloud (EC2) web service that allows users to hire computer systems so that they can use their own computer application. The

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 180–190, 2011.

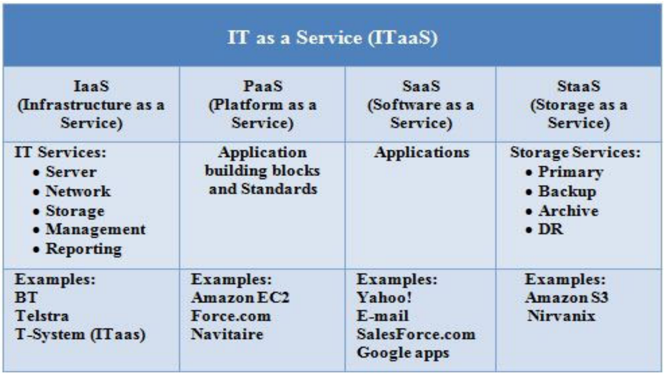
pay-per-use service feature and subscription based usage extends Information technology's current capabilities [5]. Cloud computing involves utilizing the virtualized resources over the internet [6].

2 Types of Clouds

Public Cloud. In public cloud, user accesses the cloud services with the help of interfaces using conventional browsers. Public cloud is mainly a pay and use web service. With the help of this, the clients can match the entire operational level information technology expenditure by reducing infrastructure level capital IT expenditure [7]. In view of the security aspect public clouds are less secure compared to other clouds. This is because they are susceptible to malicious attacks. A solution is that both the provider and client can mutually agree to set up a joint responsibility to check cloud security and validate the cloud processing actions. The trust and privacy issues are very important in public clouds.

Private Cloud. Private cloud is used within an organization's server and data centre. Therefore they are more secure and easier to provide security checks. All the security compliances, regulatory requirements fit properly. Since it is located within the enterprise deployment and usage becomes easier. All the virtual applications are available in the private cloud for user to share and use. The difference between private and public clouds is that all the virtual applications and cloud resources are managed by the enterprise and not by the cloud provider.

Hybrid Cloud. Hybrid cloud is more or less similar to a private model cloud but it is attached to an external cloud. It is managed from a central unit and identified as whole single unit confined by an impregnable network [8]. Hybrid clouds are generally used for back up purpose. The local data can be duplicated on public clouds. The open architecture feature in hybrid clouds allows access and interacts with other systems in a secure way.



IT as a Service (ITaaS)			
IaaS (Infrastructure as a Service)	PaaS (Platform as a Service)	SaaS (Software as a Service)	StaaS (Storage as a Service)
IT Services: • Server • Network • Storage • Management • Reporting	Application building blocks and Standards	Applications	Storage Services: • Primary • Backup • Archive • DR
Examples: BT Telstra T-System (ITaas)	Examples: Amazon EC2 Force.com Navitaire	Examples: Yahoo! E-mail SalesForce.com Google apps	Examples: Amazon S3 Nirvanix

Fig. 1. Types of cloud computing services examples

3 Cloud Security

According to Gartner, cloud computing is filled with security risks. Customers will definitely ask for security assurance from third party vendors before entering into the cloud applications. Gartner states that cloud computing security involves various attributes such as data privacy, data integrity, recovery, data ownership and other legal issues such as regulatory compliances and auditing.

The security issues in cloud computing basically fall into two different categories. One is security issues faced by clients using their services and another one faced by cloud vendors including Software, Platform, or Infrastructure-as-a-Service and. Before engaging in commercial service usage, the client should ensure that vendor has provided them all the security measures to protect their data [9].

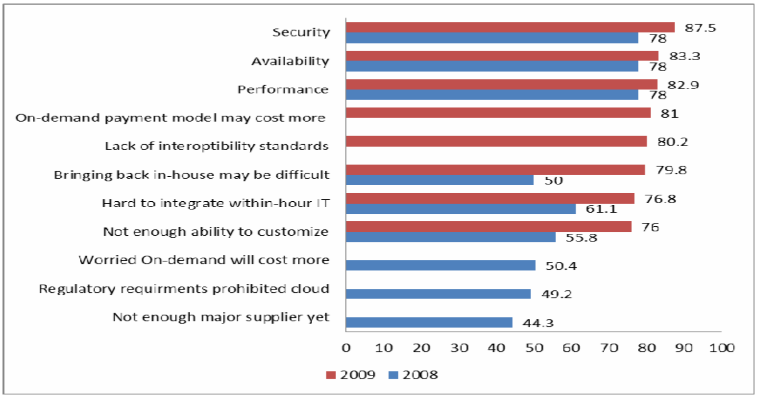


Fig. 2. IDC 08:09 Cloud Challenges Survey

3.1 Dimensions of Cloud Security

Cloud Security Alliance identifies fifteen areas of concern. These dimensions have been grouped into three general areas: Compliance, Legal or Contractual Issues and Security and Privacy [10]. The major security areas in cloud computing are

Data protection. For the client's data to be secure proper data segregation of data should be done. During data access, the data should move securely from the remote location to the client's desk. Few secure systems are provided by the vendors to prevent data exposure or misuse by third party vendors.

Identity management. Identity management system is used in enterprises to control user access to secure data and other useful virtualized resources. Cloud vendors use SSO and federation technology to integrate the client's identity into their own infrastructure. Using this, the vendors provide their own identity management.

Physical and personnel security. Cloud vendors must have secure physical machines and must ensure that access to these machines is restricted. Also each client's personal security must be maintained. The access to the data must be recorded for future reference.

Availability. Cloud providers must assure their customers about non interrupted data access. The data access must regular and predictable.

Application security. The applications provided as cloud services must be secured by implementing important testing cases and other regulatory acceptance measures for outside application package codes. All the application level security measures and application firewalls must be ensured by the vendor.

Privacy. All the critical data such as pin numbers must be masked and authentication measures must be provided so that only authorized users can access the data. The customers' digital signatures and credentials must be secured. The data collected and produced by the cloud vendors must be privately protected.

Business continuity and data recovery. In case of service failure cloud providers have data recovery and other continuity methods and plan so that they can continue their business. These plans should be reviewed by their clients.

Logs and audit trails. The cloud vendors ensure that logs and audit trails are produced securely and maintained. These trails would be used for purpose of investigation in case of emergency.

3.2 Security Concerns

SaaS concerns. In SaaS, users rely mainly on cloud provider's security. It is the responsibility of the provider to enforce privacy requirements. The provider must provide protection for underlying infrastructure from intruders and has the most important responsibility of authentication details. There is no proper assurance that a required application would be readily available when needed. The customer faces a problem to get the details that ensure that right security measurements are being done.

PaaS concerns. In PaaS, the provider gives some level of control to the developers like the developers can develop their own authentication systems and data encryption methods. But the host or network intrusion prevention security measures are with the cloud provider. Platform provider should provide assurance that the client's data is inaccessible between different applications.

IaaS concerns. In IaaS, the developer has better control over the cloud security. This is mainly because applications run on separate virtual machines on the same physical machine. This control helps developers to address security concerns with a disadvantage that building the application is more time consuming process and needs huge investment. Backing up of the customers data is another critical issue. Providers can increase the cost and make it difficult to get the clients' sensitive data and other resources off their network.

3.3 Security Issues

Some of the security issues are as follows

1. The physical security is lost in cloud model because all other resources are shared with other organizations. There is no control or identification where the resources run in remote location.

2. Some of the cloud provider's services may differ from other provider's services because of incompatibility issues. Therefore during transfer from one service to another provider service incompatibilities and security issues might come up [11].
3. There is certain amount of risk involved in law violation by he provider.
4. Basically, the encryption and decryption keys should be operated by the customers so as to prevent misuse.
5. Common data integrity standard does not cover all the security aspects. The transfer, storage and retrieval of data mean maintaining data integrity with respect to data transactions only.
6. PCI DSS data logs should be provided to the security managers for future reference [12] [13] [14].
7. Clients must be up to date with the security measures involved in cloud applications.
8. Government have rules that citizen's private data should not be used and stored for a long time, and some banking regulations state that client's financial and business numbers data should stay in their respective country.
9. The provider may be sued by the customer if the provider knowingly violates their privacy rights. Security concerns arise when providers request personal information from the customers because they are not sure how the information would be used or passed.
10. **Storage.** Another issue involves about the location of data storage in cloud. The issue revolves around whether any personal data was moved to another data centre in another country. Few security laws in countries restrict on the transfer of personal information to other countries.
11. **Data destruction.** The providers should not make additional copies of personal data and retain them. Instead the providers should destruct it after verification. The clients are unsure whether their personal information is secure and why their information is kept for so long with vendor.
12. **Security breaches.** The client should ensure that cloud provider should notify them whenever there is a security or privacy breach. The provider should notify the client who is responsible for handling the breach and how. There should be some contract which includes liability dues. Provider should ensure how the contract would be enforced and should determine the cost charges associated with the entire process.
13. **Audit and monitoring.** Clients should monitor the providers and provide necessary assurance to their customers that all privacy and security requirements are met with the personal identifiable information in cloud.
14. **Data access.** The clients should know what personal information of theirs is stored with the provider. They must have a right to stop the flow of data from their servers to cloud servers. When the data subjects ask for deletion of their personal information, the providers should ensure that all of their information is deleted from the cloud and no additional copied have been retained.

Few security issues stated by Gartner that clients and other decision makers, should discuss with Cloud computing [15] vendors before utilizing the vendor's services are:

Privileged access. Which users utilizing services have been granted privileged access? Who and from where the security administrators would be hired?

Regulatory compliance. The cloud provider should undergo external security checks and regular audits.

Data location. In which remote location the data is stored and who has the control over the client's data?

Data segregation. Whether data encryption and decryption is available for all applications and who designed the encryption schemes?

Recovery. In case of a system crash or failure, what happens to the user's data? And is there any complete back up option and security of critical data. In case of a failure how long does the recovery process take?

Investigative support. Does the cloud provider have any kind of right or capability to look into any illegal activity?

Long-term viability. If the provider closes the business, what happens to the client data and in what format the client's data is returned? Whether the provider has retained additional copies of client data?

Data availability. What happens when the user's data is moved into a different platform or environment? Is the data availability hampered in new environment? What happens to the existing environment?

3.4 Security Policy in Cloud Computing Environment

The security policy for solving the security issues should consist of the following points

1. The user should have secure connection with the help of SSL, VPN, PPTP, etc. There should be multiple authorizations among various users and agents so that user accesses the data securely.
2. User's requirements differ from person to person. So, different data protection measures should be enforced for different users.
3. The dynamic user requirements like single sign on authentication, proxy, joint certification and different security domain's certification should be solved and security measures should be provided for the same.
4. The cloud environment should be divided into different domains and each domain should have a security policy. Different security domains must have mutual authentication and each domain security should have a mapping between the global and local domain.
5. The service requests from the users should undergo various safety tests to check whether they contain malicious requests.
6. Third party monitoring mechanisms must be enforced to ensure that the cloud operating environment is safe and secure.

3.5 Information Security Requirements

For cloud computing to be a secure technology, the information security requirements as proposed by ISO (ISO-7498-2) must satisfy the security requirements in all the cloud delivery models. The information security requirements along with the deployment models and cloud delivery models are matched with other to provide an assessment level in context of cloud security as shown in figure 3. Figure illustrates the information security requirements along with delivery models and cloud deployment models [16].

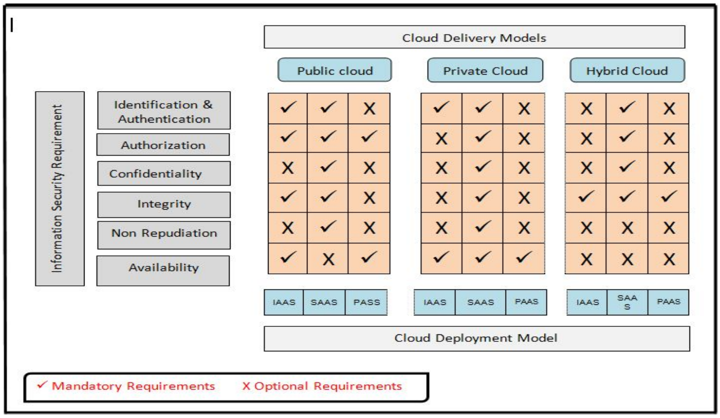
Identification & Authentication. In cloud computing, this process is aimed at providing a valid username and password for cloud user profiles. Only specific predefined identified users must be first created and access rights should be provided to them accordingly.

Authorization. Authorization requirement maintains referential integrity. Client's authorization adds some amount of control and exclusive rights over process flows within a cloud. In a private cloud the authorization is controlled by an administrator.

Confidentiality. Confidentiality maintains the control over organization's data from distributed databases. Confidentiality is must in public clouds because of public accessibility. In order to provide confidentiality of user's data security rules must be provided at various cloud layers.

Integrity. Diligent efforts must be taken by the provider while accessing and retrieving user's data in a public domain. The ACID (atomicity, consistency, isolation and durability) properties should be enforced across all cloud delivery models.

Non-Repudiation. Digital signatures, timestamps, token passing, security certificates, confirmation messages and other e-commerce security protocols can help in achieving the non-repudiation security requirement.



	Public cloud			Private Cloud			Hybrid Cloud		
	IAAS	SAAS	PASS	IAAS	SAAS	PAAS	IAAS	SAA S	PAAS
Identification & Authentication	✓	✓	X	✓	✓	X	X	✓	X
Authorization	✓	✓	✓	X	✓	X	X	✓	X
Confidentiality	X	✓	X	X	✓	X	X	✓	X
Integrity	✓	✓	X	X	✓	X	✓	✓	✓
Non Repudiation	X	✓	X	X	✓	X	X	X	X
Availability	✓	X	✓	✓	✓	✓	X	X	X

✓ Mandatory Requirements X Optional Requirements

Fig. 3. Cloud Computing Security Requirements

Availability. The SLA i.e. the service level agreement document formulated by mutual agreement between client and provider provides the entire details of availability of services and other cloud resources. This requirement plays a prime factor in deciding which public, private cloud providers to use and also in deciding the appropriate delivery models.

4 Few Technical Issues

4.1 Browser Security

One of the important concerns of cloud computing is browser security. In cloud computing, remote servers are the servers in which all the computations and operations are done. The client's systems are used only for authorizing, sending and receiving the command to the respective cloud. A browser should be universal, platform independent and it should comprise of all security and safety issues.

Web Browsers with Asynchronous JavaScript XML (AJAX) is well suited for I/O but as far as security is concerned TLS (Transport Layer Security) to be taken in account which is used for host authentication and data encryption. Web browsers cannot encrypt data using XML encryption or XML signature, so TLS has to be used to encrypt data. The main objective is to serve a secure connection using TLS and at the same time XML based encryption has to be included in the Web browser.

4.2 Attacks on Browser-Based Cloud Authentication

Federated Identity Management (FIM) protocols can be used to explain the realization of security issues in browser-based protocols with clouds. Since the Web browser is not able to authenticate against the cloud, so a third party is introduced. Microsoft's Passport is the prototype for this class of protocols [17] which has been broken off by Slemko [18]. An Http is used to redirect to the passport login server when there is no direct login is possible to the server. Passport Server is used to convert authentication into Kerberos token. This Kerberos tokens are sent to the calling server through another HTTP redirect. As Kerberos tokens are not attached to the Web browser, which becomes a main security problem with Passport and SOP is the only protocol which protects these tokens. All Services of a victim can easily be accessed by an attacker, if he can access this token. These Passports uses a REST type of communication. The successors of Passport MS CardSpace and the SAML family of protocols emphatically belong to the world of Web Services.

However the same problem prevails. GROB [19] analyzed SAML browser profiles, and some of the other authors also demonstrated the attacks on MS CardSpace [20] [21] which can be also applicable to the SAML browser profiles. Current Web browsers authentication protocol is not well secured. The reason behind it is that web browser in not able to issue XML based security by itself. Another main reason is that FIM (Federated Identity Management) systems store the tokens within the web browser. Insecure SOP is the only protocol which is protecting the tokens from threats.

4.3 Secure Browser-Based Authentication

We can secure FIM system by integrating TLP and SOP in a better way. With the help of TLS, there are four methods to protect SAML protocols.

TLS federation. In this method, the tokens are sent within X.509 client certificate. This SAML token is replaced with other identification data with different names. This validity of the certificate and the SAML token are same [22].

SAML 2.0 HOK assertion profile (Holder-of-Key). Transport Layer Security is used with authentication of client's here. But there is no transfer of any authorized information by client's certificate. Instead, the token is bounded to the public key that is contained in this certificate, by adding this type of key in a HOK assertion [23]. More detailed analysis on the requirements and security requirements for this approach is given in [24].

Strong locked same origin policy. The previous approach was purely dependent on the server authentication of the client. Here it is strengthened by making the client to take reliable security decisions. This can be done by using server's public key as a base for taking decisions for the SOP (Same Origin Policy) instead of using unsafe DNS [25].

TLS session binding. The token binding to a specific TLS session ensures that the data sent in response to SAML token is secured by the same Transport Layer Security channel thereby making sure that the data reaches the same unknown user who has sent the SAML token.

5 Future Browser Enhancements

When we are embedding TLS in the Web browser, it still not enough to authenticate the user's data for cloud computing. Many web Services functionalities can be added in the web Browser by adding an appropriate JavaScript library during runtime. This does not apply for XML Encryption and XML signature .This is because the algorithms and cryptographic keys require higher protection. It is appropriate to add two additional functionalities to the Web browser security API.

5.1 XML Encryption

In XML signature standard API can be easily adapted. In this XML Signature approach, XML knowledge is not necessary. Here encryption or decryption of only a byte stream of data has to be done. The Cryptographic keys can be accessed by a naming scheme. An API should grant to access this scheme. Therefore the malicious code can be accessed easily since the decrypted are stored within the web browser.

5.2 XML Signature

XML Signature is non-trivial. It is because this XML signature data element should be verified inside the Application Program Interface. The processing of the structure element takes place inside core browser. The processing includes the transformation on the signed part and the 2 step hash methods. Additionally, steps to negate the XML wrapping attacks must be implemented. And also the API must be Powerful enough to help all Standard key agreement methods that is specified in WS-Security.

All the resulting keys must be stored within the browser. By enhancing the API security (e.g. PKCS#11) this could be done. This is applicable to web services and therefore it will also apply for Cloud Computing.

6 Conclusions

In this paper we have briefed about various cloud security issues and other security concerns. We have also discussed about various security areas in cloud computing and presented brief information about cloud's information security requirements. Additionally few technical securities such as browser security and xml encryption standards have also been introduced. Cloud security is one of the important topics and will certainly be discussed and researched more in future years of cloud computing.

References

1. McCarthy, J.: Recursive functions of symbolic expressions and their computation by machine. Communications of the ACM 3(4), 184–195 (1960)
2. Sharon Eisner Gillett., Mitchell Kapor.,: The Self-governing Internet, Coordination by Design. In: Coordination and Administration of the Internet Workshop at Kennedy School of Government, Harvard University (1996)
3. Amazon mechanical Turk, Artificial Intelligence, https://www.mturk.com/mturk/welcome
4. Khalid, A.: Cloud Computing, Applying Issues in Small. In: International Conference on Signal Acquisition and Processing (2010)
5. Knorr, E., Gruman, G.: What cloud computing really means (2008), http://www.infoworld.com/auhor-bios/galengruman
6. Gartner.: Gartner Say's Cloud Computing Will Be as Influential As E-business. Gartner.com (2010) ; Gruman, G.: What cloud computing really means. InfoWorld (2009) (retrieved)
7. A Platform Computing Whitepaper: Enterprise Cloud Computing-Transforming IT, Platform Computing, p. 6 (2010)
8. Global Netoptex Incorporated: Demystifying the cloud. Important opportunities, crucial choices, 4–14 (2009), http://www.gni.com
9. Swamp Computing (Cloud Computing).: Web Security Journal (2009), http://security.sys-con.com/node/1231725
10. Sampling of issues we are addressing: Cloud Security Alliance, http://www.cloudsecurityalliance.org/issues.html#ediscovery
11. Casassa-Mont, M., Pearson, S., Bramhall, P.: Towards Accountable Management of Identity and Privacy- Sticky Policies and Enforceable Tracing Services. In: Casassa-Mont, M., Pearson, S., Bramhall, P. (eds.) Proc. DEXA, pp. 372–382. IEEE Computer Society, Los Alamitos (2003)
12. PCI Security Standard, https://www.pcisecuritystandards.org/index.shtml
13. Payment Card Industry Security, http://en.wikipedia.org/wiki/Payment_Card_Industry_Data_Security_Standard
14. Brodkin J., Gartner: Seven cloud-computing security risks. Infoworld (2009)

15. http://www.infoworld.com/d/security-central/gartner-seven-cloudcomputing-security-risks-853
16. Dlamini, M.T., Eloff, M.M., Eloff, J.H.P.: Internet of People, Things and Services, The Convergence of Security. Trust and Privacy (2009)
17. Kormann, M., Rubin, A.: Risks of the passport single sign on protocol. Computer Networks 33(1-6), 51–58 (2000)
18. Slemko, M.: Microsoft passport to trouble (2001), http://alive.znep.com/~marcs/passport/
19. Grob, T.: Security analysis of the SAML single sign on browser/artefact profile. In: Proc. 19th Annual Computer Security Applications Conference (2003)
20. Gajek, S., Schwenk, J., Steiner, M., Xuan, C.: Risks of the cardSpace protocol. In: Samarati, P., Yung, M., Martinelli, F., Ardagna, C.A. (eds.) ISC 2009. LNCS, vol. 5735, pp. 278–293. Springer, Heidelberg (2009)
21. Chen, X., Gajek, S., Schwenk, J.: On the Insecurity of Microsoft's Identity Metasystem CardSpace, Horst Görtz Institute for IT-Security, Tech. Rep. 3 (2008)
22. Bruegger, B.P., Hühnlein, D., Schwenk, J.: TLS Federation A secure and Relying-Party-friendly approach for Federated Identity Management. In: Proceedings of BIOSIG: Biometrics and Electronic Signatures. LNI, vol. 137, pp. 93–104 (2008)
23. Scavo, T.: SAML V2.0 Holder-of-Key Assertion Profile, Working Draft 09, http://www.oasis-open.org/apps/org/workgroup/security/download.php/30782/sstc-saml2-holder-of-key-draft-09.pdf
24. Gajek, S., Jager, T., Manulis, M., Schwenk, J.: A browser-based kerberos authentication scheme. In: Jajodia, S., Lopez, J. (eds.) ESORICS 2008. LNCS, vol. 5283, pp. 115–129. Springer, Heidelberg (2008)
25. Schwenk, J., Liao, L., Gajek, S.: Stronger Bindings for SAML Assertions and SAML Artifacts. In: Proceedings of the 5th ACM CCS Workshop on Secure Web Services (SWS 2008). ACM Press, New York (2008)

Effect of Noise on Recognition of Consonant-Vowel (CV) Units

Anil Kumar Vuppala[1], K. Sreenivasa Rao[2], and Saswat Chakrabarti[1]

[1] G. S. Sanyal School of Telecommunications
[2] School of Information Technology
Indian Institute of Technology Kharagpur
Kharagpur - 721302, West Bengal, India
anil.vuppala@gmail.com, ksrao@iitkgp.ac.in, saswat@ece.iitkgp.ernet.in

Abstract. This paper presents the experimental evaluation for recognition of consonant-vowel (CV) units under noise. Noise is one of the common degradation in real environments which strongly effects the performance of speech recognition system. In this work, initially effect of noise on recognition of CV units is studied by using two-stage CV recognition system proposed in our earlier studies. Later spectral processing based speech enhancement methods such as spectral subtraction and minimum mean square error (MMSE) are used for preprocessing to improve the CV recognition performance under noise. Performance of the CV recognition is studied on Telugu broadcast database for white and vehicle noise. Experimental results show that the speech enhancement techniques gives the improvement in the CV recognition performance under noise case.

Keywords: Recognition of consonant-vowel (CV) units, noisy speech recognition, speech enhancement, spectral subtraction, minimum mean square error (MMSE).

1 Introduction

The goal of automatic speech recognition is to convert speech into text. Commonly used approach for speech recognition is based on segmenting speech into subword units and labeling them using a subword unit recognizer [1]. Phonemes are widely used subword units of speech for speech recognition, but recent studies reveal that syllables (combinations of phonemes) are the suitable subword units for recognition [2],[3] in Indian languages. Context-dependent units such as syllables capture significant co-articulation effects and pronunciation variation compared to phonemes. In general, the syllable-like units are of type C^mVC^n , where C refers to consonant, V refers to a vowel, m and n refers to the number of consonants preceding and following in a syllable. Among these units, the CV units are the most frequently (around 90%) occurring basic units [2] in Indian languages, and hence CV units are considered to carry out this study.

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 191–200, 2011.

The issues involved in the recognition of CV units are the large number of CV classes and high similarity among several CV units. In literature Hidden Markov Models, Support Vector Machines, and Multi Layer FeedForward Neural Network (MLFFNN) are used for recognition of CV units in Indian languages [2,3,4]. As the number of CV classes are more, multi stage acoustic models may work better compared to monolithic (single level) acoustic models [2],[5]. So, in our earlier works we proposed a two-stage HMM and SVM based approach [5] for acoustic modeling of CV units. In the first stage of two-stage CV recognition, vowel category will be recognized using HMM models and at the second stage consonant category will be recognized using SVM models. Performance of two-stage CV recognition approach is observed to be superior compared to existed monolithic HMM or SVM based approaches.

In this work we analyzed the effect of noise on the recognition performance of CV units for two different noise types, white and vehicle noise at different signal-to-noise ratios (SNRs). CV units from Telugu broadcast database are used to carry out this study, and noise samples are collected from NOISEX-92 database. Later CV recognition performance under noise has shown to be improved by using spectral processing based enhancement techniques. Speech enhancement methods used in this study are spectral subtraction [6] and minimum mean square error (MMSE) [7].

Rest of the paper is organized as follows. Database used in this study is presented in section II. Two-stage CV recognition system and experimental setup used for this work are presented in section III. Experimental results for recognition of CV units under noise are discussed in section IV. Section V presents the speech enhancement methods used for this study and experimental results for recognition of CV units under noise by using preprocessing techniques. Summary and conclusions of the present work are mentioned in Section V.

2 Database

Database contains the Telugu broadcast news corpus developed at speech and vision lab, Indian Institute of Technology, Madras, India [2],[8]. Duration of Database is about five hours, collected over 20 sessions by 11 male speakers and 9 female speakers. Among 20 sessions, 15 sessions (8 male + 7 female) are used for training and 5 sessions (3 male + 2 female) are used for testing the CV recognizer. Manual marked syllable boundaries are available in Database. In the context of Indian languages context there are 145 (29 consonants and 5 vowels) most frequently occurred CV units [2]. In this work, among 145 CV units, 95 CV classes whose frequency of occurrence in the database is more than 50 are considered for the analysis and they are shown in Table 1. Among 95 CV units, *a*, *e*, *i*, *o* and *u* vowel groups contains 26, 16, 22, 10, and 21 consonant classes respectively. We consider both short and long vowels as one vowel only, as it is very difficult to recognize short and long vowels in the continuous speech. Simple language model will take care of short and long vowels during speech

Table 1. List of 95 CV units from Telugu broadcast news corpus

Vowel class	CV units
a subclass	ka, cha, Ta, ta, pa kha, Tha, tha, pha ga, ja, Da, da, ba gha, dha, bha, na, ma ya, ra, la, va, ha, sha, sa
e subclass	ke, che, Te, te, pe phe, je, De, de, ne, me ye, re, le, ve, se
i subclass	ki, chi, Ti, ti, pi thi, gi, ji, Di, di, bi dhi, bhi, ni, mi, yi, ri li, vi, hi, shi, si
o subclass	ko, cho, to, po, do mo, yo, ro, lo, so
u subclass	ku, chu, Tu, tu, pu thu, gu, ju, Du, du, bu dhu, bhu, nu, mu, yu ru, lu, vu, shu, su

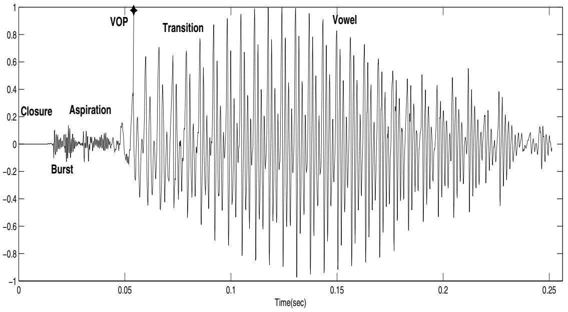


Fig. 1. Regions of significant events in the production of the CV unit */ka/* [5]

recognition. In Fig 1 an example CV unit */ka/* is shown. Different regions of significant events in the production of the CV unit */ka/* along with vowel onset point (VOP) [5] are also shown in Fig 1.

3 Two-Stage CV Recognizer [5]

In two-stage CV recognition approach, 95 CV units considered in this study are divided into five subclasses based on vowel to reduce the influence of vowel on recognition of CV units. Two-stage CV recognizer is shown in shown Fig 2. In the first level of CV recognition, vowel category will be recognized and at

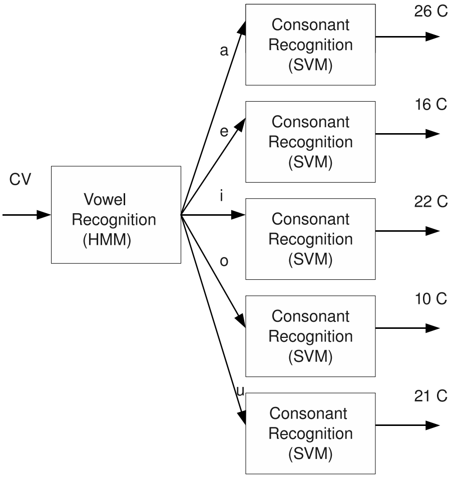

CV
Vowel Recognition (HMM)
a
e
i
o
u
Consonant Recognition (SVM)
Consonant Recognition (SVM)
Consonant Recognition (SVM)
Consonant Recognition (SVM)
Consonant Recognition (SVM)
26 C
16 C
22 C
10 C
21 C

Fig. 2. Two stage CV recognition system [5]

the second level consonant category will be recognized. Total number of CV utterances considered are 52,703 (38,729 are used for training, and 13,974 are used for testing), and it is more than 95 % of CV units present in the Database.

Vowel onset point (VOP) plays an anchor role in selecting the features for two-stage CV recognition system. Performance of VOP detection method presented in [9] is superior among the existed VOP detection methods, so we consider that method in this study for detecting the VOPs. This VOP detection method uses combined evidences from excitation source, spectral peaks, and modulation spectrum energies [9] for robust VOP detection.

Vowel recognition models in two-stage CV recognition system are trained using features extracted from VOP to end of CV segment and consonant models are trained using features extracted from beginning of CV segment to 40 ms in addition to VOP. In this work, 40 ms in addition to VOP is considered as transition region. Features extracted from both consonant and transition region are used for consonant recognition. In two-stage CV recognizer, HMM models are used for vowel recognition and SVM models for consonant recognition [5]. HMM models perform better for vowel recognition, because HMMs are good at capturing the state sequence corresponds to the sequence of vocal tract shapes. The sequences of vocal tract shapes are unique to each vowel. SVM models perform better for consonant recognition, because SVM models are trained using one against rest approach to capture the discriminative information present in highly similar consonant classes.

Thirteenth order Mel-frequency cepstral coefficients (MFCC) [10] extracted from every 20 ms of CV segment with 5 ms frame shift are used for developing the acoustic models. HMM models are developed using maximum likelihood approach using HMM tool kit (htk) [11]. Feature vectors of size 39 dimension (13 MFCC + delta + delta-delta coefficients) are used for developing of HMM models. In the proposed method HMM models are developed using 3 states and 64 mixtures.

SVM models are developed using one against the rest approach using open source SVMTorch [12]. Fixed pattern length of 10 and Gaussian kernel with standard deviation of 40 are used to build SVM models. Fixed pattern length is obtained by using below equation.

$$s = (p * SL)/PL, \quad p = 0, 1, ..., PL - 1, and \\ s = 0, 1, ..., SL - 1. \tag{1}$$

Where PL is pattern length, and SL is segment length. If segment length SL is greater than PL, few frames of the segment are omitted. If the segment length SL is smaller than PL, few frames of the segment are repeated. So, from each CV utterance a fixed 130 (10 PL * 13 MFCC = 130) dimension feature vector is extracted for developing the SVM models.

Performance of the two-stage CV recognizer using Telugu broadcast database is presented in Table 2. Vowel recognition performance is around 87.22%, and average consonant recognition performance for different vowel group consonants is around 66.38%. So, overall CV recognition performance for two-stage CV recognizer is around 57.89%.

Table 2. CV recognition performance using two-stage CV recognition system for Telugu broadcast database

Category	Recognition (%)
Vowel category (1^{st} stage)	87.22
Consonant category (2^{nd} stage)	66.38
Overall (1^{st} + 2^{nd} stage)	57.89

4 Effect of Background Noise on Recognition of CV Units

Effect of noise on the performance of two-stage CV recognition system is studied by using Telugu broadcast database at different signal-to-noise ratios (SNRs) for two different noise types. Noises considered in this study are white and vehicle noises from NOISEX-92 database. White and vehicle noise added speech signals for speech utterance /*bharata padavi dalalu*/ are shown in Fig. 3 along with corresponding spectrogram plots.

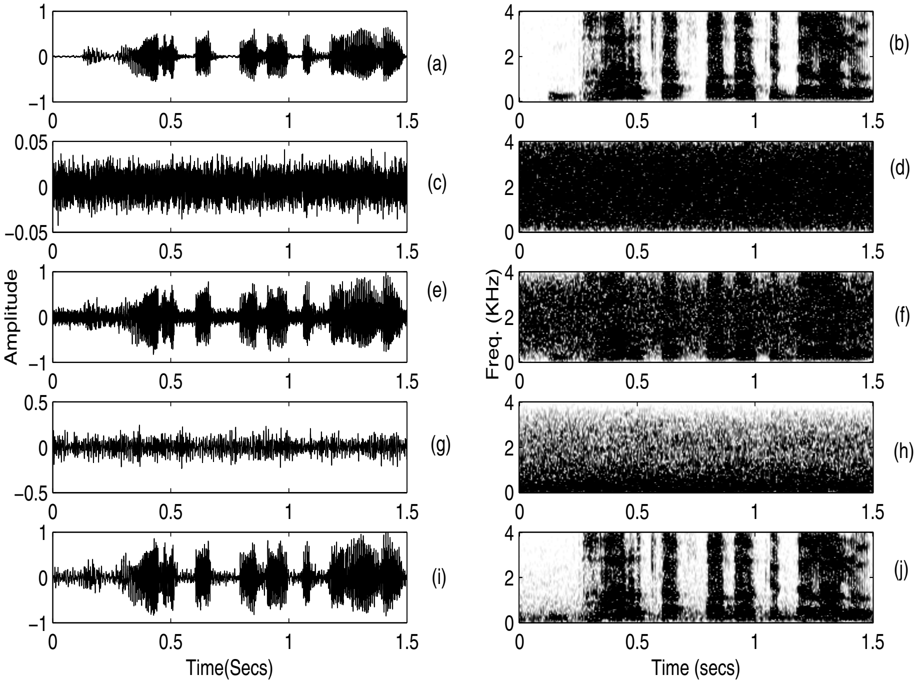


Fig. 3. Noisy speech: (a,b) clean speech and its spectrogram; (c,d) white noise and its spectrogram; (e,f) white noise (SNR 10 dB) added speech and its spectrogram; (g,h) vehicle noise and its spectrogram; (i,j) vehicle noise (SNR 10 dB) added speech and its spectrogram;

4.1 Experimental Results and Discussion

Table 3 shows the vowel recognition performance from CV units by using HMM acoustic models trained with clean speech and tested with noisy speech. Column-1 in Table 3 indicates the speech enhancement techniques used in this study. Columns-2–6 indicates the CV recognition performance under different SNR levels (varies from 0–30 dB) of white and vehicle noises. Similarly overall consonant recognition and CV recognition performance are shown in Tables 4 and 5. From the results we can observe that, CV recognition performance is reducing significantly due to noise. White noise is effecting more compare to vehicle noise. Noise strongly effecting the CV recognition performance at lower SNRs.

Table 3. Performance of vowel recognition from CV units using HMM acoustic models trained with clean speech under different noise cases

	Recognition performance (%) Clean performance **87.22**				
	different SNR levels in dB				
Noise	0	5	10	20	30
White	21.08	33.08	43.8	76.82	81.13
Vehicle	72.66	77.41	81.66	82.72	85.31

Table 4. Performance of overall consonant recognition from CV units using SVM acoustic models trained with clean speech under different noise cases

	Recognition performance (%) Clean performance **66.38**				
	different SNR levels in dB				
Noise	0	5	10	20	30
White	15.96	20.76	27.06	38.75	44.98
Vehicle	28.17	36.88	44.78	54.23	58.38

Table 5. Overall CV recognition using two-stage HMM–SVM acoustic models trained with clean speech under different background noise cases

	Recognition performance (%) Clean performance **57.89**				
	different SNR levels in dB				
Noise	0	5	10	20	30
White	3.36	6.87	11.85	29.77	36.49
Vehicle	20.47	28.55	36.57	44.86	49.80

5 Speech Enhancement Techniques for Improving CV Recognition Performance under Background Noise

Various methods have been proposed in the literature to overcome the noise effect on speech recognition. Generally, this can be accomplished in different stages of speech recognition system as follows.

1. Robustness at the signal level: In this approach noisy speech signals are enhanced before the feature extraction stage.
2. Robustness at the feature level: The features representing the speech signal are designed in order to be less sensitive to the noisy degraded conditions.
3. Robustness at the acoustic model level: The aim of the model compensation approach is to determine the influence of the noise degradation on the distribution of the speech features and to modify the models used in the recognition to take into account about the influence of the degradation.

This work aims to provide the robustness at the signal level using speech enhancement methods as a preprocessing stage. Speech enhancement methods used in this study are Spectral subtraction (SS) and minimum mean square error (MMSE).

Spectral subtraction based speech enhancement is performed by subtracting the average magnitude of the noise spectrum from the spectrum of the noisy speech [6]. In this method noise is assumed to be uncorrelated and additive to the speech signal. The noise estimation is obtained based on the assumption that the noise is locally stationary, so that the noise characteristics computed

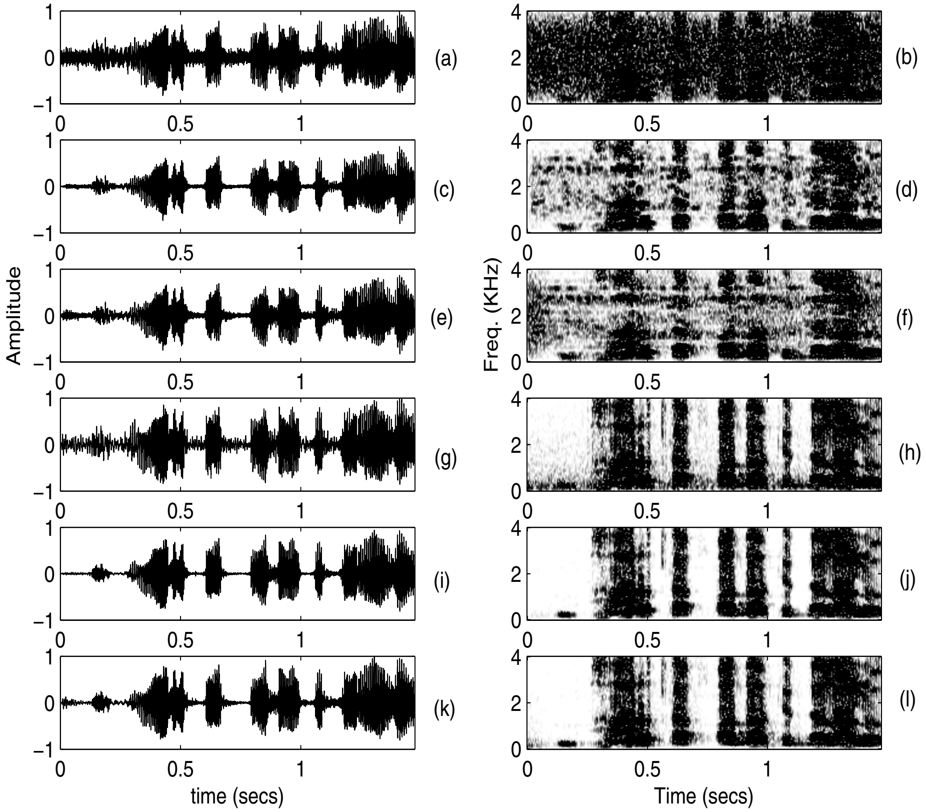


Fig. 4. Noisy speech enhancement:(a,b) white noisy (SNR of 10dB) speech and its spectrogram; (c,d) white noisy speech processed by spectral subtraction method and its spectrogram; (e,f) white noisy speech processed by MMSE method and its spectrogram; (g,h) vehicle noisy (SNR of 10dB) speech and its spectrogram; (i,j) vehicle noisy speech processed by spectral subtraction method and its spectrogram; (k,l) vehicle noisy speech processed by MMSE method and its spectrogram.

during the speech pauses are a good approximation to the noise characteristics. The MMSE short time spectral amplitude estimator (STSA) for speech enhancement aims to minimize the mean square error between the short time spectral magnitude of the clean and enhanced speech signal [7]. This method assumes that each of the Fourier expansion coefficients of the speech and of the noise process can be modeled as independent, zero-mean and Gaussian random variables. CV recognition performance under noise case by using speech enhancement preprocessing stage is presented in the following subsection. Enhanced speech along with spectrograms of white and vehicle noise added speech by using SS and MMSE methods are shown in Fig. 4.

5.1 Experimental Results and Discussion

Table 6 shows the overall CV recognition performance using two-stage acoustic models trained with clean speech and tested with enhanced speech. Column-1 in Table 6 indicates the speech enhancement techniques used in this study. Columns-2–6 indicates the CV recognition performance under different SNR levels of white and vehicle noises. From the experimental results we can observe that, CV recognition performance has improved significantly by using speech enhancement techniques. From the results we can also observe that, around 15% and 17% average recognition performance improvement by using spectral subtraction and MMSE methods respectively.

Table 6. Overall CV recognition performance using two-stage acoustic models trained with clean speech and tested with different background noise cases after using speech enhancement techniques. In table abbreviations DEG, SS and MMSE refer to degraded speech, multi band spectral subtraction and MMSE-STSA estimator.

	Recognition performance (%) Clean performance is **57.89**				
	different SNR levels in dB White noise				
Enhancement	0	5	10	20	30
DEG	3.36	6.87	11.85	29.77	36.49
SS	22.12	28.25	32.11	36.36	43.97
MMSE	24.81	31.20	34.42	39.06	44.76
	Vehicle noise				
DEG	20.47	28.55	36.57	44.86	49.80
SS	29.23	37.34	40.81	47.20	51.21
MMSE	31.07	38.59	42.75	48.31	52.08

6 Summary and Conclusions

In this paper we presented the performance of two-stage (HMM-SVM) CV recognition system under two noises (white and vehicle) at different SNR levels. From the results it is evident that, recognition performance of CV units is decreasing as SNR level decreasing. Further, performance of CV recognition under noise is studied by using speech enhancement as preprocessing. Spectral subtraction and MMSE methods are used for enhancing the degraded speech. MMSE method is giving marginally higher CV recognition performance compared to spectral subtraction method in case of both white and vehicle noises. Present study can be further extended to exploring feature level and model level compensation techniques for noise for the improvement of CV recognition performance under noise.

References

1. Rabiner, L.R., Juang, B.H.: Fundamentals of speech recognition. PTR Prentice Hall, Englewood cliffs (1993)
2. Gangashetty, S.V.: Neural network models for recognition of consonant-vowel units of speech in Multiple Languages. PhD thesis, IIT Madras (October 2004)
3. Chandra Sekhar, C.: Neural Network models for recognition of stop consonant-vowel (SCV) segments in continuous speech. PhD thesis, IIT Madras (1996)
4. Chandra Sekhar, C., Lee, W.F., Takeda, K., Itakura, F.: Acoustic modeling of subword units using support vector machines. In: WSLP (2003)
5. Vuppala, A.K., Chakrabarti, S., Rao, K.S.: Effect of speech coding on recognition of Consonant-Vowel (CV) units. In: Proc. Int. Conf. contemporary computing (Springer Communications in Computer and Information Science, pp. 284–294 (August 2010) ISSN: 1865-0929
6. Suppression, B.S.F.: of acoustic noise in speech using spectral subtraction. IEEE Trans. Acoust., Speech, Signal Process 27, 113–120 (1979)
7. Ephrain, Y., Malah, D.: Speech enhancement using minimum mean square error short-time spectral amplitude estimator. IEEE Trans. Acoust., Speech, Signal Process 32, 1109–1121 (1984)
8. Suryakanth, V.G., Sekhar, C.C., Yegnanarayana, B.: Spotting multilingual consonant-vowel units of speech using neural networks. In: An ISCA tutorial and research workshop on non-linear speech processing, pp. 287–297 (April 2005)
9. Prasanna, S.R.M., Reddy, B.V.S., Krishnamoorthy, P.: Vowel onset point detection using source, spectral peaks, and modulation spectrum energies. IEEE Transactions on audio, speech, and language processing 17(4), 556–565 (2009)
10. Joseph, W.P.: Signal modeling techniques in speech recognition. Proceedings of the IEEE 81, 1215–1247 (1993)
11. Young, S., Kershaw, D., Odell, J., Ollason, D., Valtchev, V., Woodland, P.: The HTK Book Version 3.0. Cambridge University Press, Cambridge (2000)
12. Collobert, R., Bengio, S.: SVMTorch, support vector machines for large-scale regression problems. J. Mach. Learn. Res. 1, 143–160 (2001)

Effect of Noise on Vowel Onset Point Detection

Anil Kumar Vuppala[1], Jainath Yadav[2], K. Sreenivasa Rao[2],
and Saswat Chakrabarti[1]

[1] G. S. Sanyal School of Telecommunications
[2] School of Information Technology
Indian Institute of Technology Kharagpur
Kharagpur - 721302, West Bengal, India
anil.vuppala@gmail.com, jaibhu38@gmail.com, ksrao@iitkgp.ac.in,
saswat@ece.iitkgp.ernet.in

Abstract. This paper discuss the effect of noise on vowel onset point (VOP) detection performance. Noise is one of the major degradation in real-time environments. In this work, initially effect of noise on VOP detection is studied by using recently developed VOP detection method. In this method, VOPs are detected by combining the complementary evidence from excitation source, spectral peaks and modulation spectrum to improve VOP detection performance. Later spectral processing based speech enhancement methods such as spectral subtraction and minimum mean square error (MMSE) are used for preprocessing to improve the VOP detection performance under noise. Performance of the VOP detection is analyzed by using TIMIT database for white and vehicle noise. In general, performance of VOP detection is degraded due to noise and in particular performance is effected significantly due to spurious VOPs introduced at low SNR values. Experimental results indicate that the speech enhancement techniques provides the improvement in the VOP detection performance by eliminating spurious VOPs under noise.

Keywords: Vowel onset point (VOP), excitation source, spectral peaks, modulation spectrum, speech enhancement, spectral subtraction, minimum mean square error (MMSE).

1 Introduction

Vowel onset point (VOP) is the instant at which the onset of vowel takes place in speech signal. VOP plays an anchor role in applications such as consonant-vowel (CV) unit recognition, speaker recognition, speech segmentation and speech rate modification [1],[2],[3],[4],and [5]. There are various methods developed in literature for the detection of VOPs. VOP detection methods available in literature are based on raising trend of resonance peaks in the amplitude spectrum [6]; zero-crossing rate, energy and pitch information [8]; wavelet transform [7]; neural network [3]; dynamic time warping (DTW) [9]; and excitation source information [10]. Recently, a method has been proposed by combining the complementary evidences from excitation source, spectral peaks, and modulation spectrum [1]

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 201–211, 2011.

for the detection of VOP. In this work this VOP detection method is termed as combined method. Performance of the other existing methods are inferior compared to the recent combined method. Hence, combined method is used in the present study.

In real-time environments noise is one of the major degradation, so in this work we analyzed the effect of noise on detection of VOP for white and vehicle noise at different signal-to-noise ratios (SNRs). TIMIT database is used to carry out this study, and noise samples are collected from NOISEX database. Later VOP detection performance under noise has shown to be improved by using spectral processing based enhancement techniques. Speech enhancement methods used in this study are spectral subtraction [11] and minimum mean square error (MMSE) [12].

This paper is organized as follows. Section II describes the detection of VOPs by combining the evidences from source, spectral peaks, and modulation spectrum. Experimental results for detection of VOPs under noise are discussed in section III. Section IV presents the speech enhancement methods used for this study and experimental results for detection of VOPs under noise by using preprocessing techniques. Summary and conclusions of the present work are mentioned in Section V.

2 VOP Detection Using Combined Method [1]

Vowel onset point detection algorithm presented in [1] uses combined evidence from excitation source, spectral peaks, and modulation spectrum energies. Excitation source information is represented using the Hilbert envelope (HE) of the linear prediction (LP) residual. Vocal tract shape can be represented by using the sum of ten largest peaks of discrete Fourier transform (DFT) spectrum. Slowly varying temporal envelope of speech signal can be represented using modulation spectrum. Each of these three features represents complementary information about the VOP. So, the individual evidences are combined for the detection of VOPs.

2.1 VOP Detection Using Excitation Source Information[1],[10]

VOP detection using excitation source information is carried out in following sequence of steps. (1) Find the linear prediction (LP) residual (also known as excitation source) of speech signal. (2) Find the Hilbert envelope of the LP residual. (3) Smooth the HE of the LP residual by convolving with a Hamming window of 50 ms. (4) The change at the VOP is available in the smoothed HE of the LP residual, and is further enhanced by computing its slope with the help of a first-order difference (FOD). (5) These enhanced values are convolved with the first order Gaussian difference (FOGD) operator and the convolved output is the VOP Evidence Plot using excitation source. VOP evidence plot using excitation source for speech signal */Don't ask me to carry an oily rag like that/* is shown in Fig 1(b).

2.2 VOP Detection Using Spectral Peaks Energy [1]

VOP detection using spectral peaks energy is carried out in following sequence of steps. (1) For each block of 20 ms (with a shift of 10ms) of speech signal a 256-point DFT is computed, and the ten largest peaks are selected from the first 128 points. (2) The sum of these amplitudes is plotted as a function of time, and it is the representation of energy of spectral peaks. VOP can be observed as significant change in the sum of ten peaks in the DFT spectrum. (4) The change at the VOP available in the spectrum of the speech signal is further enhanced by computing its slope with the help of a first-order difference (FOD). (5) These enhanced values are convolved with the FOGD operator and the convolved output is the VOP Evidence Plot using Spectral Peaks. VOP evidence plot using spectral peaks energy for speech signal */Don't ask me to carry an oily rag like that/* is shown in Fig 1(c).

2.3 VOP Detection Using Modulation Spectrum Energy [1]

Modulation components refer to the slowly varying temporal envelope components in speech. The temporal envelope of speech is dominated by low-frequency components of several Hz. Detection of VOP using modulation spectrum energy is carried out in following sequence of steps. (1) The speech signal is analyzed using approximately 18 trapezoidal critical band filters (minimal overlap) between 0 and 4 kHz. (2) In each band, an amplitude envelope signal is computed by half-wave rectification and low pass filtering with cutoff frequency of 28 Hz. (3) Each amplitude envelope signal is then down sampled to 80 samples/s and normalized by the average envelope level in that channel, measured over the entire utterance. (4) The modulations of the normalized envelope signals are analyzed by computing the DFT over 250-ms Hamming windows with shift of 12.5 ms, in order to capture the dynamic properties of the signal. (5) Finally, the 4–16 Hz components are added together, across all critical bands to plot modulation spectrum energy. (6) The change at the VOP available in the modulation spectrum energy is further enhanced by computing its slope with the help of a first-order difference (FOD). (7) These enhanced values are convolved with the First Order Gaussian Difference (FOGD) operator and the convolved output is the VOP Evidence Plot using modulation spectrum energy. VOP evidence plot using modulation spectrum energy for speech signal */Don't ask me to carry an oily rag like that/* is shown in Fig 1(d).

2.4 VOP Detection Using Combined Method [1]

Each of the above three methods uses complementary information present in the speech signal. Combined method combines the independent evidences from excitation source, spectral peaks, and modulation spectrum energies. So the combined method may have stronger and robust information about the VOPs.

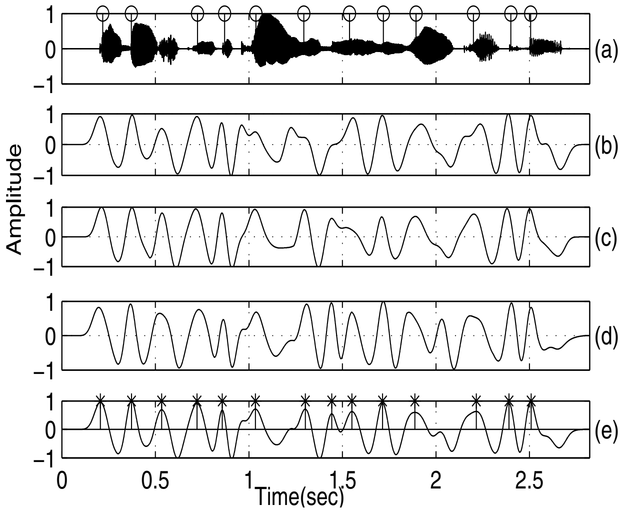


Fig. 1. VOP detection using combination of all three evidences. (a) Speech signal. VOP evidence plot for (b) excitation source. (c) Spectral peaks. (d) Modulation spectrum. (e) Combined VOP evidence plot [(b)+(c)+(d)].

This will lead to improved performance. VOP detection using individual and combination of all three evidences for speech signal /*Don't ask me to carry an oily rag like that*/ are shown in Fig 1.

Experiments are conducted to analyze the performance of VOP detection using TIMIT data base [13]. About 220 sentences (120 sentences are spoken by female speakers and 100 sentences are spoken by male speakers) having 2407 manually marked VOPs are considered for analyzing the performance of the proposed VOP detection method. Among 2407 VOPs, 1013 VOPs correspond to the utterances spoken by male speakers, and the rest 1394 VOPs correspond to the utterances spoken by female speakers. The VOPs hypothesised within a deviation of 40 ms from actual VOPs are considered as the detected VOPs. If there are no hypothesised VOPs within 40 ms deviation from actual VOPs, then those are considered as missed VOPs. Hypothesised VOPs beyond 25 ms around the actual VOPs are considered as spurious ones. Table 1 shows the accuracy in detection of VOPs using different methods. Column-1 indicates different methods considered in the analysis for detecting the VOPs. Columns 2 indicate the percentage of VOPs detected within the 40 ms deviation. Columns 3 and 4 indicate the % of missed and spurious VOPs respectively. From the results it is evident that combined method is working better compare to individual methods. Approximate VOP detection performance using combined method is 96 % with in ± 40 ms deviation.

Table 1. Performance of VOP detection using excitation source (EXC), spectral peaks (VT), modulation spectrum (MOD) and combined (COMB) methods on TIMIT database. All are expressed as % of the ratio of respective VOPs to the total number of VOPs 2407.

VOP detection method	VOPs detected within 40 ms	Missed VOPs	Spurious VOPs
EXC	95	5	4
VT	94	6	5
MOD	92	8	2
COMB	96	4	3

3 Effect of Noise on Vowel Onset Point Detection

This section describes the effect of noise on VOP detection using combined method. Effect of noise on speech utterance is illustrated in Fig. 2 with the help of spectrogram. Fig. 2 shows the white and vehicle noise added speech signals for speech utterance */Don't ask me to carry an oily rag like/* along with their spectrogram plots. From Fig. 2, we can observe that white noise is effecting more compared to vehicle noise.

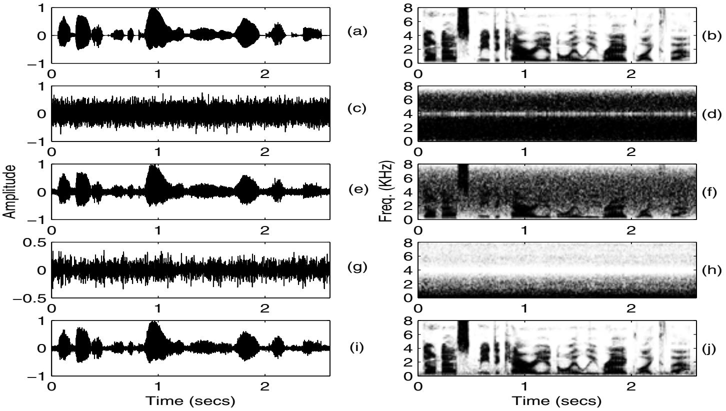


Fig. 2. Noisy speech: (a,b) clean speech and its spectrogram; (c,d) white noise and its spectrogram; (e,f) white noise (SNR 10 dB) added speech and its spectrogram; (g,h) vehicle noise and its spectrogram; (i,j) vehicle noise (SNR 10 dB) added speech and its spectrogram;

Fig. 3 shows the VOP detection using combined method under noise. Fig. 3(a) shows the speech utterance */Don't ask me to carry an oily rag like/*, and Fig. 3(c) and (e) shows the white and vehicle noise (of SNR 5 dB) added speech utterances. Combined VOP evidence plot with hypothesised VOPs for Fig. 3(a),(c) and (e) are shown in Fig. 3(b),(d) and (f) respectively. Hypothesised VOPs indicated in red color are spurious VOPs. From 3 (b),(d) and (f) we can observe that spurious VOPs are increasing due to noise. We can also observe that number of spurious VOPs are more in case of white noise compared to vehicle noise.

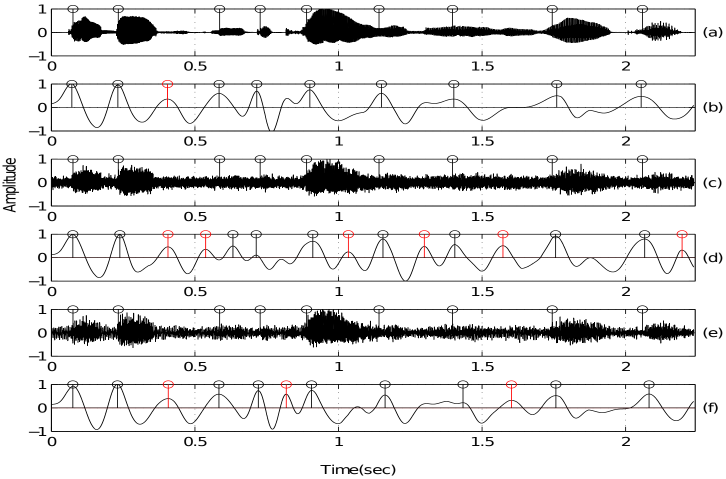


Fig. 3. VOP detection using combined method under noise: (a), (c) and (e) are clean, white noise added (SNR of 5 dB) and vehicle noise added (SNR of 5 dB) added speech signals respectively with manually marked VOPs. (b), (d) and (f) are combined VOP evidence plot for (a), (c) and (e) respectively with hypothesised VOPs.

Effect of noise on the performance of combined VOP detection method is studied for two different noise types by using TIMIT database with 2407 manually marked VOPs as described in previous section. Noises considered in this study are white and vehicle noises from Noisex-92 database [14] at SNR levels of 0, 5, 10, 20 dBs. Table 2 shows the performance of VOP detection using combined (COMB) method on TIMIT database under noise. From results, we can observe that VOP detection performance is degrading due to noise and noise effect is prominent in-terms of spurious VOPs (see column-4 in Table 2). We can also observe that white noise effect is more compared to vehicle noise (see in Table 2). Spurious VOPs degrade the performance significantly in VOP applications like speech segmentation and speech rate modification, so in this work we try to remove spurious VOPs by using speech enhancement techniques.

Table 2. Performance of VOP detection using combined (COMB) method on TIMIT database (2407 genuine VOPs) under noise. All are expressed as % of the ratio of respective VOPs to the total number of VOPs 2407.

Testing	VOPs detected within 40 ms	Missed VOPs	Spurious VOPs
Clean	96	4	3
SNR (dB)	White noise		
0	85	15	39
5	88	12	32
10	92	8	30
20	94	6	18
	Vehicle noise		
0	89	11	32
5	93	7	28
10	95	5	25
20	96	4	15

4 Speech Enhancement Techniques for Improving Vowel Onset Point Detection Performance under Noise

In previous section we observed the effect of noise on performance of VOP detection. This section presents the performance of VOP detection using combined method under noise with speech enhancement techniques. Speech enhancement techniques used in this study are spectral subtraction and MMSE methods. Spectral subtraction based speech enhancement is performed by subtracting the average magnitude of the noise spectrum from the spectrum of the noisy speech [11]. In this method noise is assumed to be uncorrelated and additive to the speech signal. The noise estimation is obtained based on the assumption that the noise is locally stationary, so that the noise characteristics computed during the speech pauses are a good approximation to the noise characteristics.

The MMSE short time spectral amplitude estimator (STSA) for speech enhancement aims to minimize the mean square error between the short time spectral magnitude of the clean and enhanced speech signal [12]. This method assumes that each of the Fourier expansion coefficients of the speech and of the noise process can be modeled as independent, zero-mean and Gaussian random variables. Fig. 4 shows the enhanced speech of white and vehicle noise added speech (SNR of 10 dB) by using SS and MMSE methods along with their spectrograms.

VOP detection for white and vehicle noise (SNR of 5 dB) added speech utterance /*Don't ask me to carry an oily rag like*/ by using combined method with and with out speech enhancement are shown in Fig. 5 and Fig. 6 respectively. Fig. 5 (a), (c) and (e) shows the white noisy (SNR of 5 dB) added speech utterance, speech utterance enhanced using spectral subtraction and MSSE respectively. Fig. 5 (b), (d) and (f) shows the combined VOP evidence plot for (a),

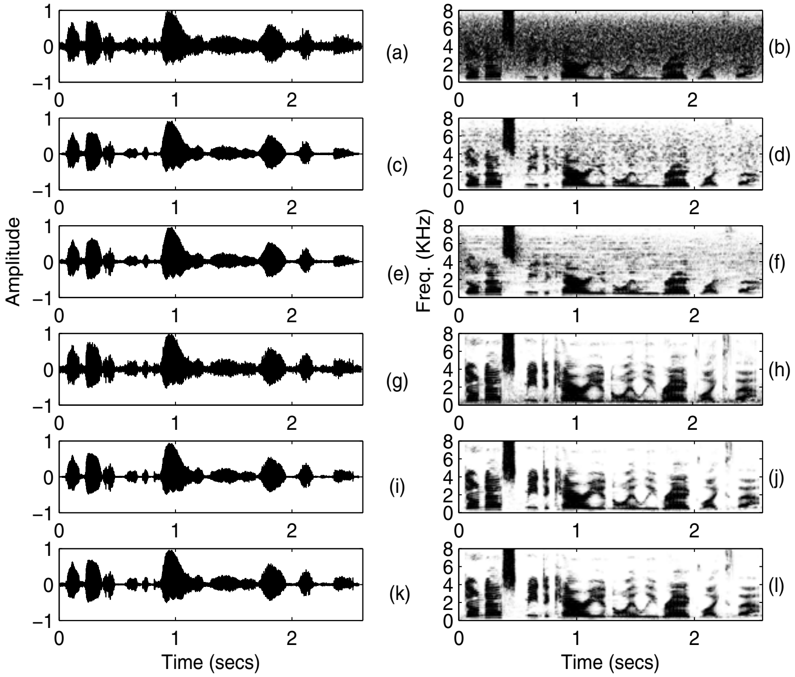


Fig. 4. Noisy speech enhancement:(a,b) white noise (SNR of 10 dB) speech and its spectrogram; (c,d) white noisy speech processed by spectral subtraction method and its spectrogram; (e,f) white noisy speech processed by MMSE method and its spectrogram; (g,h) vehicle noisy (SNR of 10 dB) speech and its spectrogram; (i,j) vehicle noisy speech processed by spectral subtraction method and its spectrogram; (k,l) vehicle noisy speech processed by MMSE method and its spectrogram

(c) and (e) respectively with hypothesised VOPs. From Fig. 5 (b), (d) and (f), we can observe that spurious VOPs are reducing by using speech enhancement preprocessing techniques. Similar results are shown in Fig. 6 for vehicle noise added speech utterance.

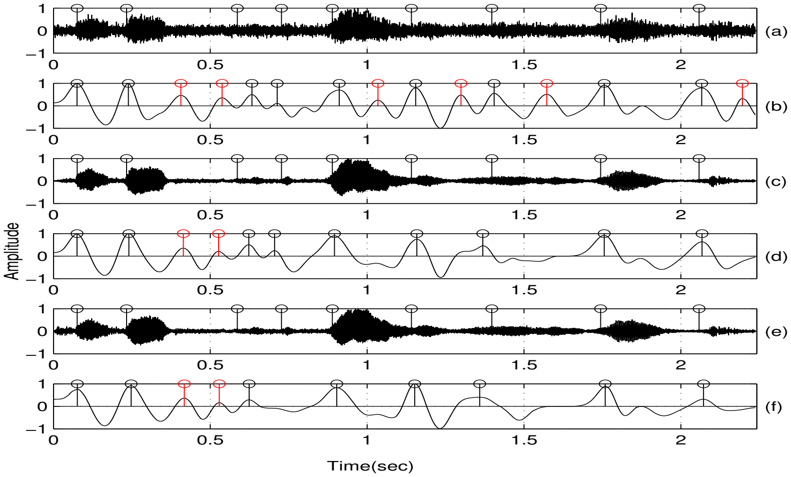


Fig. 5. VOP detection for white noise added speech signal by using combined method with and with out speech enhancement: (a), (c) and (e) are white noisy (SNR of 5 dB) added speech utterance, speech utterance enhanced using spectral subtraction and MSSE respectively with manually marked VOPs; (b), (d) and (f) are combined VOP evidence plot for (a), (c) and (e) respectively with hypothesised VOPs

Performance of VOP detection using combined method on TIMIT database under noise after using speech enhancement based preprocessing techniques is shown in Table 3. From column 4, 7, and 10 in Table 3 we can observe that, spurious VOPs are reduced significantly by using speech enhancement based preprocessing techniques, but VOP detection with in 40 ms performance is reducing by using preprocessing techniques (see column 2, 5, and 8). This may be due to the missing of finite VOP evidences in speech signal while using speech enhancement techniques to reduce the noise. From results, we can also observe that VOP detection performance under noise by using preprocessing based on MMSE method is slightly better compared to spectral subtraction method.

5 Summary and Conclusion

Effect of noise on performance of VOP detection was presented in this paper. Noise effect on VOP detection was studied by using recently developed combined VOP detection method. Experiments are conducted using TIMIT database for

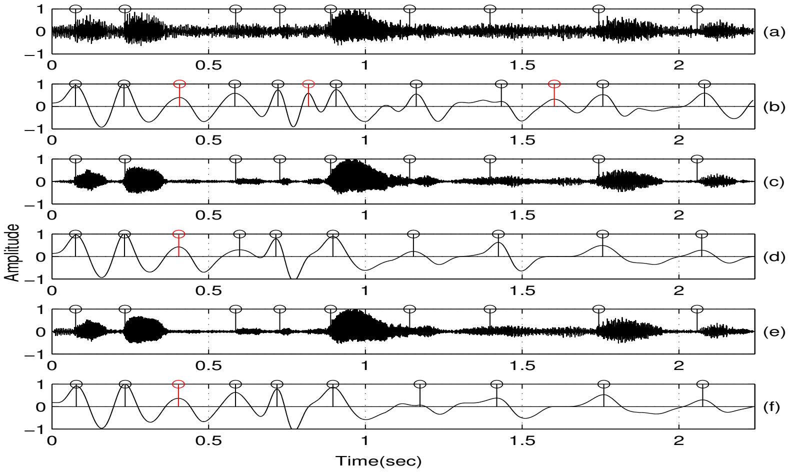


Fig. 6. VOP detection for vehicle noise added speech signal by using combined method with and with out speech enhancement: (a), (c) and (e) are white noisy (SNR of 5 dB) added speech utterance, speech utterance enhanced using spectral subtraction and MSSE respectively with manually marked VOPs; (b), (d) and (f) are combined VOP evidence plot for (a), (c) and (e) respectively with hypothesised VOPs

Table 3. Performance of VOP detection using combined (COMB) method on TIMIT database under noise after using speech enhancement techniques. All are expressed as % of the ratio of respective VOPs to the total number of VOPs 2407.

	Without enhancement			With spectral subtraction			With MMSE estimator		
Testing SNR dB	VOPs det. 40 ms 40 ms	Miss. VOPs	Spu. VOPs s	VOPs det. within 40 ms	Miss. VOPs	Spu. VOPs	VOPs det. within 40 ms	Miss. VOPs	Spu. VOPs
	White noise								
0	85	15	39	84	16	8	82	18	6
5	88	12	32	87	13	6	86	14	5
10	92	8	30	91	9	4	87	13	4
20	94	6	18	93	7	3	90	10	3
	Vehicle noise								
0	89	11	32	86	14	7	84	16	5
5	93	7	28	92	8	5	88	12	5
10	95	5	25	94	6	4	90	10	3
20	96	4	15	95	5	3	92	8	3

white and vehicle noise. From experimental results, we observed that VOP detection performance is reducing due to noise and spurious VOPs are more at low SNR values. Spectral processing based speech enhancement methods such as spectral subtraction and minimum mean square error (MMSE) were used for preprocessing to improve the VOP detection performance under noise. By using preprocessing techniques number of spurious VOPs were reduced significantly. We observed that VOP detection performance under noise by using preprocessing based on MMSE method was slightly better compared to spectral subtraction method. We also observed that, VOP detection performance with in 40 ms deviation is reducing while using speech enhancement techniques to reduce the noise. Hence, new VOP detection methods need to explored to remove spurious VOPs as well as improving the VOP detection performance under noise.

References

1. Prasanna, S.R.M., Reddy, B.V.S., Krishnamoorthy, P.: Vowel onset point detection using source, spectral peaks, and modulation spectrum energies. IEEE Transactions on audio, speech, and language processing 17(4), 556–565 (2009)
2. Prasanna, S.R.M., Suryakanth, V.G., Yegnanarayana, B.: Significance of vowel onset point for speech analysis. In: Signal Processing and Communications (Biennial Conf., IISc), pp. 81–88 (2001)
3. Suryakanth, V.G., Sekhar, C.C., Yegnanarayana, B.: Detection of vowel onset points in continuous speech using autoassociative neural network models. In: Proc. Int. Conf. Spoken Language Processing, pp. 401–410 (October 2004)
4. Suryakanth, V.G., Sekhar, C.C., Yegnanarayana, B.: Spotting multilingual consonant-vowel units of speech using neural networks. In: An ISCA Tutorial and Research Workshop on Non-linear Speech Processing, pp. 287–297 (April 2005)
5. Rao, K.S., Yegnanarayana, B.: Duration modification using glottal closure instants and vowel onset points. Speech Communication 51, 1263–1269 (2009)
6. Hermes, D.J.: Vowel onset detection. J. Acoust. Soc. Amer. 87, 866–873 (1990)
7. Wang, J.-H., Chen, S.-H.: A C/V segmentation algorithm for Mandarin speech using wavelet transforms. In: Proc. Int. Conf. Acoust. Speech, Signal Process., vol.1, pp. 1261–1264 (September 1999)
8. Wang, J.-F., Wu, C.-H., Chang, S.-H., Lee, J.-Y.: A hierarchical neural network based on C/V segmentation algorithm for Isolated Mandarin speech recognition. IEEE Trans. Signal Process. 39(9), 2141–2146 (1991)
9. Suryakanth, V.G., Sekhar, C.C., Yegnanarayana, B.: Extraction of fixed dimension patterns from varying duration segments of consonant-vowel utterances. In: Proc. IEEE ICISIP, pp. 159–164 (2004)
10. Prasanna, S.R.M., Yegnanarayana, B.: Detection of vowel onset point events using excitation source information. In: Proc. of Interspeech, pp. 1133–1136 (2005)
11. Boll, S.F.: Suppression of acoustic noise in speech using spectral subtraction. IEEE Trans. Acoust., Speech, Signal Process 27, 113–120 (1979)
12. Ephrain, Y., Malah, D.: Speech enhancement using minimum mean square error short-time spectral amplitude estimator. IEEE Trans. Acoust., Speech, Signal Process 32, 1109–1121 (1984)
13. Garofolo, J.S.: TIMIT Acoustic-Phonetic Continuous Speech Corpus Linguistic Data Consortium, Philadelphia, PA (1993)
14. Noisex-92, http://www.speech.cs.cmu.edu/comp.speech/Section1/Data/noisex.html

DBCCOM: Density Based Clustering with Constraints and Obstacle Modeling

Neelam Duhan and A.K. Sharma

Department of Computer Engineering,
YMCA University of Science and Technology, Faridabad, India
{neelam_duhan,ashokkale}@rediffmail.com

Abstract. Spatial data clustering groups similar objects based on their distance, connectivity, or their relative density in space whereas in the real world, there exist many physical constraints e.g. highways, rivers, hills etc. that may affect the result of clustering. Therefore, these obstacles when taken into consideration render the cluster analysis a hopelessly slow exercise. In this paper, a clustering method is being proposed that considers the presence of physical obstacles and uses obstacle modeling as a preprocessing step. With a view to prune the search space and reduce the complexity at search levels, the work further incorporates the hierarchical structure into the existing clustering structure. The clustering algorithm can detect clusters of arbitrary shapes and sizes and is insensitive to noise and input order.

Keywords: Spatial Databases, Data Mining, Cluster Analysis, Density based Clustering.

1 Introduction

Data mining [1] is a powerful tool that can extract hidden predictive information from large databases. *Clustering* [1], is a primary method for data mining that can divide data objects into groups such that based on some criterion, the objects in the same group are similar to each other, while objects in different groups are dissimilar. Cluster analysis has various applications in GIS, image processing, market analysis, mine field detection, satellite imaginary, medical imaginary and chemical analysis etc.

Many effective and scalable clustering methods have been developed in last few years and can be categorized into partitioning methods [2], hierarchical methods [3], density based methods [4], [5] and [6], grid based methods [7], graph theoretic [8], fuzzy methods [9], probabilistic techniques [10] and model based methods [11]. It may be noted that only some of these methods take into account physical obstacles, which can otherwise significantly affect the result of clustering. In spatial databases [12], for instance two points may be close enough with respect to a distance measure but still be restrained from forming a cluster owing to the existence of a physical obstacle such as river or highway between them.

Consider the dataset and physical obstacles shown in Fig 1. Depending upon whether obstacles are ignored or considered, the result of cluster formation will be

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 212–228, 2011.

different i.e. three and six clusters are formed respectively when obstacles are ignored and when they are taken into consideration. The subsequent section presents an overview of existing work and identifies the various limitations found while clustering data objects in the presence of physical obstacles.

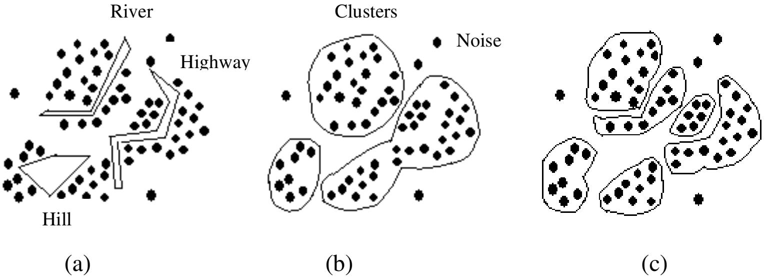


Fig. 1. Clustering data objects in the presence of obstacles. (a) Original Dataset (b) Clusters when ignoring obstacles (c) Clusters when considering obstacles.

2 Literature Review

According to the available literature, these clustering algorithms [6], [13], [14], [15] and [16] have been proposed recently: COD-CLARANS [13] based on partitioning approach, AUTOCLUST+ [14] based on graph partitioning approach, DBCluC [6] based on density based approach and ACAOC [15] based on the basic ant model, which can cluster spatial data in the presence of obstacles.

COD-CLARANS employs an efficient k-medoids method for clustering. It builds BSP tree and visibility graphs to find visibility between two nodes. The visibility graph is expensive to build and considered as preprocessed to reduce the complexity of main algorithm. This algorithm calculates two distance measures: direct Euclidean distance and obstructed distance [13] to find the distance between two points while ignoring the obstacles and considering the obstacles respectively. Here the problem of *Clustering with Obstacle Distance* (COD) is to partition a database D into k clusters C_1, C_2...C_k, such that the following squared error function (1) is minimized:

$$E = \sum_{i=1}^{k} \sum_{p \in C_i} (d'(p, c_i))^2 \tag{1}$$

where c_i is the centre of cluster C_i that is determined by the clustering. If the space between any two obstacles is considered as a corridor, then the cluster centers discovered by the COD-CLARANS are placed at the entrance of the corridors. Such centers are easily accessible by points from other corridors.

AUTOCLUST+ clusters data objects in the presence of obstacles based on Voronoi Modeling and Delaunay Diagrams [14]. It can detect boundaries of clusters having different densities in the dataset. The algorithm is free of user-supplied parameters and detects high-quality clusters i.e. the clusters of arbitrary shapes, clusters of different densities, sparse clusters adjacent to high-density clusters, multiple bridges between clusters and closely located high-density clusters. Consequently, it successfully supports more general locational optimization problems in the presence of obstacles. All this can be done within *(n log n + (m+R) log n)* expected time, where *n* is the number of data points, *m* is the total number of line-segments constituting the obstacles and *R* is the number of Delaunay edges intersecting some obstacles.

ACAOC algorithm can give attention to local converging and the whole converging. Because of the use of Approximate Nearest Neighbor (ANN), the computing speed is increased greatly.

DBCluC (*Density Based Clustering with Constraints*) uses density based clustering and hence the DBSCAN [4] algorithm as primary algorithm to cluster data objects in the presence of obstacles. Obstacle modeling [17] is adopted by the algorithm to speed up the clustering process. This algorithm is better in terms of efficiency and effectiveness than COD-CLARANS.

All the three algorithms can detect clusters of arbitrary shapes and can clearly distinguish between clusters and noise. But a critical look at the existing mechanisms highlights the following limitations:

1. Visibility graphs and delaunay structures are difficult and expensive to build, thus make clustering process also expensive.
2. Performance of COD-CLARANS degrades on large values of k.
3. DBCluC considers the visibility of objects in its cluster formation, which can affect the clustering results. The concept of visibility can be made more efficient to better detect accurate clusters.
4. Multiple-level cluster analysis is not considered by existing mechanisms. For example, if it is desired to have 100 or more points in a cluster then clusters having size smaller than 100 can be merged to produce a large cluster.
5. Obstacle reduction adopted by DBCluC requires a complex preprocessing to be done before clustering.

With a view to handle the above-mentioned drawbacks, an algorithm called "*DBCCOM*: *D*ensity *B*ased *C*lustering with *C*onstraints and *O*bstacle *M*odeling" has been proposed. It is an effective density based clustering algorithm wherein all the obstacles are assumed to be represented using simple polygons like in DBCluC. A novel polygon edge reduction algorithm is also proposed that is easy to formulate and reduces the number of edges required to test during clustering while preserving visibility between points in space, thereby reducing the search space. Further hierarchical version of the algorithm has been designed for analyzing the clusters at various levels.

The rest of the paper is organized as follows: In subsection 2.1 and 2.2, concepts pertaining to density based clustering have been discussed. Subsection 2.3 shows how to model physical obstacles using polygons. Section 3 presents the proposed work wherein subsection 3.1 gives a novel approach for polygon reduction, subsection 3.2 discusses the proposed DBCCOM clustering algorithm and in subsection 3.3, it is shown how to apply an efficient hierarchical clustering to DBCCOM. Section 4 gives the performance of proposed algorithms in terms of time and space complexity. Finally section 5 concludes the paper.

2.1 Density Based Clustering

The key idea behind the density based clustering [4] is that for each point of a cluster, the neighborhood of a given radius (say *Eps*) has to contain at least a minimum number of points (say *Minpts*) i.e. the density of the neighborhood has to exceed some threshold. Following are some definitions to formalize the notion of cluster and noise in the density based clustering.

Definition 1. Eps-neighborhood of a point
The Eps-neighborhood of a point p denoted by $N_{Eps}(p)$, is defined as:

$$N_{Eps}(p)=\{q \in D \mid dist(p,q) \le Eps\} \tag{2}$$

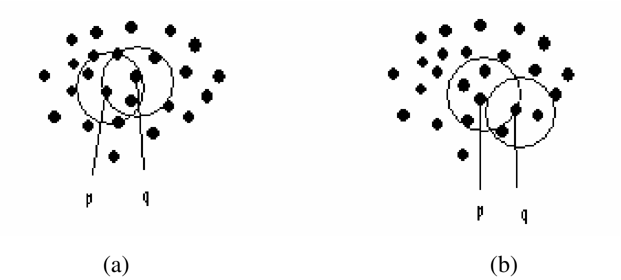
where D is a database of points or objects. Density based clustering require that for each point of a cluster, there should be at least a minimum number of points *Minpts* in the Eps-neighborhood of that point.

Definition 2. Directly-density-reachable
A point *q* is directly-density-reachable from a point *p* wrt *Eps* and *Minpts* if

1. $q \in N_{Eps}(p)$ and
2. $| N_{Eps}(p)| \ge$ Minpts

The data points can be divided core points and border points, where core points satisfy density criterion and exist in the core of the dataset, while border points don't satisfy density criterion and exist on the borders of the dataset. Eps-neighborhood of a border point contains significantly less number of points than that of a core point.

Directly-density-reachability is symmetric for a pair of core points otherwise it is asymmetric as shown in Fig. 2, where a sample dataset is taken. In Fig. 2(a), for instance, *p* and *q* are core points. It may be observed that both are directly-density-reachable from each other, so they form a symmetric pair. Considering Fig. 2(b), where *p* is a core point and *q* is a border point, here *q* is directly-density-reachable from *p* but *p* is not from *q*, so this particular case shows asymmetry between *p* and *q*.



(a) (b)

Fig. 2. Directly-density-reachability with *Minpts*=4 (a) Symmetric pair (p, q) (b) Asymmetric pair (p, q)

Definition 3. Density-reachable
A point *p* is density-reachable from a point *q* wrt *Eps* and *Minpts* if there exists a chain of points $p_1, p_2...p_n$ and $p_1=q$, $p_n=p$, such that p_{i+1} is directly-density-reachable from p_i.

It may be noted that density-reachability is also symmetric for a pair of core points as shown in Fig 3(a).

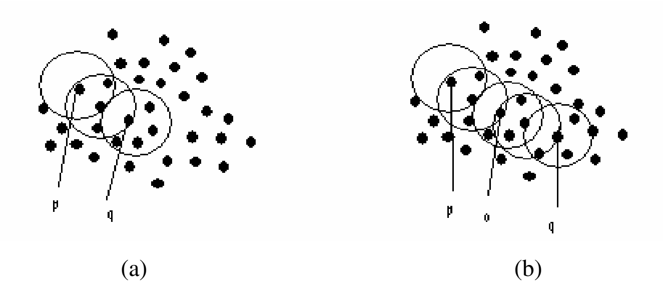


Fig. 3. Density-reachability and density-connectivity with *Minpts*=4 (a) p density-reachable from q but q not from p (b) p and q density-connected to each other by point o

Definition 4. Density-connected
A point *p* is density-connected to a point *q* wrt *Eps* and *Minpts* if there exists a point *o* such that both *p* and *q* are density-reachable from *o* wrt *Eps* and *Minpts* (see Fig. 3(b)). In fact, density-connectivity is also a symmetric relation.

Definition 5. Cluster
Let *D* be a database of points. A cluster *C* wrt *Eps* and *Minpts* is a non-empty subset of *D* satisfying the following conditions:

1. Maximality: $\forall$p,q$\in$D, if p$\in$C and *q* is density-reachable from *p* wrt *Eps* and *Minpts*, then q$\in$C.
2. Connectivity: $\forall$p,q$\in$C, *p* and *q* are density-connected to each other wrt *Eps* and *Minpts*.

Definition 6. Noise
Let C_1, C_2...C_k be the clusters created from the database *D* wrt *Eps* and *Minpts*, then noise is the set of points in the database not belonging to any cluster i.e.

$$\text{noise} = \{p\in D \mid \forall i: p\notin Ci, 1\leq i\leq k\} \qquad (3)$$

2.2 The DBSCAN Algorithm

DBSCAN is a density based clustering algorithm for clustering large spatial databases containing noise. It finds clusters in the database by starting with an arbitrary point (say *p*) and retrieving all points which are density-reachable from *p* wrt *Eps* and *Minpts*. All these points are assigned the same clusterId. These points are further explored to retrieve their density reachable points and the process continues. After discovering that no more points can be added to this cluster, it starts from the next point in the database which is not assigned any clusterId yet. The procedure is repeated until no point in the database remains to be checked.

2.3 Obstacle Modeling Using Polygons

The polygons can be divided into two types: simple polygons and crossing polygons [18]. A simple polygon is the polygon in which any edge of the polygon is not intersected with any other edge of the polygon and in a crossing polygon, at least one edge is intersected with any other edge of the polygon [17]. The proposed work considers simple polygons because almost all obstacle shapes can be represented using them. Simple polygons can further be divided into two types: convex and concave [6]. A polygon is a convex if all vertices make same directional turn either clockwise or anticlockwise, while all other polygons are said to be concave (Fig. 4).

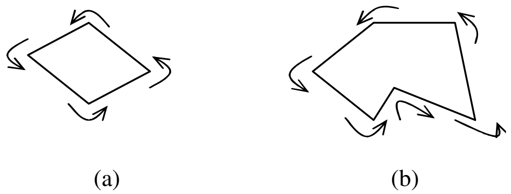


Fig. 4. Types of polygons on basis of direction (a) Convex Polygon (b) Concave Polygon (with one concave vertex)

Definition 7. Polygon
A simple polygon is denoted by an undirected graph P(V,E) where *V* is a set of *k* vertices: V = {v_1, v_2…v_k} and *E* is a set of *k* edges: E = {e_1, e_2…e_k} where e_i is a line segment joining v_i and v_{i+1}, 1≤i≤k,. i+1=1 if i+1>k.

Definition 8. Obstacle
An obstacle can be thought of a simple polygon P(V,E) where *V* is the set of vertices and *E* is the set of edges of the polygon.

Definition 9. Visibility

1. **If polygon reduction is not used**: Let D={d_1, d_2…d_n} is the dataset of *n* points. Visibility is a relation between two data points. Two points d_i and d_j, i≠j (1≤i,j≤n) are visible to each other if the line segment joining d_i and d_j is not intersected with any polygon edge e_i; 1≤ i≤ k.
2. **If polygon reduction is used**: Let L= {l_1, l_2…l_m} be a set of m reduction lines produced by a polygon reduction algorithm. Two points d_i and d_j , i ≠ j (1≤i,j≤n) are visible to each other if the line segment joining d_i and d_j is not intersected with any *reduction line* l_i; 1≤ i≤ m.

According to the above definition of visibility, the number of line segments to check is the total number of edges of the polygons if polygon reduction is not used, which for a large data space is large in number. This number can be reduced to nearly half by the proposed polygon-edge reduction algorithm. Let us call the reduced number of lines as *reduction lines*. When polygon reduction is applied, the second definition of visibility applies to the main clustering algorithm.

3 The Proposed Clustering Method

A density based clustering algorithm *DBCCOM* (*D*ensity *B*ased *C*lustering with *C*onstraints and *O*bstacle *M*odeling) is proposed here, that can cluster data objects in the presence of obstacles. DBCCOM makes the assumption that all physical obstacles are represented by polygons and tries to reduce the complexity of cluster formation by proposed polygon reduction method. DBCCOM algorithm operates in three phases:

- First, compress/reduce the obstacles i.e. polygons.
- Second, perform cluster formation.
- Third, apply hierarchical clustering on resultant clusters.

The first phase constitutes a preprocessing phase, while second and third phases contribute toward the main algorithm. The three phases of the algorithm are discussed in the following subsections.

3.1 The Polygon Reduction Algorithm

An efficient polygon reduction method is proposed in this section that can delimit the number of edges of a polygon to be checked during visibility (Def. 9) by reducing their cardinality. While clustering data objects, in addition to normal constraints, visibility between various data objects is checked and it is assumed that two points visible with each other can be grouped in the same cluster.

The algorithm is given below:

```
Algorithm: polygon_reduction(P)
Input: A polygon P(V,E) stored in A[n×n]
Output: A set of reduction lines in output matrix O[n×n]
// Start of the Algorithm
 Identify convex & concave vertices; //Let m= no. of convex vertices
 If (polygon is Convex)                     //Here, m= n
   { O[m×m]=0;                           // initialize the matrix O
     For ( i= 1 to $\lceil n/2 \rceil$  vertices)      //consider in an order
       {  Take $i^{th}$  vertex where Flag[i]=0;
          Join it with $j^{th}$= ( $\lfloor n/2 \rfloor$ + i)$^{th}$ vertex;
           //line lies interior or coincides with any edge of polygon
          O[i,j]=1;
       } //end for
    Return output matrix O[m×m]                // i.e. Reduction Lines
  } // end if
else                               //polygon is concave
  { //  Treating m  convex vertices
    O[n×n]=0;
    Flag[p]=0;     // 1≤ p ≤ n, all vertices are unvisited initially
    For (all unvisited convex vertices)
     { Take one vertex (say k) which is convex & Flag[k]=0;
       Set Flag[k]=1;
       For (all other vertices (j, where j is convex and Flag[j]=0))
       {  If (edge <s,r> exists i.e. A[s,r]=1)
           {   O[s,r]=1 and Flag[r]=1; }
          else
            {  If (line joining (s,r) lies interior to polygon)
                  {  O[s,r]=1 and Flag[r]=1; }
               else
                   place vertex s in the list of unchecked vertices;
            }
```

```
        } //end for
     } //end for
   For (all unchecked convex vertices)
    {  find the adjacent concave vertex t;
       find the adjacent convex vertices of t (say i & j);
       O[t,i]=1 and O[t,j]=1 and Flag[t]=1;
    }
   // Treating (n-m) concave vertices
   For (all unvisited concave vertices)
    { If ( count (unvisited vertices) > 1)   //more than one concave
       {  pick any vertex t where Flag[t]=0;
          For (remaining concave vertices t' where Flag [t']=0)
           {
            If (line (t, t') is not intersected by any line of O[n×n])
             {
               find the adjacent convex vertices of  t  (say i & j);
               O[t,i]=1 and O[t,j]=1 and Flag[t]=1;
             }
           } // end inner for
       } // end if
      else
         break;
    } // end outer for
   Return output matrix O[n×n]             // i.e. Reduction Lines
} end else
```

In this algorithm, first, the convex and concave vertices of the polygon are segregated by geometric application [18] because convex vertices serve the major purpose of finding the visibility between two data objects. Assume that in a total of *n* polygon vertices (|V|=n), *m* are the convex vertices, m≤n. Assume *n* vertices of the polygon P(V, E) be stored in an adjacency upper half matrix *A* of order *[n×n],* where:

$$A[i,\, j] = \begin{cases} 1 \; if \;\; edge \;\; (i,\, j) \; exists \;\; between \quad vertices \quad i \; and \quad j \\ 0 \; if \;\; edge \;\; (i,\, j) \notin E \end{cases}$$

The entries in the upper half of *A* are checked so as to avoid the repetition because the obstacle/polygon is an undirected graph. The algorithm returns the output upper half matrix *O[n×n]*, where:

$$O[i,\, j] = \begin{cases} 1 \; if \; reduction \;\; line \; exists \; between \; vertices \; i \; and \;\; j \\ 0 \; otherwise \end{cases}$$

The algorithm makes the following assumptions:

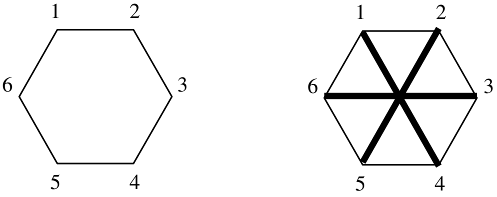
1. Diagonal entries in the output matrix are zero.
2. All reduction lines should be interior to polygon P.
3. Each convex vertex should have at least one reduction line from it, while such constraint does not apply to concave vertices. Concave vertex may or may not have any reduction line emerging from/to it.

The algorithm operates as follows: *First*, it identifies whether the polygon is convex or concave. *Second*, if polygon is convex i.e. all vertices are convex; it visits all vertices in a particular order. The $\lfloor n/2 \rfloor$reduction lines are obtained for *n* vertices and process of obtaining reduction lines repeats $\lceil n/2 \rceil$ times. Each time a vertex *i* is joined with $(\lfloor n/2 \rfloor + i)^{th}$ vertex resulting in a reduction line. The reduction lines obtained in this manner always lie inside the polygon or on any edge of the polygon, thus are valid reduction lines. The corresponding reduction lines are placed in the output matrix *O*. *Third*, on the other hand, if polygon is concave, then both convex and concave vertices are treated separately. For *convex vertices*, same procedure is followed as before but each time the algorithm should pick the convex vertex, which is surrounded by other convex vertices and lies approximately in the middle of them, so that most of the convex vertices get visited at one time. The polygon edges can also be treated as reduction lines. Place the corresponding reduction lines in *O.* Repeat the procedure until possible number of convex vertices gets visited, but there may remain some convex vertices not satisfying the third assumption of the algorithm. In this case, for each such convex vertex, the adjacent concave is found and algorithm joins that concave with its adjacent convex vertices, resulting in two reduction lines. For *concave vertices,* if there is a single concave vertex in the polygon, then do not consider it. Otherwise, examine every pair of concave vertices by checking whether the line joining them intersects with any previously found reduction line. If this is so, then leave them i.e. no need to draw more reduction lines, otherwise join any one of them with their adjacent convex vertices to return two reduction lines. *Fourth*, output matrix *O* is returned depending on the polygon type.

The algorithm always follows the assumption that, there should be at least one reduction line from each convex vertex in the resulting reduced polygon, but this is not true for concave vertices. A reduction line is admissible if it is either inside the polygon or coincides with any edge of the polygon.

3.1.1 Example Showing Polygon Reduction

Take the example convex polygon shown in Fig. 5(a) that has six convex vertices and six edges. Corresponding to these six convex vertices, the input matrix *A* is shown. As a result of the application of the polygon_reduction() algorithm, the output matrix *O* is shown in Fig. 5(b). The output-reduced polygon is constructed according to output matrix *O,* it is no more closed and is shown in Fig. 5(b), which contains four reduction lines instead of six polygon lines.



$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 1 \\ & 0 & 1 & 0 & 0 & 0 \\ & & 0 & 1 & 0 & 0 \\ & & & 0 & 1 & 0 \\ & & & & 0 & 1 \\ & & & & & 0 \end{bmatrix} \qquad \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ & 0 & 0 & 0 & 1 & 0 \\ & & 0 & 0 & 0 & 1 \\ & & & 0 & 0 & 0 \\ & & & & 0 & 0 \\ & & & & & 0 \end{bmatrix}$$

(a) (b)

Fig. 5. Example representing polygon reduction. (a) Input convex polygon (b) Output reduced polygon represented by dark edges.

This is the case where significant improvement is not achieved but in the case of concave polygons, a remarkable improvement can be obtained as is shown in Fig. 6, where *ten* edges are being reduced to only *three* reduction lines (see Fig. 6(a)).

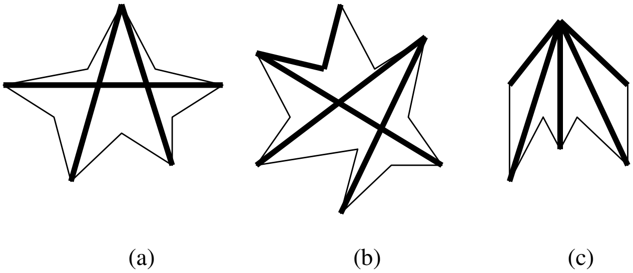


Fig. 6. Concave Polygons and Reduction Lines (a) 3/10 (b) 5/12 (c) 5/8

On an average, the number of reduction lines obtained by the polygon reduction algorithm is approximately half the number of edges in polygon. So, in large datasets, where the number of obstacles can be large in number, polygon_reduction() algorithm can be applied to reduce the number of lines to test during the clustering procedure.

3.1.2 Performance Evaluation

Complexity of polygon_reduction() is much less than $O(n^2)$ (*n* is the total number of vertices in the polygon). The reduction algorithm can be used as a preprocessing step for the clustering algorithm. Although the complexity of the reduction algorithm costs a little bit, but benefits the clustering process. The visibility between every pair of points can be tested easily and at a faster rate, reducing the overall running time of the clustering procedure very much. Let *n* be the number of points of a polygon P, and *p* and *m* are the number of concave points and the number of convex points respectively with $n=p+m$. The convexity test for P requires $O(n)$ [6]. The complexity of the polygon reduction is of the order of $O(n^2)$ in the worst case.

3.2 The Clustering Algorithm DBCCOM

The clustering algorithm *DBCCOM* (pronounced *D-B-COM*) takes into account the problem of clustering in the presence of physical obstacles (constraints) while modeling the obstacles by polygon reduction algorithm. Some important definitions related to the algorithm DBCCOM, besides those in density based clustering, are further added to have a clear idea about the algorithm. The input parameters *Eps* and *Minpts* are the same as are in DBSCAN presented in the literature. In the presence of obstacles, the new definition of cluster becomes:

Definition 10. Cluster

Let *D* be a database of points. A cluster *C* wrt *Eps* and *Minpts* is a non-empty subset of D satisfying the following conditions:

1. Maximality: $\forall$ p, q$\in$D, if $p\in C$ and *q* is density reachable from *p* wrt *Eps* and *Minpts*, then $q \in C$.

2. Connectivity: $\forall$ p, q$\in$C, *p* and *q* are density connected to each other wrt *Eps* and *Minpts*.
3. Visibility: $\forall$ $c_{i,}$ $c_j$$\in$C, there exists a chain of zero or *k* points: c_1, c_2...c_k, k$\neq$i, j and k$\leq$n such that both c_i and c_j are visible to each other either directly or indirectly through the pair wise chain of *k* points. For every $c_i$$\in$C there exists at least one point $c_j$$\in$C, such that c_i and c_j are visible to each other.

The concept of visibility is shown in Fig. 7 when considering obstacles and their reduction lines. According to old definition (Def. 5), points p and q (Fig. 7(b)) can not form a cluster together, while they can be in the same cluster according to the new definition (Def. 10).

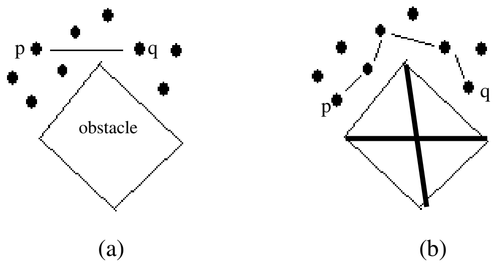


Fig. 7. Visibility between points of a cluster. (a) Direct visibility between two points (b) Indirect visibility between two points.

The clustering algorithm *DBCCOM* is given below where polygon_reduction() is used as a preprocessing step. The clustering procedure in *DBCCOM* is quite similar to *DBSCAN*. In the algorithm, database *D* is a set of data points to be clustered. First, all the reduction lines from the set of obstacles are extracted by an iterative call to polygon_reduction(). After that clustering procedure is initiated.

```
Algorithm: DBCCOM (D, O)
Input: database of n points, a set of obstacles, Eps, Minpts
Output: A set of Clusters C.
//Start Obstacle Modeling
   L=ϕ; C=ϕ;                      // L is the set of reduction lines
   For (obstacle ∈ O)
       {  Lines=Polygon_reduction (obstacle);
          L= L ∪ lines;
       }
//Start Clustering
   ClusterId=nextId (noise);       //nextId() assigns a new ClusterId
   C= { ClusterId };
   For (pt ∈ D)
       {  If (ClusterId(pt)=unclassified)
            {   If ( ExpandCluster(D, pt, ClusterId, Eps, Minpts, L))
                     {   ClusterId= nextId(ClusterId);
                         C=C∪ ClusterId;
                     }
            } //end if
       } //end for
 Return C
```

For every point of the database which is not assigned any *ClusterID* yet, a new ID is assigned and the *Eps-neighborhood* of the point is retrieved by making a call to the subroutine *ExpandCluster().* The subroutine extracts the neighbors by considering obstacles and iteratively retrieves neighbors of previous neighbors till density constraints are fulfilled. After discovering that no more points can be added to the current cluster, it returns true. The *DBCCOM* then starts with a new point and a new *ClusterId.* It iteratively performs the same process until all points are assigned any *ClusterID.* The *ExpandCluster()* [6] is given below and is approximately similar to the subroutine in DBSCAN. However, the distinction is that it considers the reduction lines in its function *RetrieveNeighbours()* illustrated afterwards. Accepted neighbors placed in *SEED* list that are retrieved by *RetrieveNeighbours()* continue to expand a cluster, if number of elements in the *SEED* is greater than or equal to *Minpts*.

```
Algorithm: ExpandCluster (D, pt, ClusterId, Eps, Minpts, L)
Input:Database, data point, ClusterId, Eps, Minpts, Reduction lines L
Output: True or False
//Start Algorithm
 SEED = RetrieveNeighbours (pt, Eps, L);
 If (SEED.SIZE< Minpts)
    {  Classify pt as NOISE;
       Return False;
    }
 Change cluster Id of all elements in SEED into ClusterId;
 Delete pt from SEED;
 While (SEED.SIZE>0)
   {  Current-Point = SEED. First ();
      RESULT = RetrieveNeighbours (Current-Point, Eps, L);
      If (RESULT.SIZE>=Minpts)
        { For (element ∈ RESULT)
           { If (ClusterId (element)= unclassified )
                {   Put it into SEED;
                    Set its cluster id to ClusterId;
                }
             If (element is NOISE) //Noise is now assigned to cluster
                {    Set its cluster id to ClusterId;   }
           } //end for
        } //end if
        Delete Current-Point from SEED;
   } //end while
 Return True;
```

The *RetrieveNeighbours()* function is illustrated below, it retrieves the neighbors of a point and by the way of *Check-Visibility()* function, checks the visibility (Def. 5 & 10) of point with its neighbors.

```
Algorithm: RetrieveNeighbours (Point, Eps, L)
Input: a data object, Eps, Reduction lines L
Output: A set of neighbor data points
//Start Algorithm
 RESULT = GetNeighbour(Point, Eps);    //Extract Eps-neighborhood
 For (object ∈ RESULT)
    {   If (Check-Visibility (Point, object, L)=FALSE )
          {   RESULT.Delete (object);   }
    }
 Return RESULT;
```

Given a query point, neighbors of the query point can be retrieved using R*-tree or SR tree [19,20]. The average run time of a neighbor query is *O(log N)* where *N* is the number of data objects. The visibility between two data objects in the presence of obstacles is examined using a line segment joining the two objects. If this line is intersected with any reduction line, then the two data points are not grouped together, since they are not visible to each other. A data object is labeled by a proper clusterId, if retrieved neighbors are satisfied with the parameter *Eps* and *Minpts*. The *RESULT* in the above algorithm is a set of neighbor points of a given point in question. The final *RESULT* elements are constructed by removing data objects that are not visible to the sample point because of the presence of obstacles between them.

3.2.1 Performance Evaluation

Polygon Reduction algorithm is a pre-processing phase that precedes the clustering process. The complexity of the clustering algorithm alone, is in the order of *O((NlogN)L)*, where *L* is the number of reduction lines generated by the polygon reduction algorithm, and *N* is the number of points in the database. To check the visibility between two data points, all reduction lines are tested. The number of reduction lines to be checked during visibility can further be reduced by evaluating only lines in the neighborhood of the points [6], which can easily be found by applying mathematical techniques.

3.3 The Hierarchical DBCCOM

The hierarchical version of DBCCOM takes the concept of hierarchical clustering [1] to produce a hierarchy of clusters. By producing a hierarchy, the search space can be pruned and cluster analysis can be performed at multiple clustering levels. It applies agglomerative approach of hierarchical clustering to a set of clusters produced by DBCCOM algorithm until a specified termination condition is satisfied. To better understand this, here is another definition to find the distance between two clusters.

Definition 12. Distance between two clusters

Let C = {C_1,C_2...C_k} be the set of *k* clusters produced by clustering algorithm. The distance between two clusters C_i and C_j is defined as:

$$dist(C_{i,}\ C_j) = Min\{\ dist(p,q) \mid p \in C_i \text{ and } q \in C_j\} \tag{4}$$

This distance function takes distance between two clusters as the distance between two nearest points from two clusters. The other method to define this measure is:

$$dist(C_{i,}\ C_j)= dist(cc_i,\ cc_j) \mid cci,\ ccj \text{ are centres of } Ci\ \&\ Cj \tag{5}$$

This measure states that the distance between two clusters is the distance between their centres. Centres of the clusters can be easily calculated by finding the mean of all points in a cluster. The measure (5) is adopted here due to less time complexity.

The algorithm for hierarchical version of DBCCOM is given below. The threshold distance D_{min} is taken to have an upper bound on the acceptable distance between two clusters. Two clusters can be merged at a subsequent step if the distance between them is no more than D_{min}. The clusters at subsequent steps are merged together until the current clustering stage becomes identical to the previous stage i.e. the number of clusters is same at two stages; this forms the termination condition for the *hierarchical_DBCOMM()*.

```
Algorithm: DBCCOM_Heirarchical (D, O, Dmin)
Input:Database of points,set of obstacles,threshold distance $D_{min}$>Eps
Output: A hierarchy of Clusters CH={CS1, CS2,…CSm}.
       //CSi is the set of clusters at level i, where CSi= {$C_1$,$C_2$,…$C_k$}
//Start Clustering
 C = DBCCOM(D,O);                // C is set of original clusters
 $CS_1$= C;                          // Start Hierarchical Clustering
 CH= {$CS_1$};                       // The first level Cluster set
 i= 1; $CS_{i+1}$=ϕ ;
 Label: Initialize all Ck∈ $CS_i$ as unvisited;   //Ck.Visited= False
 For (all unvisited Ck ∈ $CS_i$)
   { For (all other unvisited Cj∈ $CS_i$, j≠ k)
      { If ( dist (Ck, Cj)≤  Dmin )
         {  Merged_Pts= merge (Ck, Cj);  //merge pts of both clusters
            ClusterId= next_Id (ClusterId);    //take new clusterId
            Assign ClusterId to Merged_Pts;
            Ck.Visited= Cj.Visited= True;
            $CS_{i+1}$=  $CS_{i+1}$∪ {ClusterId}; //$CS_{i+1}$ is next level clusterset
         }
      } //end inner for
   } // end outer for
 For (all remaining unvisited Ck ∈ $CS_i$ )
   { $CS_{i+1}$=  $CS_{i+1}$∪{Ck}; //place unvisited in $CS_{i+1}$ after being checked
   }
 If ($CS_i$== $CS_{i+1}$)                   // Algorithm termination condition
   {  Return CH;  }  //output multiple cluster sets CSi, stored in CH
 else
   {  CH= CH ∪ $CS_{i+1}$;
      i=i+1;
      $CS_{i+1}$=ϕ ;
      GoTo Label;
   }
```

Fig. 8 shows one example illustrating this idea. In this figure C= {a, b, c, d, e, f, g, h} is the original clustering produced by the clustering algorithm *DBCCOM()* wrt *Eps, Minpts* and is at stage 1. Further merging of clusters result in stage 2 where C becomes: C= {i, c, j, k, h}. In this way process goes up to stage 4, which is similar to stage 3, where process of hierarchical clustering stops.

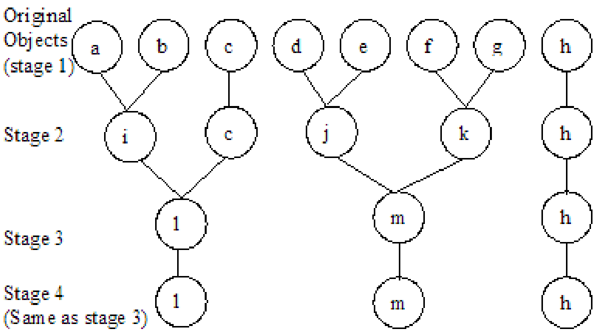


Fig. 8. Example Hierarchical Clustering

4 Experimental Results

In this section, the experimental evaluation of the polygon reduction and the clustering algorithm *DBCCOM* has been presented. The performance of polygon reduction in terms of scalability is evaluated. For experimental purposes, some synthetic databases and hypothetical obstacles with arbitrary shapes (and sizes) have been considered. The polygon reduction was applied to varying size polygons (both convex and concave) and was found to scale well. Fig. 9 shows the performance of the algorithm in terms of reduction achieved (reduction shown for concave as well as convex polygons) e.g. 13 concave and convex polygons with 151 edges overall achieved a reduction of 66 lines as shown in figure.

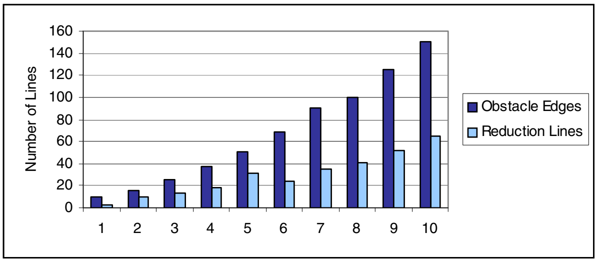


Fig. 9. Reduction in Concave/Convex polygons

Some self assumed databases with different sizes and reduction lines were input to the clustering algorithm. In Fig. 10, performance of the algorithm is shown while considering obstacle edges and reduction lines. Ten datasets are taken for experimental work varying in sizes from 500 to 5000 points with a regular increment of 500 data points. Same number of edges and reduction lines (here 45/100) are considered for each database of points. The time to compress the polygons is not considered in the clustering time.

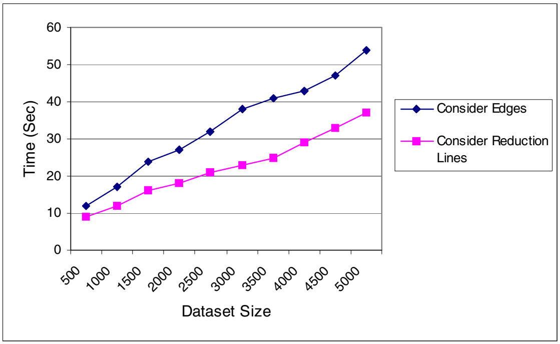


Fig. 10. Time Complexity with respect to varying size Databases

The figure shows that the algorithm performs well in the presence of reduction lines. The DBCCOM, when compared with the existing algorithm DBCluC describes that proposed polygon reduction achieves approximately same (and superior in some cases) reduction as proposed by DBCluC, but does the same without applying the complex shortest path algorithms for concave vertices. Moreover, it performs multiple-level cluster analysis thus reducing the search space and enhancing the efficiency of analysis.

5 Conclusion and Future Work

Clustering is a good practice to help analysts focus on the most desired grouping in the dataset. In the presence of obstacles, it becomes difficult to find these groupings. In this paper, the problem of clustering in the presence of obstacles has been considered. Other factors such as multi-level cluster analysis are also taken into account. A polygon reduction method has been proposed, which models the physical obstacles for reducing the number of line segments to check during the clustering procedure and applies this method in the cluster formation process. The concept of visibility in clustering has been improved that tries to capture as many points in a cluster as possible. The procedures called by the main clustering algorithm are nearly same as that of *DBSCAN* and *DBCluC,* but proposed algorithm *DBCCOM* operates faster and can reduce the line segments to a better digit. It can find the clusters of arbitrary shapes and sizes while discriminating noise from clusters. The method is well scalable in terms of number of data objects as well as in terms of number of obstacles. The hierarchical version of DBCCOM can produce a hierarchy of clusters with minimum domain knowledge. By producing a hierarchy of clusters, the cluster analysis at various levels can be performed.

Future research includes: In this paper, no such scheme is used that can further reduce the number of reduction lines to be checked for visibility. By using some nearest neighbor detection schemes, only lines in the neighborhood of the point can be checked and complexity can be reduced. Algorithm can be extended to operate in n-dimensional space. Some method may be devised so as to model the physical obstacles in higher dimensions and thus to cluster data objects in more than 2-dimensions. The obstacles are assumed to be simple polygons while there may be obstacles in the real world, which may take different shapes like circular or elliptical etc. There must be some method to model these types of obstacles.

References

1. Han, J., Kamber, K.M.: Data Mining: Concepts and Techniques. Morgan Kaufmann, San Francisco (2000)
2. Kaufman, L.: Finding groups in data: An introduction to cluster analysis. Wiley, New York (1990)
3. Karypis, G., Han, E., Kumar, V.: Chameleon: A hierarchical clustering algorithm using dynamic modeling. IEEE Computer, 68–75 (1999)

4. Ester, M., Kriegel, H.-P., Sander, J., Xu, X.: A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In: Proceedings of 2nd International Conf. on Knowledge Discovery and Data Mining, pp. 226–231. AAAI Press, Portland (1996)
5. Jain, A.K., Dubes, R.C.: Algorithms for Clustering Data. Prentice-Hall, Inc., Englewood Cliffs (1988)
6. Zaiane, O.R., Lee, C.-H.: Clustering spatial data in the presence of obstacles: A density based approach. In: IDEAS 2002, Edmonton, Canada (2002)
7. Schikuta, E.: Grid clustering: An efficient hierarchical clustering method for very large data sets. In: Proceedings of 13th International Conference on Pattern Recognition, vol. 2, pp. 101–105 (1996)
8. Zahn, C.: Graph-theoretical methods for detecting and describing gestalt clusters. IEEE Transactions on Computers 20, 68–86 (1971)
9. Bezdek, J., Hathaway, R.: Numerical convergence and interpretation of the fuzzy c-shells clustering algorithm. IEEE Transactions on Neural Networks 3(5), 787–793 (1992)
10. Brailovsky, V.: A probabilistic approach to clustering. Pattern Recognition Letters 12(4), 193–198 (1991)
11. Shavlik, J.W., Dietterich, T.G.: Readings in Machine Learning. Morgan Kaufmann, San Francisco (1990)
12. Ng, R.T., Han, J.: Efficient and Effective Clustering Methods for Spatial Data Mining. In: Proceedings of 20th International Conference on Very Large Data Bases, Santiago, Chile, pp. 144–155. Morgan Kaufmann Publishers, San Francisco (1994)
13. Tung, A.K.H., Hou, J., Han, J.: Spatial clustering in the presence of obstacles. In: Proceedings of International Conference on Data Engineering, ICDE 2001 (2001)
14. Estivill-Castro, V., Lee, I.: Autoclust+: Automatic clustering of point-data sets in the presence of obstacles. In: International Workshop on Temporal and Spatial and Spatio-Temporal Data Mining (2000)
15. Qu, J., Liu, X.: A Revised Ant Clustering Algorithm with Obstacle Constraints. In: WRI World Congress on Computer Science & Information Engineering (2009)
16. Zhang, X., Deng, G., Liu, Y., Wang, J.: Spatial Obstructed Distance based on the Combination of Ant Colony Optimization and Particle swarn Optimization. In: Fourth IEEE Conference on Industrial Electronicsand Applications, ICIEA (2009)
17. Lee, C.-H., Zaïane, O.R.: Polygon reduction: An algorithm for minimum line representation for polygons. In: 14th Canadian Conf. on Computational Geometry (2002)
18. Stone, M.: A mnemonic for areas of polygons. Amer. Math. Monthly 93, 479–480 (1986)
19. Beckmann, N., Kriegel, H.-P., Schneider, R., Seeger, B.: The R*-tree: An Efficient and Robust access Method for Points and Rectangles. In: Proceedings of ACM SIGMOD International Conference on Management of Data, Atlantic City, NJ, pp. 322–331. ACM Press, New York (1990)
20. Katayama, N., Satoh, S.: The SR-tree: an index structure for high-dimensional nearest neighbor queries. In: Proceedings of the ACM SIGMOD Intl. Conf., pp. 369–380 (1997)

GOREWEB Framework for Goal Oriented Requirements Engineering of Web Applications

Shailey Chawla[1], Sangeeta Srivastava[2], and Punam Bedi[1]

[1] Department of Computer Science, University of Delhi, North Campus, Delhi - 110007, India
[2] Department of Computer Science, Bhaskaracharya College, Sec-2, Dwarka, Delhi, India
shaileychawla@gmail.com, sangeeta.srivastava@gmail.com,
pbedi@cs.du.ac.in

Abstract. In this paper, we propose a framework for modeling goal driven requirements of web applications. Web engineers mostly focus on design aspects only overlooking the real goals and expectations of the user. Goal oriented Requirement Engineering is a popular approach for Information system development but has not been explored much for Web applications. However, in today's times Web is dominating in every business making it imperative that its requirements are analyzed carefully and in profundity. Goal driven requirements analysis helps in capturing stakeholders' goals very finely, by choosing between alternatives and resolving conflicts. Detailed classification of both functional and non-functional requirements specific to web applications is discussed in the presented work. A framework, GOREWEB (Goal oriented Requirements Engineering for Web Applications) is proposed for analyzing goals and translating them into functional and non-functional web requirements.

Keywords: Goal oriented Requirements Engineering, Web engineering, Goals, Requirements, URN.

1 Introduction

Although web applications have mushroomed a great deal but they have still not received much attention from the requirements engineering community. Like the traditional information systems, where Requirements analysis is given utmost importance amongst all the phases, with web applications the focus is usually more on the presentation. Web applications involve multiple stakeholders, and the size and purpose of the applications are also varied [1]. Gaus et al in [2] defined Requirements Engineering (RE) as the set of activities intended at assuring that a software system fulfills the goals, the needs and the expectations of all the relevant stakeholders. In the requirements engineering community, the requirements have been divided into functional [3] and non-functional requirements[4]. There has been a lot of emphasis on the functionality; however the functionality is not useful or usable without the necessary non-functional characteristics [5]. According to Rolland et al [6] Requirements engineering extends the 'what is done by the system' approach with the 'why is the system like this' view. This 'why' is answered in terms of organizational

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 229–241, 2011.

objectives and their impact on information systems supporting an organization. Poor requirements augment the risk of missing the opportunity of meeting customers' needs and enhancing the user experience [7].

The rest of the paper is organized as follows. In section 2, we describe the related work in the area of Goal oriented Requirements Engineering. Also, we give a brief overview of User Requirements Notation (URN) that we would be using for Goal oriented Requirements Analysis. In section 3, we propose a framework for incorporating goal oriented requirements analysis for engineering web application. For integrating goals with web specific requirements in the framework, we feel that a proper categorization of web specific functional and non-functional requirements needs to be done. We have provided a web specific functional and non-functional requirement categorization in section 4. Further, the framework has been explained using a case study on web based education in section 5. Finally, section 6 summarizes our work and concludes the paper.

2 Goal Oriented Requirements Engineering

In recent times, Goal oriented Requirements Engineering proposed by Mylopoulos [4] has become very popular for analyzing the requirements. A goal describes the objectives that the system should achieve through the cooperation of agents in the software-to-be in a given environment as defined by Liu et al in paper [8]. According to Lamsweerde, Goal-oriented requirements engineering (GORE) is concerned with the use of goals for eliciting, elaborating, structuring, specifying, analyzing, negotiating, documenting, and modifying requirements[9]. The goal based analysis helps to explore the alternatives, resolving conflicts, and relate them to the organizational objectives [4]. It has been also established in [10] that stakeholders pay more attention to goal models compared to the UML models because they can relate to the concepts more closely. There has been a massive amount of work on linking goals and scenarios together [11],[12], [13], [14]. The obvious reason for this linking is that scenarios and goals have complementary characteristics; the former are concrete, narrative, procedural, and leave intended properties implicit; the latter are abstract, declarative, and make intended properties explicit. Scenarios and goals thus complement each other nicely for requirements elicitation and validation. Based on a bidirectional coupling between scenarios and goal, Rolland et al [16] propose heuristic rules for finding out alternative goals covering a scenario, missing companion goals, or sub goals of the goal under consideration.

Many approaches have been developed for Goal oriented Requirements Engineering for generic systems [16],[17], [18]. However, the notations and models developed for generic applications do not address very important issues of web applications like navigation, adaptation etc. Some work has been done by researchers [19], [20], [21],[22] on web engineering approaches taking into account the Goal driven analysis, but many concepts of goal driven analysis like design rationale, conflict resolution, goal prioritization have been surpassed and not taken in totality.

URN [23],[24] refers to User Requirements Notation. It is currently the only standard that combines goals and scenarios in one notation. It is a combination of two notations GRL (Goal Requirements Language) and UCM (Use Case Maps). User Requirements notation aims to capture goals and decision rationale that finally shape a system and model dynamic systems where behavior may change at run time. GRL is *Goal Requirements Language* that focuses on Goal analysis. It help in defining the goals including the non-functional requirements, evaluating them, resolving conflicts etc. UCM stands for *Use Case Maps* that are the visual notation for scenarios. UCM notation employs scenario paths to illustrate causal relationships among responsibilities. The combination of GRL and UCM as depicted in Fig. 1 helps to improve the definition of new goals and satisfy them. GRL as described by Amyot in [23] supports five kinds of intentional elements explained below:

- **Goal:** Quantifiable high-level (functional) requirement (rounded cornered rectangle).
- **Soft goal:** Qualifiable but unquantifiable requirement, essentially non-functional (irregular curvilinear shape).
- **Task:** Operationalized solution that achieves a goal or that *satisfices* a soft goal which can never be fully achieved due to its fuzzy nature hexagon).
- **Resource:** Entity whose importance is described in terms of its availability (rectangle).
- **Belief:** Rationale or argumentation associated to a contribution or a relation (ellipse).

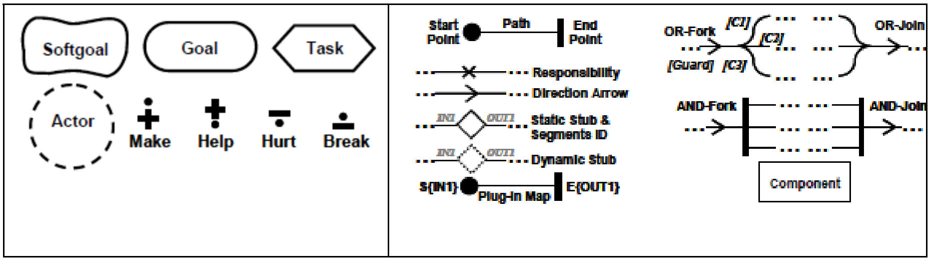


Fig. 1. Subset of GRL &UCM notation

There are also five categories of intentional relations, which connect elements:

- **Contribution:** Describes how soft goals, tasks, beliefs, and relations contribute to each other. Each contribution can be qualified by a degree: equal, break, hurt, some-, undetermined, some+, help, or make.
- **Correlation:** Contribution that indicates side-effects on other intentional elements (dashed line).
- **Means-end:** Link for tasks achieving goals. Different alternatives are allowed.
- **Decomposition:** Defines what is needed for a task to be performed (refinement), always AND.
- **Dependency:** Link between two actors depending on each other (half-circle).

UCMs have following basic concepts according to Amyot [23].

- **Start point:** Captures preconditions and triggering events (filled circle).
- **Responsibilities:** locations where computation (procedure, activity, function, etc.) is necessary (cross).
- **End point:** Represents resulting events and post-conditions (bar).

- **Paths:** Connects start points to end points and can link responsibilities in a causal way.
- **Component** represents an abstract entity (object, server, database etc.)(rectangle).

There has been work on Goal oriented Requirement analysis like i*, NFR framework, URN [17], [5], and [24] but these are for generic systems. The specific needs of web applications like heterogenous user group, specific emphasis on navigation and presentation need a special focus. The framework described in the next section overcomes the gaps in web engineering approaches.

3 GOREWEB Framework

For enhancing the requirements engineering activities involved in web application development, GOREWEB: **G**oal **O**riented **R**equirements **E**ngineering for **Web** applications framework offers goal oriented requirement analysis of web applications. GOREWEB model extends the concepts of User Requirements Notation (URN) for comprehensive study of web application requirements. Amongst numerous differences, the concepts of navigation, adaptation, presentation are distinctive to web applications. Bolchini has provided a metamodel for integrating goals with web engineering approach in [19] where he uses i* for goals modelling and maps the goals to web requirements and later to a web design approach WebML. However, we have enhanced the metamodel in our framework by taking URN as the backbone of Goal analysis that couples functional and soft goals with the scenario modelling. Also, as described in the next section web application requirements have been enhanced and redefined. The framework is shown in Fig. 2. The framework created using the standard UML class diagram depicts the relation between the raw goals captured from the stakeholders and the requirements to be used by the web designers. For realization of web specific requirements, we first need to categorize web application requirements, so that we know how goals are mapped to specific class of requirements. In the next section we categorize web application requirements so that they can be appropriately mapped with the sub goals and tasks in the GOREWEB framework.

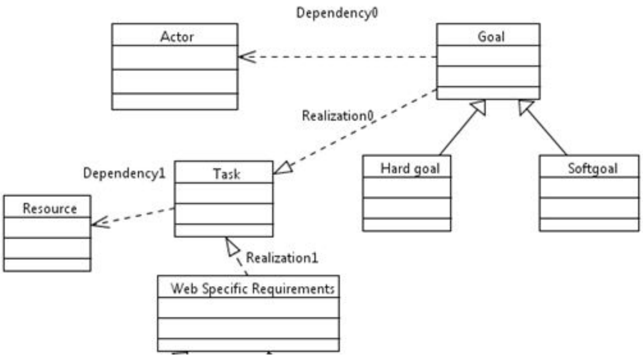


Fig. 2. GOREWEB Framework

4 Web Application Requirements

As with generic systems, web applications requirements are classified as functional and non-functional requirements. Although many other researchers classified the web requirements (mostly functional), a summarized unanimity is presented in [25]. However, we found that the focus is still more on functional requirements. We hereby extend and redefine web application requirements. The categorization is depicted in Fig. 3 below.

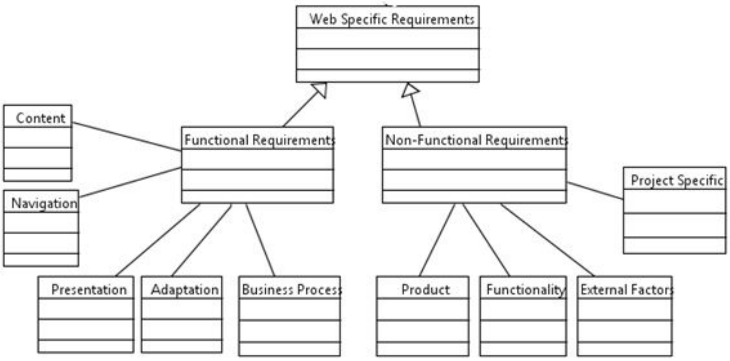


Fig. 3. Web Application Requirements Classification

4.1 Functional Web Requirements

The functional requirements are defined as requirements referring to the functionalities and behaviour of the system. They are classified in context of web as:

a) *Content requirements*: These requirements specify what information or ideas are to be communicated to the user through web application. The content includes text, graphics, images, audio and video data that has to be provided by web application. The content requirements for a e-bookshop can be ‘provide image of the book cover and information about the author’, ’provide information about the organization’s history on a web page’

b) *Navigation Requirements* Navigation is defined as finding your way around a website. The requirements state the navigation structure and navigable elements. It will define what all web pages are linked with each other and how are the links provided. For example, ‘connect a book with other books by the same author, and books on the same subject’.

c) *Presentation Requirements*: These cover the visual elements and interface layouts. The stakeholders can give some insight on the aesthetics of web applications, the physical positioning of the graphics, the colour scheme or style.

d) *Adaptation Requirements:* The website has to adapt itself depending upon user or environmental profile. The personalization can be done by profiling of users, regions etc. and suitably changing the content/ presentation for it.

e) *Business Process Requirements*: These are the requirements required to accomplish the structured activities or tasks required to serve a particular goal, and focuses on the operational purpose for which web application is being created. It can be refined into User or system operation. *User operation requirements* are the tasks that user will be doing like pressing buttons, browsing, searching etc. These operations are observable by the end-user. These operations are directly initiated by the user like 'posting comments on a blog', 'add items to shopping cart'. *System operations* are the tasks done by the system at the back end like database operations so that user operations can be completed. The system operations are like 'authenticating the user', 'validating a financial transaction', 'track web usage of the user for personalization'.

4.2 Non-functional Web Requirements

The term "non-functional requirements" is used to delineate requirements focusing on "how good" software does something as opposed to the functional requirements, which focus on "what" the software does." [26]. However, we simply state that Non-functional requirements are the requirements that specifies the criteria used to adjudge the system. This should be contrasted with functional requirements that focus only on the operational aspects of the system, i.e. which are needed for a function to operate. It is imperative that unless the non-functional requirements are satisfied, the product is of no use for example, if the information presented in the web application can't be comprehended by the users, it doesn't serve its purpose or the web application doesn't fulfil the soft goals of the organization like increase the profitability; the entire design, presentation of the Web application goes waste. Much work has been done for classification of quality and non-functional requirements[27],[28],[29],[30],[31]. However, in view of web applications no concise categorization of important non-functional requirements exists that would help the engineers create an eminent product. In this paper, we classify and explain the non-functional requirements based on the concerns- product, functionality, external factors and project specific concerns. The categorization is summarized in Table 1.

Table 1. Non-functional Requirements for Web applications

Concern	**NFRs**	
Product	Usability, Conformance, Security, Efficiency	
Functionality	Content	Credibility,Readability,completeness,Communicativeness,up-to-date
	Navigation	Accessibility, Consistency, Predictability, Relevance, Convenience
	Presentation	Attractiveness, Relevance, Clarity, Consistency
	Adaptation	Customizability, Suitability
	Business Process	Responsiveness, Simplicity, Unambiguity
External Factors	Organizational	Objectives
	Actor	User friendliness, Empathetic, Understandability
	Legal	Conformance to standards, Legal issues
	Environmental	Compatibility, Sustainability
Project Specific	Resource Constraints, Cost, Human Proficiency	

1. **Product:** The non-functional requirements enumerated in this category describe the quality expectations from the final product i.e. web application. We propose that a web application in totality is judged for its usability, conformance and performance. *Usability* The International Standards Organization's (ISO) defines usability as: "the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use" [32]. The usability for web application has similar meaning and it is used to measure the satisfaction of the user.

Conformance is whether the product delivered suffices all the initially set targets entirely. For example, if it was decided in initial meetings with participants/contributors/ that web application will provide songs playback that can only be availed online. In the final product, song can be played online but with a right click it can be saved in the user's disk, it means the requirement is not met properly.

Security is protection of the product's sensitive content and provide secure mode of data transmission and guard it against external threats. This is a very important requirement for web applications, because the data has to be transmitted through the network. The transactions should take place on secure mode, and encryption techniques need to be applied for transmitting and receiving data.

Efficiency of web application is how fast it loads the pages, graphics and responsiveness.

2. Functionality In this category, we explain the qualitative characteristics expected out of the operations in web applications that captures "how good" a system should function. We describe below how for each functional requirement, its qualitative expectations can be captured.

a) Content

Credibility according to the dictionary means whether the information provided is from a trusted source and correct and whether it can be relied upon. Similarly, we state that credibility of the information presented in the website is its reliability, correctness and a surety that it is from a trusted source.

Readability is defined as a measure of the ease of reading and understanding the information from web page, comprehensibility or understandability of written text.

Communicativeness is defined as the ability to communicate the intended information or idea effectively.

Up-to-date means the content should be updated as per the nature of information it is exhibiting. For example, for a news website the latest news should be reported in a matter of hours. Likewise a website for selling mobile phones should have the images, price and reviews for all the latest mobile phone models.

Completeness refers to the totality of the information posted on the web page.

b) Navigation

Accessibility is defined as approachability of hyperlinks when user want to navigate away from a page to a desired or unexplored information.

Consistency is defined as uniformity in positioning of the link on every web page , like we can say that the list of links always appear on left side of the web pages and in top row expandable list of links can be seen.

Predictability is defined as after accessing one page of web application user's can easily guess the placement and content of the hyperlinks of other web pages.

Relevance means the navigation links appropriate to the messages in the web page are displayed. The links have to be semantically related to complete a cognitive or operational task. For example, in an educational website the hyperlink to previous year question papers should be placed in the Web page related to examinations.

Convenience: We define the convenience of navigation links as the ease of reach and the prominence of links on a web page.

c) Presentation

Attractiveness is defined as web application's power of pleasing or appealing the intended users with the look and feel of the web pages.

*Relevance is defined as p*ertinence or suitability of the visual interface of web application with its purpose. For example, application for kids can have bold and cheerful appearance but a professional application should have an corporate appeal.

Clarity is stated as clearness and comprehensibility of appearance of web application.

Consistency is defined as the logical coherence of appearance of various parts of a web application. It means that different web pages of the same website should have similar look and feel like company's logo placement at a uniform place, the color schemes used for menu's, the font etc.

d) Adaptation

Customizability according to dictionary means the ability to be modified dynamically to meet the individual requirements. In context of web we define it as the ability of web application to modify its contents/presentation/process dynamically according to the user profile and other factors. For example, in an online book store application depending upon the navigational usage of the user, he can be given suggestions of similar books using web mining techniques.

Suitability of a web application is defined as the quality of having the properties that are right for a specific purpose. i.e. After the application has personalized to individual needs, the outcome's suitability to the purpose or the user's profile is also significant. The customization or personalization should be such that it appropriate for the needs of the user's to maximum possible extent.

e) Business process

Responsiveness is defined as the quality of readily reacting to any stimulus like pressing buttons, playing videos, performing tasks etc. in web application by the user/ system.

*Simplicity means that th*e business processes that involve both system and user operations must not be complex and be uncompounded to make interaction with the user easy. The complex tasks should be broken into straightforward stepwise parts to ease the procedure.

Unambiguity is defined as complete lack of confusion or uncertainty in the business process. The process should be clear and concise.

3. External Factors In this category, we describe non-functional requirements that are outside the system but greatly affect the ability of web application. The organizational factors, actor's expectations, legal requirements and environmental factors come under this category. They are described in detail below.

a) Organizational

Objectives: Web application is expected to meet organizational long term and short term objectives like profitability, expansion of business, attract new clients etc.

b) Actor:

User friendly: The ease of learning and memorability of web application. This means that the usage of web application should preferably not require special training and its operations should be memorable so that repeated visits are more easy to use.

Empathetic means understanding other's situation in dictionary. In context of Actor's requirement from web application it means that the creation of the web application should be done through identification with and understanding of users' situation and motives. It is a very significant non-functional requirement because in case of web applications, due to vast audience, the web designer should have some understanding of user's needs to make the application acceptable.

Understandability means ability of the users to comprehend the functioning of web application.

c) Legal

Conformance to standards: Besides completing the operational requirements, web application's abide by the relevant legal standards.

Legal issues: Various other legal issues like patenting, copyright etc. have to be looked into while making web applications.

d) Environmental:

Compatibility: Web applications interface and interaction with existing software / hardware shouldn't change the intended behaviour of the application or disrupt normal functioning of interacting items.

Sustainability: Web application should be capable of being sustainable or maintainable. This property involves web applications maintainability like change of the content , repair of some business process, change of technology etc. Also, web application should be able to work with the changes in the environment, like a new browser, a different platform.

4. **Project Specific:** In this category the non-functional requirements related to the project are listed. The state in which the project has to operate, with kind of resources, budget and human expert skills. They have been categorized as Resource Constraints, cost and human proficiency.

Resource Constraints:- The engineering of web application is dependent on many resources like hardware, software, time limit etc.

Cost- The budget is also a limitation and can affect the creation of web application like in choice of alternatives.

Human Proficiency- The quality of web application is principally dependent on the knowledge and experience of the engineers and designers creating it.

5 Case Study: Web Based Education System

There are many kinds of web applications with different set of requirements and demands [33]. For exhibiting the model and its benefits we take an example of Web application for providing Education. The organization's primary goal is to provide web based distance education. Also, it's the aspiration of the management of the organization that the web application increases the enrollment of the students and runs in minimal cost. The analysis should also consider specific issues like electricity, connectivity and local language. Web application must provide tutorials, take assignments, conduct on line exams and also have facility for doubt clarification.

After eliciting the goals of the organization, the goals are analyzed and modeled using User Requirements Notation. Fig. 4 shows the GRL diagram. The main goal of the organization is to provide education with the following objectives in mind: Provide subject tutorials, Provide Assignments, Clarify Doubts, Conduct online Examination, Increase enrollment, Increase profit, Minimize cost and expand the reach. Amongst the above-said objectives, the last four objectives- increase enrollment, increase profit, minimize cost, expand the reach are the softgoals. In the GRL model, the softgoal increase usability is explored in detail. The target audience of the website will be students that may have diverse backgrounds, so it would be beneficial to give an online demo on usage of the website Also, the application can also overcome the language barrier and can give options of translating the contents to local language. The cost can be minimized by choosing alternatives that are pocket friendly. The goal of clarifying the doubts can be done in various ways depending upon various factors. The clarify doubts subgoal can be done by using email, discussion forum, chat or video conferencing. The email and discussion forum are easy on cost as they don't require the faculty to be present online and can be done on convenience of time. However, to improve the students' experience, i.e. to increase usability of the application, live chat or video conferencing are better options. Various alternatives can be weighed and chosen in conformance with the stakeholders.

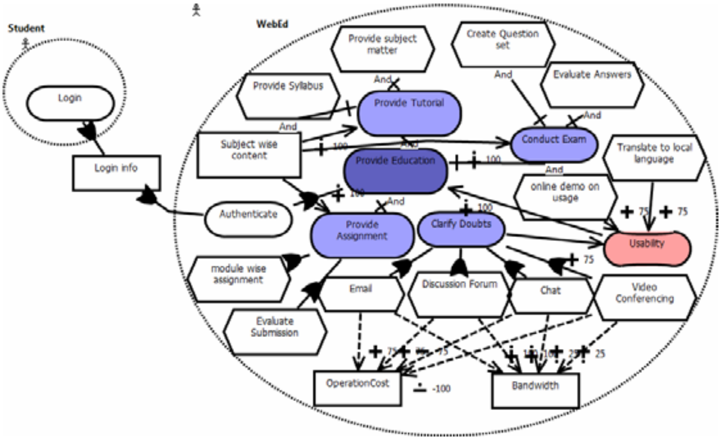


Fig. 4. GRL model for WebEd system

For analyzing this further, the study of this tasks scenario can be done and its operations can be visualized (Fig. 5). A combination of GRL and UCM helps in making decisions, e.g., it can be dynamically decided whether a registered student will be allowed to chat or do video conferencing. The scenarios help in visualizing the situation and UCMs provide a clear vision of the same. We have modeled the example using jUCMNAV tool [34] that supports both GRL and UCM. It is an eclipse based tool for modeling User Requirements Notation.

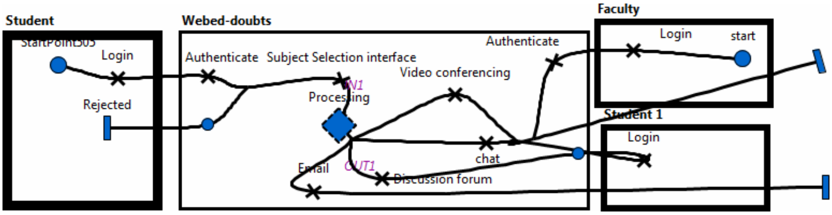


Fig. 5. UCM diagram showing scenarios to Clarify doubts goal

Using Goal analysis, many conflicts amongst goals are resolved and there is reasoning of goals as explored by researchers in papers[35][36]. The non-functional or the quality factors also need reasoning as discussed by Bedi et al in [37] and Jureta et al in paper [38].We would cover this aspect in our future work to ease the incroporation the Goal oriented Requirements Engineering in the existing Web Engineering methodologies.

6 Conclusion

A framework GOREWEB for analyzing web application requirements from a goal driven perspective is presented in this paper. The GOREWEB framework uses User Requirements Notation (URN) for analysis of goals and scenarios. The stakeholders' expectations and goals are captured and modeled using Goal Requirements Language (GRL). GRL models both functional goals and softgoals. The goals are operationalized by tasks, which are then modeled by creating user scenarios using Use Case Maps (UCM). After careful analysis of both GRL diagrams and UCM, the goals are mapped to requirements of web application. The user goals can be both hard goals and soft goals; hence it is needed to map them to functional and non-functional requirements. A classification of functional and non functional requirements in context of web applications has also been provided in the paper. The framework is described in detail with the help of a case study on Web based education. As Goal driven approaches are closer to the stakeholder's thoughts, the presented framework will help the designers to have clarity of requirements of web applications knowing the goals from the early stages of development.

References

1. Srivastava, S., Chawla, S.: Multifaceted classification of websites for goal oriented requirement engineering. In: Ranka, S., Banerjee, A., Biswas, K.K., Dua, S., Mishra, P., Moona, R., Poon, S.-H., Wang, C.-L. (eds.) IC3 2010. CCIS, vol. 94, pp. 479–485. Springer, Heidelberg (2010)

2. Gause, D.C., Weinberg, G.M.: Exploring requirements: quality before design. Dorset House, New York (1989)
3. Somerville, I.: Software Engineering, 7th edn., ch. 6 (2004)
4. Mylopoulos, J., Chung, L., Yu, E.: 'From Object-Oriented to Goal-Oriented Requirements Analysis'. Communications of the ACM 42(1) (1999)
5. Chung, L., do Prado Leite, J.C.S.: On non-functional requirements in software engineering. In: Borgida, A.T., Chaudhri, V.K., Giorgini, P., Yu, E.S. (eds.) Conceptual Modeling: Foundations and Applications. LNCS, vol. 5600, pp. 363–379. Springer, Heidelberg (2009)
6. Rolland, C., Prakash, N.: From Conceptual Modeling to Requirements Engineering
7. Brinck, T., Gergle, D., Wood, S.D.: Usability for the Web: Designing Web Sites that Work. Morgan-Kauffmann, San Francisco (2002)
8. Liu, L., Yu, E.: From Requirements to Architectural Design–Using Goals and Scenarios
9. van Lamsweerde, A.: Goal-Oriented Requirements Engineering: A Guided Tour. In: 5th Intl. Symp. Req. Eng. (2001)
10. van Lamsweerde, A.: GORE: From Research to practice. In: 12th IEEE International Requirements Engineering Conference, Kyoto (2004)
11. Keller, S.E., Kahn, L.G., Panara, R.B.: Specifying Software Quality Requirements with Metrics. In: Thayer, R.H., Dorfman, M. (eds.) Tutorial: System and Software Requirements Enginering, pp. 145–163. IEEE Computer Society Press, Los Alamitos (1990)
12. Yu, E.S.K.: Modelling Organizations for Information Systems Requirements Engineering. In: 1st Intl Symp. on Requirements Engineering, vol. 0, IEEE, Los Alamitos (1993)
13. Letier, E., van Lamsweerde, A.: Deriving Operational Software Specifications from System Goals. In: 10th ACM Symp. On the Foundations of Software Engineering, Charleston (2002)
14. van Lamsweerde, A., Willemet, L.: Inferring Declarative Requirements Specifications from Operational Scenarios. IEEE Trans. on Sofware Engineering (1998)
15. Letier, E., van Lamsweerde, A.: Reasoning about Partial Goal Satisfaction for Requirements and Design Engineering. In: 12th ACM Symp. on the Foundations of Software Eng. (2004)
16. Rolland, C., Grosz, G., Kla, R.: Experience With Goal-Scenario Couplin. In: Requirements Engineering. In: IEEE International Symposium on Requirements Engineering, Limerick, Ireland (1998)
17. Castro, J., Kolp, M., Mylopoulos, J.: Towards Requirements-driven Information Systems Engineering: the Tropos Project. Information Systems 27, 365–389 (2002)
18. Antoń, A.: Goal identification and refinement in the specification of software-based information systems. Dissertation, Georgia Institute of Technology, Atlanta, USA (1997)
19. Bolchini, D., Paolini, P.: Goal-Driven Requirements Analysis for Hypermedia-intensive Web Applications. Requirements Engineering Journal 9, 85–103 (2004); RE 2003 Special Issue
20. Jaap, et al.: e-Service design using i* and e3 value modeling. IEEE software 23(3) (2006)
21. Azam, et al.: Integrating value based requirements engineering models to WebML using VIP business modeling framework (2007)
22. Shailey, C., Sangeeta, S.: Goal driven Requirements engineering: A comparative study. In: CEE 2010 (2010) (accepted)
23. Amyot, D.: Introduction to the User Requirements Notation: Learning by Example. Computer Networks 42(3), 285–301 (2003)

24. ITU-T, Recommendation Z.151 (11/08): User Requirements Notation (URN) – Lanuage Definition
25. Escalona, M.J., Koch, N.: Requirements Engineering for Web Applications: A Comparative Study. Journal on Web Engineering 2(3), 193–212 (2004)
26. Paech, B., Kerkow, D.: Non-Functional Requirements Engineering - Quality is Essential. In: 10th Anniversary International Workshop on Requirements Engineering: Foundation for Software Quality, REFSQ 2004 (2004), `http://www.sse.uni-essen.de/refsq/downloads/toc-refsq04.pdf`
27. Boehm: Characteristics of a software quality. North Holland, New York (1978)
28. Gillies, A.C.: Modelling software quality in the commercial environment. Software Quality Journal 1, 175–191 (1992)
29. McCall, J.A., et al.: Concepts and definitions of software quality, Factors in software quality. In: NTIS, vol. 1 (1977)
30. Roman, G.-C.: A Taxonomy of Current Issues in Requirements Engineering. IEEE Computer, 14–21 (April 1985)
31. Grady, R., Caswell, D.: Software Metrics: Establishing a Company-wide Program. Prentice-Hall, Englewood Cliffs (1987)
32. Karat, J.: Evolving the Scope of User-centered Design. Communications of the ACM 40(7), 33–38 (1997)
33. Sangeeta, S., Shailey, C.: Goal oriented requirements analysis for Web applications. In: ICCSM 2010, Manila, December 4-5. IEEE, Los Alamitos (2010)
34. Roy, J.-F., Kealey, J., Amyot, D.: Towards integrated tool support for the user requirements notation. In: Gotzhein, R., Reed, R. (eds.) SAM 2006. LNCS, vol. 4320, pp. 198–215. Springer, Heidelberg (2006), `http://jucmnav.softwareengineering.ca/`
35. van Lamsweerde, A., Letier, E.: Handling Obstacles in Goal-Oriented Requirements Engineering
36. Jureta, I.J., Mylopoulos, J., Faulkner, S.: Analysis of Multi-Party Agreement in Requirements Validation. In: ACE 2009 (2009)
37. Bedi, P., Gaur, V.: Prioritizing quality specification of Multiagent systems. In: Proceedings of World Congress on Engineering, WCE 2007, vol. 1 (2007)
38. Jureta, I.J., Faulkner, S., Schobbens, P.-Y.: Clear justification of modeling decisions for goal-oriented requirements engineering. Requir. Eng. 13(2), 7–115 (2008)

Malware Attacks on Smartphones and Their Classification Based Detection

Anand Gupta, Spandan Dutta, and Vivek Mangla

Department of Computer Engineering
Netaji Subhas Institute Of Technology, New Delhi, India
omaranand@nsitonline.in, spandan1989@gmail.com,
vivekmangla89@gmail.com

Abstract. Smartphones nowadays, with colossally large number of users have become very prominent[1]. Furthermore, this increasing prominence goes arm in arm with the rising number of malwares[2], thus making it inevitable to take cognizance of the need for an efficient malware detection mechanism. However, sundry former associated works like [3] & [4] for malware detection, have not cited a novel strategy which we feel can be attributed to the lack of malware classification in them. Fundamentally, classification of malwares provides a head start to the detection mechanism by curtailing the search space & the processing time of the detection mechanism. So in order to accomplish the malware classification, we develop few malwares and discuss their behavior and aftereffects on the device. And then we utilize the resource victimized by these malware on the phone as base for classification and allocate same class to those malwares that affect same resource. Finally by employing the aforementioned malware classification, we outline a strategy for their detection. Experimentation of the detection scheme on the malwares with and without classification reveals that with classification the real and CPU time consumed by detection process are almost 45% and 22% of the respective times without classification, which thus elucidates the fact that classification based malware detection in future can be employed as a propitious tool.

Keywords: Malwares, Smartphones, Resources, Malware Detection, Malware Classification.

1 Introduction

According to the research, Smartphone: King of Convergence, conducted by Parks Associates, as smartphones are finding more and more users for themselves, the number of smartphones worldwide is forecasted to surpass a billion by 2014[1]. Also, according to Harry Wang, Director of Health and Mobile Product Research at Parks Associates, the smartphone ownership prior to iPhone's launch, raised from 9% in 2007 to 28% in 2009 among U.S. broadband households[5](See Fig. 1).

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 242–253, 2011.

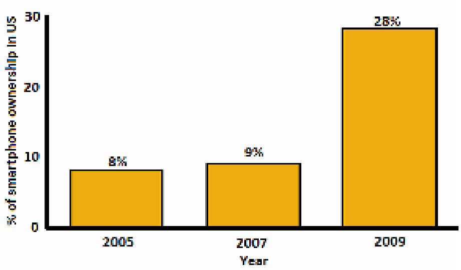


Fig. 1. Smartphone adoption rate-U.S. broadband households[5]

But this increasing popularity of smartphones has witnessed a major rise in the number of malware attacks also. Moreover, the malware attacks on smartphones are more troublesome than those on the desktops due to the limited battery on the device and the excessive amount of personal & confidential data stored by the users. In this regard, analysts have witnessed a major hike in the number of malwares to exploit the smartphone's vulnerability to a malware attack[2]. According to Lookout Mobile Security Firm, an Android Trojan, christened Geinimi, in China, forwards the personal data stored on the phone to remote servers[6]. In another survey, conducted by Kaspersky, according to the report Kaspersky Security Bulletin 2006, 43 variants of different mobile viruses appeared in February – April, 2006 alone[7] (See Fig. 2).

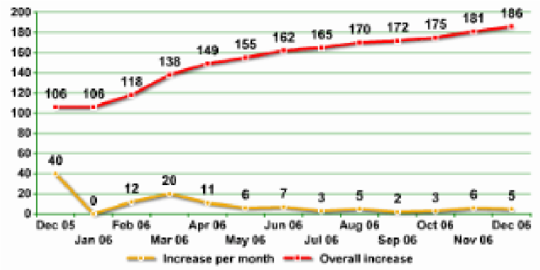


Fig. 2. Increase in number of mobile virus variants in 2006[7]

All the aforementioned surveys reveal the alarming rate at which the infamous malware attacks are proliferating and thus necessitate the development of a detection mechanism to fight back. Although the former associated works [3] and [4] have suggested malware detection mechanisms, but there are few frailties and bottlenecks associated with their approaches that have been discussed in the subsequent section. Also, their approach lacks malware classification, presence of which can eventually assist in working out an efficient and a speedy detection mechanism. So we render an in-depth study of their work, assay the bottlenecks and attempt for amelioration.

1.1 Prior Work on Malware Detection

The strategy followed by Bickford et al[3] elaborates on malware's implications and detection by exhibiting a few attacks while the strategy followed by Dixon et al[4] discusses malware detection only. The prior work [3], by utilizing Neo Freerunner smartphone running the Openmoko Linux distribution exhibits three malware attacks and elucidates their behavior and social implications. The approach [3], as a demonstration for malware attacks, records confidential conversations of the user from a remote site, messages the user's current geo location using GPS to the traitor and exhausts the battery by repeatedly switching on and off services like GPS and Bluetooth that consume a lot of power. With the objective to detect the malwares, [3] proposes the use of Virtual machine monitors i.e. the smartphone's operating system and the monitor execute on different virtual machines (VM). The monitor queries the virtual machine that runs the phone's operating system, accesses and scans the phone's memory contents for malignant files. On the other hand, the latter approach [4], unlike the previous one, utilizes Android platform and avers that any malware detection mechanism that can be implemented on a desktop, cannot be implemented on a phone as it would consume a lot of power and hence the battery would deplete very soon. So instead of running the detection mechanism on a smartphone, the approach [4] suggests to run the same on a desktop. And, to refrain from downloading all the files from phone to desktop, approach [4] suggests providing hashes to all files and storing them on the desktop as well. Whenever the phone is connected to desktop, hashes are always sent from phone and juxtaposed with the ones stored on desktop. Only the files that have changed over the period of time are downloaded and scanned. But associated with these prior works [3] & [4], there also exists few bottlenecks and frailties in the development of a novel detection mechanism. These bottlenecks and frailties spurred us to extend their work by demonstrating, classifying and detecting few more rudimentary malware attacks and elucidating their behavior and aftereffects on the smartphone.

1.2 Motivation

The subsequent listed shortcomings in the former associated works have motivated us to develop a method that tries to vanquish them.

1. A malware detection scheme basically searches the input file for a particular keyword, from the database of all the available keywords, alternatively. This database, in simpler words symbolizes the search space which the detection mechanism utilizes. Detection mechanism queries the search space for a keyword, searches for the keyword in the file and if the keyword is found, it reports the file as a malware. If the keyword is not found, it again repeats the process with next keyword from the search space. Now, narrower the search space, faster the detection mechanism would be, as now it has to search for fewer keywords. So basically, the malwares belonging to same category are treated as single entity and it now requires a single keyword to identify them. This is the crucial role which the malware classification plays in the improvisation of detection mechanism, the curtailment of search space. But both the former associated works have not made endeavors for malware classification.

2. None of the former associated works have presented a detection approach that can serve as a promising tool in future. The approach [3] proposes VMM method for detection when no platform at present supports VMM installation[3]. While on the other hand, the approach [4] either downloads all the files from phone to desktop before scanning or provides hashes of all files to desktop for comparing the old and new hashes. But none of the proposed mechanisms in [4] seems to be profitable. The former case consumes a lot of time and the latter one is bound to fail when the malware conceals the hashes for modified files from the detector[4].
3. Although the method [3] discusses the behavior and aftereffects of some rudimentary malwares, still it does not ponder over sundry other basic malwares that behave in a much different manner and exploit or affect umpteen other vital and basic resources on the phone. Their effects on various aspects like user's privacy, system behavior or hardware have been neglected.

On that account, through our research work we endeavor to augment the former associated works by providing a detection mechanism that can serve as a propitious tool in the future.

1.3 Augmentation

In the present paper, we endeavor to further augment the work done in [3] and [4] by developing few more rudimentary malwares and classifying them, on the basis of resource they victimize. We also discuss the behavior and aftereffects of those malwares. And by presenting a proficient malware detection mechanism based on malware classification, we accomplish our aim of malware detection. Our detection mechanism, unlike approaches [3] and [4], does not install VMM or download files from the phone to desktop. And lastly, by scrutinizing quantitatively and qualitatively, we advocate our detection mechanism with the help of graphs and consummate the paper.

1.4 Framework of the Paper

With aim to classify the malwares and work out a proficient and speedy malware detection mechanism, we have organized the rest of the paper into sections, which are as follows: Section 2 deals with development of some malwares, discussion of their behavior and aftereffects on the device. Apart from malware development, it also covers the classification of these developed malwares. Section 3 covers the development of a novel malware detection mechanism based on malware classification presented in the aforementioned section. The same section also scrutinizes the authenticity of the detection mechanism qualitatively and quantitatively, by means of experimental tests and avers how the classification of malwares has helped in the evolution of an efficient detection mechanism. And finally, with conclusion and possible future work, Section 4 consummates the paper.

2 Malware Attacks, Implications and Classification

Before taking up the torch of malware development, classification and detection, we would like to emphasize on the tools utilized for the same.

- To demonstrate the malware attacks and detect the malwares, we have utilized the Google's Android 2.1 Éclair source code[8].
- The detection mechanism works through a desktop running on Linux platform only and requires the phone to be connected to desktop for file scanning purpose.

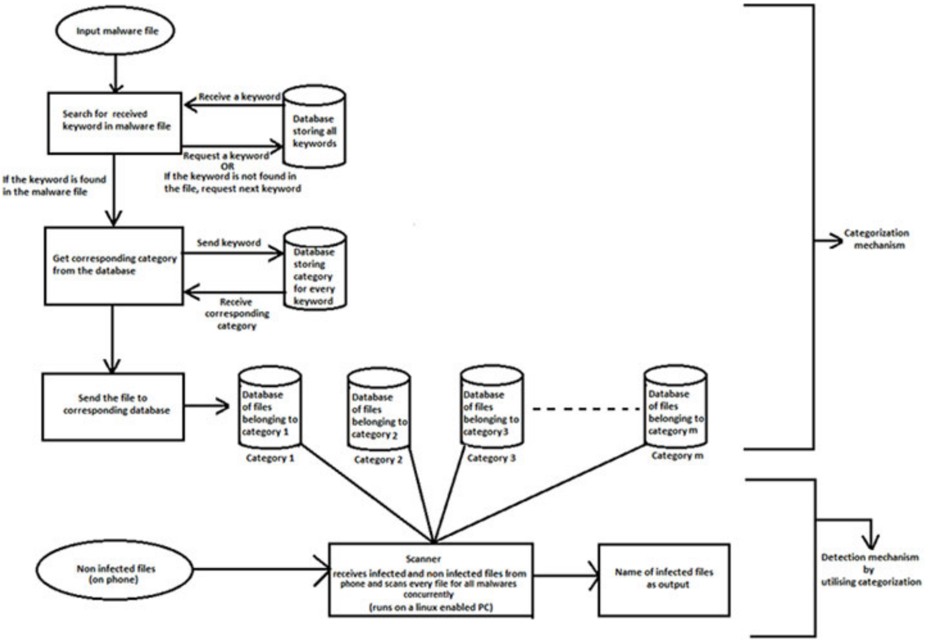

Input malware file
Search for received keyword in malware file
Receive a keyword
Database storing all keywords
Request a keyword
OR
If the keyword is not found in the file, request next keyword
If the keyword is found in the malware file
Get corresponding category from the database
Send keyword
Database storing category for every keyword
Receive corresponding category
Send the file to corresponding database
Database of files belonging to category 1
Category 1
Database of files belonging to category 2
Category 2
Database of files belonging to category3
Category 3
Database of files belonging to category m
Category m
Categorization mechanism
Non infected files (on phone)
Scanner
receives infected and non infected files from phone and scans every file for all malwares concurrently
(runs on a linux enabled PC)
Name of infected files as output
Detection mechanism by utilising categorization

Fig. 3. Framework for malware classification and detection

Study of behavior and aftereffects of a malware attack on a smartphone can be of colossal help in working out a malware detection mechanism. Also for categorizing malwares, we require some sample malwares on which we can test our categorization approach. So for this purpose, we have developed umpteen malwares that affect the device with dire consequences. We discuss their behavior and social implications in following sub-sections.

The framework presented in Fig. 3 represents a block diagram of our complete strategy for malware classification and detection. The framework comprises of two stages – Classification and Detection. We try to elaborate the former one first. Every malware, in the form of a malicious code developed by the traitor, victimizes a particular resource on the smartphone. We hence utilize this victimized resource as a base for classification. Sundry malwares may fall under the same class if they affect the same resource. Since a keyword corresponding to every malware can be treated as its signature, in order to classify the malware, we select every keyword from the database alternatively and search for that keyword in the malicious file, until the keyword is found. Upon finding the keyword, we look for the class corresponding to that keyword from another database. This way we get the class for that malware. So, we have developed total thirteen malwares and by utilizing the victimized resource as the base for classification, eight classes have been framed. Classes along with their sample malwares have been listed in Table 1.

Table 1. Malwares developed and their classes

Category	Sample Malware Developed
Boot tampering malware	Boot animator
	Restart
Network tampering malware	Airplane mode
	Poor connectivity
Sdcard tampering malware	Format the sdcard
	Remove the sdcard virtually
Battery tampering malware	Low battery
Privacy tampering malware	Providing all the text & multimedia messages to traitor
	Calling a predefined premium rate number
Security tampering malware	Mail and delete all the contacts
	Providing the browser history, saved username & password to the traitor
Bluetooth tampering malware	Playing with Bluetooth
Event tampering malware	Displaying bogus-arbitrary series of user events

We now discuss the behavior and implications of the sample malwares on the device.

2.1 Boot Animator

This malware displays the boot time screen and hides the windows, which the user has been using. The normal processes still continue to execute in the background but the phone appears to have hanged to the user. The situation is very much similar to a screensaver of which the user is not able to come out. Even if the user reboots the phone, then also the screen persists. Other than keeping the phone switched off, the user does not have any other alternative to emancipate himself from the malware.

2.2 Restart

The second malware in the boot tampering class keeps on restarting the phone without the user's permission. Once the phone restarts, the restart command is again issued by the malware and this process continues indefinitely. Since the malware does not allow the phone to come out of this restart process, the user cannot initiate a new job or carry on with an already initiated one.

2.3 Airplane Mode

This malware belongs to the class of network tampering malwares and exploits the airplane mode settings on a smartphone. The airplane mode on the phones, does not allow them to propagate any sort of communications and radio signals, as it is illicit to use cell phones during a flight. This malware continuously disables and enables the airplane mode by splashing the airplane mode window on the top of every other working window. Even this malware also, does not allow the user to perform any other task as every time his instructions or operations are overlooked and the airplane mode window is displayed which leads to utmost turmoil. Also the user cannot connect to any sort of network as the network keeps on toggling between the absent and the present state, even when he is not in a flight. However, rebooting the phone also does not help the user in anyway.

2.4 Poor Connectivity

Phones are some of the very vital tools through which the user can keep himself in touch with remote people. By means of messages and phone calls it becomes very easy for the user to be in touch with friends, colleagues and family at remote locations. However, making any call or sending messages from a phone requires the availability of network. This malware exploits this requirement and despite the network being available, shows almost zero connectivity or no network availability. Thus this malware prohibits the user from making any call or sending messages.

2.5 Format the Sdcard

Sdcard is the external storage used to store data on the device as the internal storage on the same is very limited. This malware formats the sdcard i.e. deletes all the contents of the sdcard permanently and thus leads to a permanent loss of files including the important ones also.

2.6 Remove the Sdcard Virtually

Another malware for the sdcard tampering malwares class dismounts the sdcard i.e. the phone cannot recognize the sdcard anymore. Despite of the fact, that sdcard is present in the allocated slot on the device, the malware still reports sdcard not found.

2.7 Low Battery

One of the most essential parts of a smartphone is battery. Battery should be substantially charged so that phone can run in a smooth manner. Also the operation of battery in ideal conditions ensures a longer battery life which in turn ensures longer life of the smartphone. The malware, Low Battery, attempts to ruin this essential part of the smartphone. It hides the battery symbol, shows a window displaying low battery and prompts the user to charge the same even when the battery is fully or substantially charged. Also, the low battery window is shown continuously like an abyss, even after the receiving the response from the user. This malware continues to execute even after the reboot, thus irking the user beyond belief.

A major complication of the situation to be observed is, if the phone is in charging mode i.e. connected to the power supply, the malware would still ask for connecting the charger which would leave the user baffled. And if the phone is not in charging mode and the user connects the charger at the behest of malware, then the battery will be charged beyond the permissible limit, which ultimately would deteriorate the battery life. This may pose dire consequences on the battery, like being overcharged, or the temperature of the battery may exceed the maximum permissible value.

2.8 Providing All the Text and Multimedia Messages to the Traitor

Text and multimedia messages on a phone may contain eminently sensitive, personal and confidential information, which is of utmost importance to the owner. Since these messages are accessible by the owner of the smartphone only and are also meant to be kept private, these messages epitomize the owner's privacy. With sole aim to tamper this privacy root and branch, this malware mails all the messages, whether they are received messages or sent messages or saved messages, to the traitor surreptitiously without the owner's knowledge as the malware works in the background.

2.9 Calling a Predefined Premium Rate Number[9]

This malware makes a call to a premium rate number[9] predefined by the creator of the malware, where the call is always received without the user's knowledge. As aftereffects, the malware reduces the talk time balance available to the user considerably & may also pass money from the user's account to that of the cyber criminals. Also, the pie-chart (See Fig.4)[10] detailing the battery usage by sundry processes on an Android device, reveals that the voice calls consume considerable amount of battery. So the pie chart helps to decipher that this malware drains the battery also.

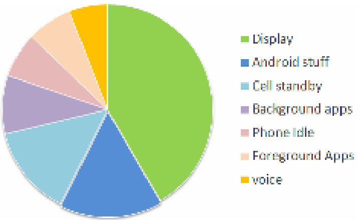


Fig. 4. Battery usage on Android[10]

2.10 Mail and Delete All the Contacts

The contacts and various other information associated with them like email address, residence address, stored in the device are very important for the user. The first malware in the security tampering class, mails all the contacts from the phone and every sort of information associated with them to the traitor and then deletes them permanently from the phone. One of the dire consequences of this malware is that deleted contacts cannot be restored even if the malware is detected and removed.

2.11 Providing Browser History, Saved Usernames and Passwords to the Traitor

This malware stealthily sends the browsing history, saved username and password of the user to the traitor and creates a vicious circle of illicit usage of victim's email account. Thus, this malware makes sure that the user's security is tampered entirely by making such confidential information available to the malware's developer.

2.12 Playing with Bluetooth

On a smartphone, switching the Bluetooth on and off and running the Bluetooth service consume a lot of power[3]. The malware developed in this class exploits the Bluetooth service and displays a window acquainting the user that Bluetooth is on and also urges him to turn it off. This leaves the user baffled and he attempts to switch the Bluetooth on and off repetitively to emancipate himself from the malware. This repetitive starting and stopping the Bluetooth service consumes a lot of power and thus drains the battery.

2.13 Displaying Bogus-Arbitrary Series of User Events

On a smartphone, there are various events like click, touch, gestures and events associated with installed applications on the phone. This malware utilizes these events and generates bogus-arbitrary series of such events. It arbitrarily starts any event and displays the event's unusual and horrifying behavior. This malware even stress-test the applications whose events are started in a haphazard and repeatable manner, because of which the application may crash, receive an unhandled exception or generate an application not responding error. This malware basically modifies the system behavior very bizarrely and continues to execute even after the phone is rebooted by the user.

3 Malware Detection

A technique for malware detection is an eminently important tool to fight back a malware attack on the smartphone. To detect the malicious files developed in the aforementioned section, we have used a Linux enabled desktop to scan the files on the phone for different keywords which characterize the malware. Our detection mechanism could detect all the malignant files successfully but we now endeavor hard for metamorphosing our detection mechanism into an efficient one, just by utilizing the malware classification. Malware classification affects and improves the proficiency of malware detection mechanism by reducing the total number of keywords for which the scanner searches in the input file, as now malwares belonging to same class are treated as a single entity. Thus multiple malwares belonging to same class are now detected concurrently, which eventually decreases the processing time taken by detection mechanism and leads to a speedy detection. To establish this fact practically, we now present the experimental results.

3.1 Experimentation and Results

To begin with, we increase the total number of files to be scanned by the detection mechanism, applied with and without classification and then juxtapose the real time and CPU time consumed by the detection mechanism under both the situations. The test was implemented on a HP Pavilion, Intel(R), Core(TM) 2 Duo Processor, 1 GB RAM, Ubuntu 10.04 system in Android 2.1 Éclair source code[8]. The graphs in Fig. 5 juxtapose the real & CPU time consumed by detection mechanism in presence of classification with those when detection is carried out in absence of classification and helps in deciphering the considerable reduction in the processing time due to classification.

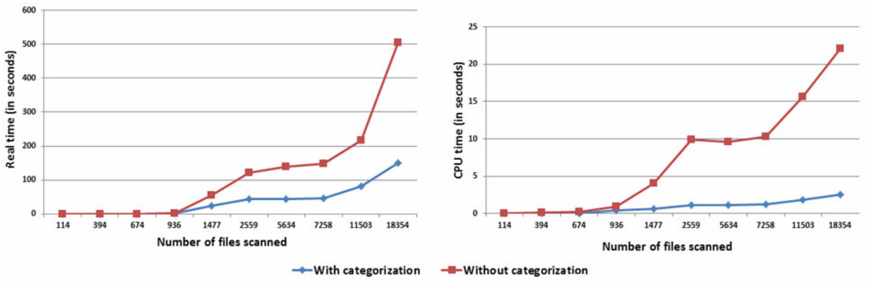


Fig. 5. Comparison of real & CPU time consumed by detection mechanism with & without malware classification

For scrutinizing our detection mechanism quantitatively, we utilize the graphs in Fig. 5 and the following relations:

$$R = real_t * 100 / real_t' \quad (1)$$

$$C = cpu_t * 100 / cpu_t' \quad (2)$$

R and C symbolize percentage ratio of real time and percentage ratio of CPU time consumed by detection mechanism with and without classification, respectively, $real_t$ and cpu_t symbolize real time and CPU time consumed by detection mechanism with classification, respectively and, $real_t'$ and cpu_t' symbolize real time and CPU time consumed by detection mechanism without classification, respectively. After detailed study of the graph with relations 1 and 2, we observe that the average value for R is 44.558%, i.e. the real time consumed by detection mechanism with classification is almost 45% of that consumed by it without classification and the average value for C is 21.562%, i.e. CPU time consumed by detection mechanism with classification is almost 22% of that consumed by it without classification. This summarizes the need for malware classification that leads to a considerably faster detection mechanism as real time in case of classification is almost 45% and the corresponding CPU time is almost 22% of those without classification.

Second test juxtaposes the values of R and C with respect to change in the total number of files scanned i.e. percentage ratio of real time and percentage ratio of CPU time with and without classification are plotted against the number of files scanned. The results are shown by the graphs in Fig. 6.

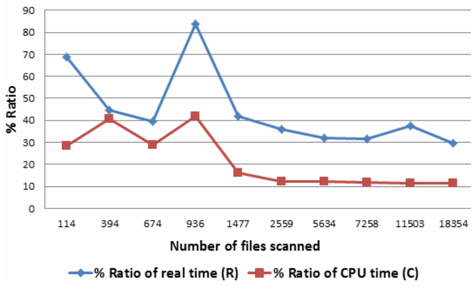


Fig. 6. Variation of % ratio of R and C (with and without classification of malwares) with total number of files scanned

Third test presents variation of the CPU time and the real time consumed by the detection mechanism with the number of malicious files to be detected. The total number of files scanned under every observation has been kept constant. The graphs in Fig. 7 illustrate the result of experiment.

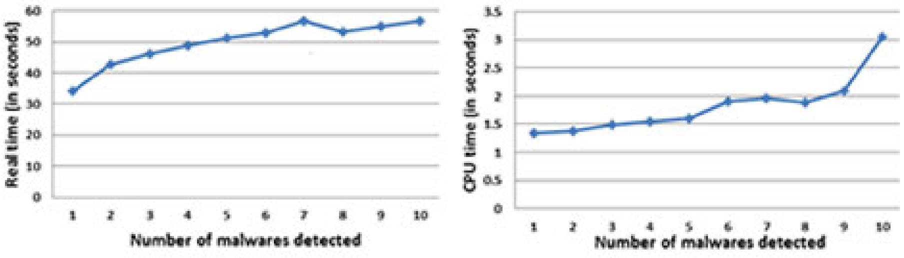


Fig. 7. Variation of real and CPU time consumed by detection mechanism with linearly increasing number of malwares on the phone

Hence our objective to implement an intelligent malware detection mechanism based on malware classification stands accomplished and justified by these tests.

4 Conclusion and Future Work

In the present paper, we hereby develop and discuss the implications & behavior of some rudimentary malwares on a smartphone, implement the malware classification and present a potent and robust detection mechanism based on malware classification. By utilizing the resource victimized by malware as the base for classification, we have discovered that the detection technique was able to identify the malignant files with reduced execution time. To evaluate the authenticity of the proposed detection technique, we have scrutinized the same by various tests and on the basis of results returned, we conclude that when the malwares belonging to same category are treated as a single entity, then the detection mechanism consumes almost 45% of real time and 22% of CPU time consumed when malwares are treated as separate individual entities. This thus justifies the improvement in the efficiency of detection mechanism due to malware classification. However, application of classification approach is constrained by those malwares that victimize multiple resources of the phone simultaneously, as for such malwares, multiple classes would be returned, which is deplorable. Also, since the detection mechanism cannot run on the smartphone, the phone needs to be connected via USB to a desktop that runs on Linux platform to scan the files, which is a major constraint. Thence, we aspire in future to extend successfully the application of our proposed classification mechanism, to those malwares that victimize multiple resources simultaneously & to run the detection mechanism on the phone, not on the desktop.

References

1. Number of Smartphone Users to Quadruple, Exceeding 1 Billion Worldwide by (2014), `http://www.marketwire.com/press-release/Number-of-Smartphone-Users-to-Quadruple-Exceeding-1-Billion-Worldwide-by-2014-1136308.htm`
2. Smartphone malware attacks soar by a third, `http://www.channel.hexus.net/content/item.php?item=28020`
3. Bickford, J., O'Hare, R., Baliga, A., Ganapathy, V., Iftode, L.: Rootkits on Smart Phones: Attacks, Implications and Opportunities. In: Proc: IEEE 11th Int. Workshop on Mobile Computing Systems & Applications (HotMobile 2010), Annapolis, MD, USA, pp. 49–54 (2010)
4. Dixon, B., Mishra, S.: On Rootkit and Malware Detection in Smartphones. In: Proc: IEEE Int. Conf. on Dependable Systems and Networks Workshops (DSN-W), Chicago, IL, USA, pp. 162–163 (2010)
5. `http://www.parksassociates.com/bento/shop/samples/parks-Smartphones.pdf`
6. Mobile security firm warns of new Android Trojan, `http://news.cnet.com/8301-1009_3-20026804-83.html#ixzz1Awkg4ah6`
7. Kaspersky Security Bulletin 2006: Mobile Malware, `http://www.securelist.com/en/analysis/204791922/Kaspersky_Security_Bulletin_2006_Mobile_malware`
8. Get Android source Code, `http://source.android.com/source/download.html`
9. Premium rate-telephone number, `http://www.en.wikipedia.org/wiki/Premium-rate_telephone_number`
10. Where oh where your battery goes, `http://www.wirelessnorth.ca/2010/02/11/where-oh-where-your-battery-goes/`

Retracted: Performance Analysis of Handover TCP Message in Mobile Wireless Networks

Ashutosh Kr Rai[1] and Rajnesh Singh[2]

[1] Shobhit University, Meerut, India
[2] Rajnesh Singh, Manav Bharti University, Solan, India
akrai.iimt@gmail.com, rajneshcdac.mtech@gmail.com

Abstract. Transmission Control Protocol (TCP) is known to suffer from performance degradation in mobile wireless environments, as such environments are prone to packet losses due to high bit error rates and mobility induced disconnections, because TCP is well reliable protocol for wired networks. In wired network packet loss due to congestion. Proposed scheme is a true end-to-end approach, based on the idea of exclusive handover message and is used for alleviating the degrading effect of host mobility on TCP performance. Experiments are performed using the Network Simulator (NS-2). The simulator has been extended to incorporate wireless link characteristics.

Keywords: TCP, EHM, Wireless Network, Snoop protocol and DSR protocols.

1 Introduction

The most popular transport layer protocol TCP on the Internet offers reliable byte stream service. In this, packets are cumulatively acknowledged (ACK) as they arrive in sequence and out of sequence packets will cause generation of duplicate ACKs. The sender detects a loss when multiple duplicate ACKs (usually 3) arrive, implying that packet was lost [1]. TCP was basically developed for implementing on wired networks where it has less bit error rates (BER) and hence less packet losses without considering mobility factors. For better performance of TCP in a mobile network, loss due to wireless link error must be detected immediately and transmission resumed as quickly as possible. TCP performance degrades, when a Mobile Host (MH) moves between networks and some packets are dropped or lost during handover. There were number of solutions proposed to improve the performance of TCP in wireless environment [2-10].

In proposed scheme, we have calculated timeout (TO) at the base station so that (BS) quickly sends exclusive handover message (EHM) to fixed host (FH) to avoid retransmission at FH. This can be easily achieved as MH gets router solicitation signal from new base station when mobile host does handover. Freeze TCP [2] works on the measurement and prediction of signal strength. However, with Freeze TCP, the time before which actual disconnection happens; in other words warning period; is quite a critical issue to be predicted. In fact, In proposed scheme, we have calculated timeout (TO) at the base station so that (BS) quickly sends exclusive handover message

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 254–261, 2011.

(EHM) to fixed host (FH) to avoid retransmission at FH. This can be easily achieved as MH gets router solicitation signal from new base station when mobile host does handover. Freeze TCP [2] works on the measurement and prediction of signal strength. However, with Freeze TCP, the time before which actual disconnection happens; in other words warning period; is quite a critical issue to be predicted. In fact, performance improvement of this scheme is totally dependent on accurate prediction of disconnection by the MH. In M-TCP [3], Base station waits for certain time for ACK and then assume that disconnection has occurred.

2 Related Work

A good amount of research is being done to improve TCP performance in the unpredictable mobile and wireless environments where link disconnections, packets losses and delays are common.

To handle the temporary disconnections caused by handoff fading or other reasons, most of the recent solutions employ the idea of putting sender into the persist mode M-TCP and Freeze-TCP both adopt this idea. Freeze-TCP uses similar approach of forcing sender into persist mode; but sending zero window size (ZWS) is prior to the real disconnection through signal strength measurements at the wireless antenna. Another method [4] is proposed to alleviate the performance degradation as a result of disconnections due to handoffs. In [5] if the sender receives an ACK with ERN (Explicit Handover Notification) indication, it resets the retransmission timer and adjusts its send window in response to the sequence number of this ACK. ATCP [6] assumes that network layer sends a connection event signal to TCP when MH gets connected to the network and a disconnection event signal when the MH gets disconnected from the network. In Proactive WTCP [7] MH monitors its receiving signal strength. A threshold of receiving signal strength is set to predict the impending disconnection. When the signal strength is lower than the threshold, MH predicts disconnection.

To enhance traditional TCP performance with handoff loss [8], they propose the concept of active-mobile-host, which maintains the original end-to-end semantics. They assume the MH has the knowledge of RTT (Round Trip Time), which may not be practical since RTT is often measured very coarsely by the sender instead of the MH itself: The idea of active-mobile-host is to let the MH actively advertise a ZWS ACK to the sender just at the time instant of crossing the boundary of Core Area Upon receiving ZWS, the sender will freeze all retransmission timers and enter a persist mode. But the sender keeps sending zero-window-probe packets to the MH until the MH's windows opens up.

3 Mobile IP

Mobile IP (Internet protocol) [9] is primarily concerned with the maintenance of mobility. It allows MH to move from home network to the foreign network. It solves the primary problem of routing IP packets to mobile hosts. Handover occurs when the mobile host moves from its present location to a new location. If it moves from home network to foreign network, MH must register its new location through registration request and registration reply.

4 Exclusive Handover Mechanism

There are only two assumptions in exclusive handover message. Firstly, a mobile host can immediately know that its handover has occurred. This knowledge can be acquired easily by receiving signal from new base station [5]. Secondly, *a* (equation 1) is considered in the range of 50msec to 100msec [5] because round trip time and it's variation between BS and MH is small, typically in the order of milliseconds as there is only one hop between base station and mobile host [9]. EHM extends TCP by using reserved bit of TCP header shown in Figure 1. It negotiates through setting first reserved bit of TCP header to one in the EHM ACK.

1	X	X	X	X	X

Fig. 1. Reserved bits of TCP header

In our approach, connection is split into two segments at BS, FH uses unmodified TCP while BS uses EHM concept. MH keeps records of last ACK it has sent to FH.

4.1 Calculation of Time Out at Base Station

BS maintains its own timer to determine the time, when to send EHM ACK to avoid unnecessary retransmission and going into congestion avoidance and slow start phase at FH, when MH does handover. BS sends EHM ACK to FH after time out, then no congestion control or timeout operations are performed by FH. The choice of when to send a EHM ACK to avoid performance degradation is very important.

Our approach uses timer-based method, where, the BS uses a timer granularity that must be smaller than the FH timer granularity, so that it can timeout before the FH time out. Smooth round trip time estimation used in the traditional TCP is not considered in this approach because as only one hop is involved in the base station-mobile host link. After handover MH generally receives Mobile IP router advertisements from the base station with the interval of 0.1sec. Therefore, the mean time at which a MH may detect its own movement is 50 msec [5]. We estimate TO at base station is two times round trip time (rtt) between base station and mobile host [3] and extra time a(range of 50 msec to 100msec [5]). From [3][5][11] estimated TO at base station is

$$\mathrm{TO}=2\mathrm{rtt}+ a \tag{1}$$

The complete operation of EHM scheme flow is shown in Figure 2.

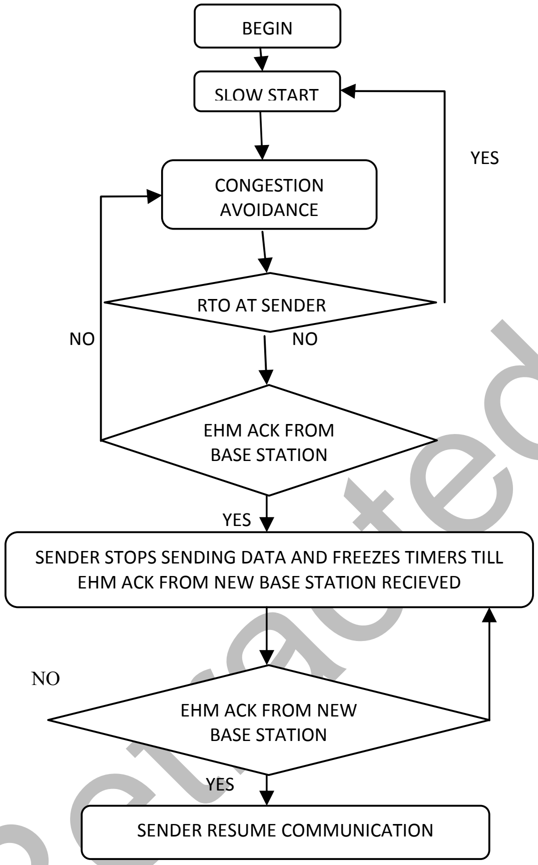

BEGIN
SLOW START
YES
CONGESTION AVOIDANCE
RTO AT SENDER
NO
NO
EHM ACK FROM BASE STATION
YES
SENDER STOPS SENDING DATA AND FREEZES TIMERS TILL EHM ACK FROM NEW BASE STATION RECIEVED
NO
EHM ACK FROM NEW BASE STATION
YES
SENDER RESUME COMMUNICATION

Fig. 2. Flow Chart of EHM scheme

5 Simulation Model

NS-2 network simulator is used to evaluate the performance of presented work and compared with TCP Tahoe and TCP Reno. Table 1 shows parameters including packet error rate (PER) used in the simulation. Figure 3 shows the network topology used in the simulation. FTP is used for transferring the data. MH and FH both use window option [1] to increase the window size. Assumption has been made that no losses on the wired link. We consider burst error model for errors on the wireless link. This error model is characterized by two state Markov Channels, where the channel is in the good state or bad state.

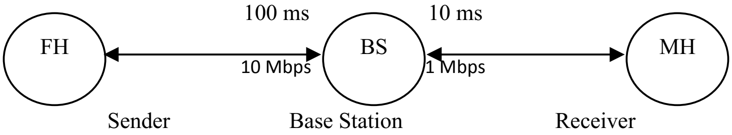


Fig. 3. Simulation Topology

Table 1. Parameters used in Simulations

Parameters	**Value**
Packet Size	1 KB
ACK Size	40 B
Size of File	1 MB
Wired Link	
Band Width	10 Mbps
Delay	100 msec
Wireless Link	
Band Width	1 Mbps
Delay	10 msec
Good State	
Good State Duration	1 sec
PER in Good State	$0.8*10^{-2}$
Bad State	
Bad State Duration	10 ms to 100 ms
PER in Bad State	$0.8*10^{2}$

Figure 4 shows an example of handover of MH from BS to BSnew' At first, MH attaches to BS. FH will send packets 1-6, MH receive packet 1-4 and returns ACK upto packet 4 only and BS will wait for ACK of packet 5 and 6 upto TO then BS assumes that MH moves from BS to BSnew' Now, BS will send EHM ACK to FH. When FH receives EHM ACK, it does not start congestion avoidance phase. As soon as the MH receives advertisement from BSnew, MH registers with BSnew and BSnew will send EHM ACK to FH. When FH receives this ACK it resumes communication with the same rate**.**

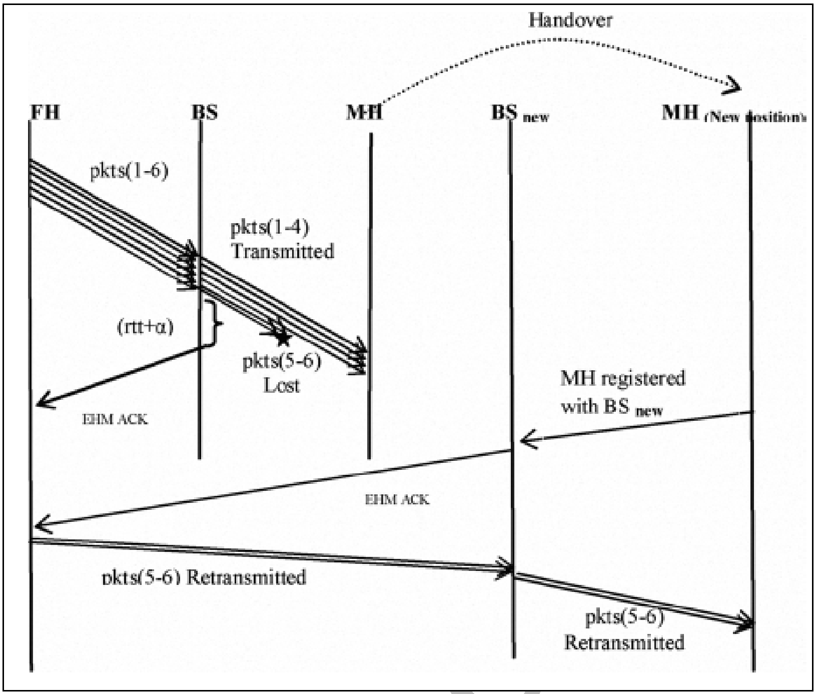


Fig. 4. Handover mechanism of Mobile Host

6 Result and Discussion

Figure 5 graphically illustrates the increased throughput, when EHM mechanism prevents sender side window from dropping and regrowing. Also, due to increase in the disconnection time, the proposed scheme makes throughput decline more gently than the rapid decrease as observed in TCP Tahoe and TCP Reno. The throughput of TCP Tahoe drops by 88.86% and in case of TCP Reno drops by 78.83% at the disconnection time of 100 msec in comparison with our scheme. EHM TCP achieves throughput values 4-8 times higher than TCP Tahoe and TCP Reno.

Table 2. Result Analysis Wireless Network (Delivery Ratio)

Protocol	No of node	Packet sent	Packet Dropped	Received
TCPNew Reno	2	41	4	39
TCPReno	2	41	6	35
EHM	2	41	1	40

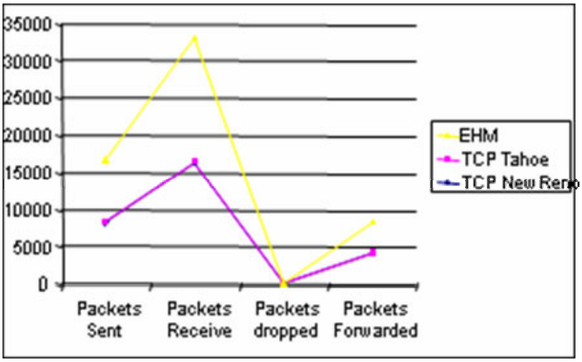


Fig. 5. Throughput comparison for TCP Tahoe, TCP Reno and Exclusive Handover Message scheme

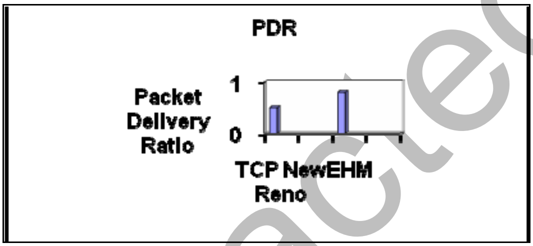


Fig. 6. Packet Delivery Ratio

7 Conclusion

The highlight of EHM scheme are: end-to-end semantics are maintained, it does not interfere with regular TCP algorithm, which runs over wired link and also, it does not require major modifications at FH and BS. The resulting performance of the TCP EHM scheme is greatly enhanced with compared to TCP Tahoe and TCP Reno. Though, we have used split connection still it supports encrypted traffic. Proposed scheme is a true end-to-end approach, based on the idea of exclusive handover message and is used for alleviating the degrading effect of host mobility on TCP performance. Factor (rtt+a) has been considered in place of arbitrary certain time at BS [3][9].

Our scheme does not need impending disconnection prediction like [2] when MH is being moved; as Freeze-TCP suffers from severe performance degradation when the prediction fails. In our approach BS is not required to forward the socket buffers to BS new and resulting in significant gain in throughput. Since sender is prevented from going into exponential back off and slow start.

There are certain modifications are required to handle burst error problems, which is not considered in proposed approach. In this paper, we have analyzed the performance of EHM, which is a thin layer between TCP and IP. We simulated the network and presented a comparison between the standard TCP and EHM in wireless network by 75%.

References

1. Xylomenos, G., PolyZos, G.C., Mahonen, P., Saaranen, M.: RCP performance Issues over wireless Links. IEEE Communications Magazine (April 2001)
2. Chadran, K., Raghunathan, S., Venkatesan, S., Prakash, R.: A feedback based scheme for improving TCP Performance in ad hoc wireless networks. In: 18th International Confrence on Distributed Computing Systems
3. Holland, G., Vaidya, N.: Analysis of TCP performance over ad hoc networks. In: ACM MOBICOM (2001)
4. Pahkwan, K., Krishnamurthy, P., Hatami, A., Ylianttila, M., Makela, J., Pichma, R., Vallastrom, J.: Handoff in hybrid mobile data networks. IEEE Personal Communication Magazine (April 2000)
5. Bakre, A., Badrinath, B.R.: I-TCP for Mobile Hosts. Tech Rep., Reuters University (May 1995)
6. Oliviera, R.D., Braun, T.: TCP in Wireless Mobile Ad Hoc Networks. Tutorial
7. Johnson, D.B., Maltz, D.A.: Dynamic source routing in ad hoc wireless networks. Mobile computing
8. Singh, R., Kumar, A., Tyagi, A.: Wireless Mobile Technology –Applications in Rural Areas. In: International Conference on Emerging India 2008 (2008)
9. Singh, R., Singh, Y.P., Tyagi, A.: Improving TCP performance over Mobile Single Hop wireless Networks. In: National Conference on IT Research & Application (2007)

Extended Biogeography Based Optimization for Natural Terrain Feature Classification from Satellite Remote Sensing Images

Sonakshi Gupta[1], Anuja Arora[1], V.K. Panchal[2], and Samiksha Goel[3]

[1] Jaypee Institute of Information Technology, Computer Science Department, A-10,
Noida-201 307, Uttar Pradesh, India
[2] Defence Terrain Research Lab, DRDO, Metcalfe House, New Delhi-110054, India
[3] Delhi University, Computer Science Department, New Delhi- 110006, India
sonakshigupta@live.com, anuja.arora@jiit.ac.in,
vkpans@gmail.com, samiksha.goel@gmail.com

Abstract. Remote sensing image classification in recent years has been a proliferating area of global research for obtaining geo-spatial information from satellite data. In Biogeography Based Optimization (BBO), knowledge sharing between candidate problem solutions or habitats depends on the migration mechanisms of the ecosystem. In this paper an extension to Biogeography Based-Optimization is proposed for image classification by incorporating the non-linear migration model into the evolutionary process. It is observed in recent literature that sinusoidal migration curves better represent the natural migration phenomenon as compared to the existing approach of using linear curves. The motivation of this paper is to apply this realistic migration model in BBO, from the domain of natural computing, for natural terrain features classification. The adopted approach calculates the migration rate using Rank-based fitness criteria. The results indicate that highly accurate land-cover features are extracted using the extended BBO technique.

Keywords: Remote Sensing, Image Classification, BBO, terrain.

1 Introduction

Classification is a prime area for image understanding. Image classification is an important tool for discerning different terrain features. Classification is obtained as an important result of remote sensing image processing. Many soft-computing techniques have been developed to solve the complex problem of classification such as Artificial Neural Network (ANN), Ant Colony Optimization (ACO), Rough set theory, Particle Swarm optimization (PSO), Fuzzy Sets and Swarm Intelligence (SI).Each of these techniques have positive and negative attributes. Biogeography Based Optimization (BBO) is a new concept devised under nature inspired techniques. BBO was introduced by Simon [1] in 2008 and has proved to be a successful method by performing efficiently on different benchmark functions. Biogeography Based Optimization has also been applied in practical cases such as ground water detection [2], power system optimization [3] and sensor selection [1].

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 262–269, 2011.

In the Biogeography Based Optimization approach information is shared by the immigration and emigration of species between habitats [4]. Another feature of BBO is that the original population is modified after every generation by migration rather than being discarded. Therefore, migration is the most important aspect of the process. In this paper a more practical migration model using Rank-based fitness for calculating the immigration and emigration rates is investigated to classify the image into different terrain features.

The organization of the paper, into six sections, is as follows: The paper is divided into six sections. Section 1 introduces the topic, *Section 2,* following the introduction, presents a general presentation of the new optimization technique method BBO. The *Section 3* presents the Extended Migration model of the BBO that is used in this study. The *Section 4* presents the methodology – the Datasets, characteristics of the region. Classification results of images of Alwar are discussed in *Section 5,* whereas *Section 6* concludes the study undertaken.

2 Biogeography Based Optimization

In BBO each individual solution is considered as a habitat with a Habit Suitability Index (HSI) [5]. The habitability of an island is represented by Suitability Index Variable (SIV). A habitat with a high HSI means that the habitat has a large number of species. On the other hand a habitat with low HSI has a smaller number of species. A good solution is analogous to an island with a high HSI and a poor solution indicates an island with a low HSI. High HSI solutions tend to share their features with low HSI solutions. Low HSI solutions accept a lot of new features from the ones with a relatively high HSI solution [1].

Migration

Habitats are modified in each generation by transporting the SIVs from one habitat to another in accordance with the immigration rate λ_i of the habitat H_i which is to be modified. The source habitat H_j is decided on the basis of the emigration rate μ_j.

The migration process is as follows:

```
Step 1:    Select $H_i$ with probability proportional to $\lambda_i$
           If $H_i$ is selected GOTO Step 2

Step 2:    Select $H_j$ with probability proportional to $\mu_j$
           If $H_j$ is selected GOTO Step 3
Step 3:    Randomly select SIV $s_j$ from $H_j$
           Replace a random SIV in $H_i$ with $s_j$
           GOTO Step 1
```

It is indicated in a study by Ma [6] that immigration rate is more influential on BBO performance than emigration rate. Therefore in case of a Universal Habitat from which SIVs need to be migrated to the candidate solution habitats, we can bypass Step 2 and go to Step 3 directly from Step 1. In this case, the island from which emigration will take place will be restricted to the Universal Habitat.

A linear migration model is followed in the original Biogeography Based Optimization technique proposed by Simon [1] wherein the migration curves used are depicted as straight lines. This linear migration model follows that habitats with large number of species have a high emigration rate and low immigration rate due to overcrowding of the habitat. Correspondingly, habitats with small number of species have high immigration rate and low emigration rate due to abundant room for species.

The weakness of this model is that it does not accommodate additional factors that influence migration rates, such as climate, human activities and size of habitat which are taken into account in the non-linear model proposed by Ma [6].

3 Extended Migration Model

In order to supplement the linear model, an extended migration model is used in the BBO technique to extract natural terrain features from the satellite image. This migration model adopts a non-linear approach which takes care of the many factors that must be considered to accurately depict make the actual migration process.

3.1 Non-linear Migration Model (Sinusoidal migration model)

This model considers the predator-prey relationships, evolution of species, population size and species mobility factors which make the migration curves sinusoidal [7]. Migration rates change slowly from the extremes when the number of species is small and they change rapidly from their equilibrium values when species count is medium.

The immigration rate is given as:

$$\lambda_i = I/2\ (\cos\ (k(i).\pi/n) + 1)\ . \tag{1}$$

The emigration rate is given as:

$$\mu_{i} = E/2\ (-\ \cos\ (k(i).\pi/n) + 1)\ . \tag{2}$$

Here I is the maximum immigration rate, E is the maximum emigration rate, k (i) is the fitness rank of the i^{th} individual and n is the total number solutions of the problem. The best rank that can be assigned is n and 1 is the worst rank. Based on equations (1) and (2) a good solution is characterized by high emigration rate and low immigration rate.

4 Methodology

This section describes the phases followed in the present study. Image Classification requires clustering to be done which is not an inherent part of the original BBO algorithm. Therefore, to extract terrain features from the image we have created clusters of the image data.

4.1 Dataset

A multi-resolution, multi-spectral, multi sensor image of the Alwar area in Rajasthan is used. The dimensions of the image are 472 X 576.Satellite images of seven different bands are taken which are- Red, Green, Near Infra-Red (NIR), Middle Infra-Red (MIR), Radarsat-1 (RS1), Radarsat-2 (RS2) and Digital Elevation Model (DEM). LISS (Linear Imaging Self Scanning Sensor)-III, sensor of Resourcesat, an Indian remote sensing satellite, is the source for the Red, Green, NIR and MIR band images. RS1 and RS2 are the images from Canadian satellite Radarsat. Digital elevation model (DEM) is derived by using images from RS1 and RS2. Fig.1 shows the 7-Band satellite image of Alwar area in Rajasthan.

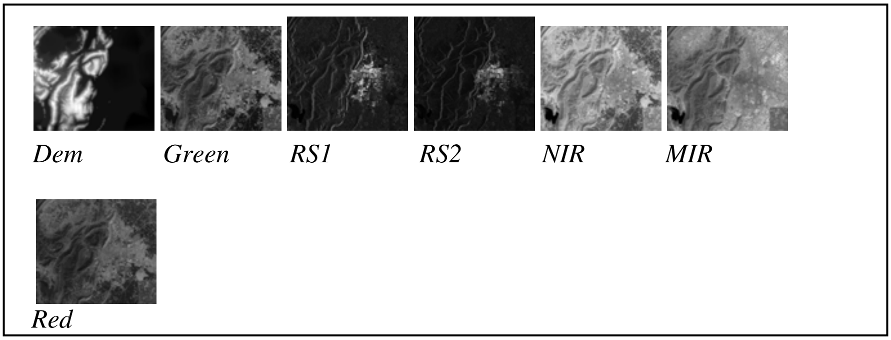


Fig. 1. Seven-band images of Alwar

4.2 Characteristics of Alwar Region

Alwar comprises of hills, plains, semi-arid and urban areas. Alwar is geographically situated at Latitude 27° 34' North and Longitude76° 35' East at an elevation of 270 meters above sea level. The city has vast stretches of dense deciduous forests that are rich in flora and fauna. Alwar area can be characterized as a Rocky area. Alwar has a greater proportion of Rocky and Vegetation area and a low proportion of water and urban area respectively.

4.3 Proposed Framework

Unsupervised clustering is performed on the image using the Fuzzy c-means theory by taking NIR and MIR bands as parameters. The resultant clusters are the species in the universal habitat. There are five feature habitats- rocky, vegetation, water, barren and urban to which the species from the universal habitat are migrated. Each feature habitat initially contains the training set pixels of that feature. The HSI or fitness of each solution is calculated as the mean of standard deviation of feature habitat. Each of the multispectral bands of the image contributes towards the Suitability Index Variable (SIV) of the habitat. Since the image in each band is a gray image, SIV Є X, where X is an integer and X Є [0,255]. A habitat- H Є SIV^m, where m is the number of spectral bands. The resultant clusters may represent a mixed population as well which is to be migrated from universal habitat to feature habitat.

4.4 Algorithm for Classification

Input –Satellite image
Output – Classified image

1. Categorize image into elementary classes using fuzzy c- means and consider them as species of the universal habitat. Consider each terrain feature as one habitat. *Total no. of habitats = universal+feature habitat*
2. Define HSI, Smax, Smin, maximum immigration rate and maximum emigration rate.
3. Calculate HSI for each feature habitat.
4. Select species from universal habitat and migrate it to one of the feature habitats and recalculate HSI.
5. Mean-value= Recalculated HSI – HSI calculated in step 3
6. Repeat steps 3 to 5 for each feature habitat
7. Rank the feature habitats in accordance with their Mean-value.
8. Calculate the immigration rate of each feature habitat.
9. Absorb the species in the habitat with lowest immigration rate.
10. If all species in universal habitat are checked then stop else go to step 3.

4.5 Parameters Considered

Consider the probability PS that the habitat contains exactly S species. PS changes from time t to (Δt + t) time as follows [7] :

$$P_S (t+\Delta t) = P_S (t) (1-\lambda_s\Delta t-\mu_s\Delta t) + P_{S-1} \lambda_{S-1} \Delta t + P_{S+1} \mu_{s+1} \Delta t . \quad (3)$$

where λ_s and μ_s are the immigration and emigration rates when there are S species in the habitat. This equation holds because in order to have S species at time (t+Δt), one of the following conditions must hold:

1) There were species S at time t, and no immigration or emigration occurred between t and (t+Δt);
2) There were species (S-1) at time t, and one species immigrated;
3) There were species (S+1) at time t, and one species emigrated.

For the immigration rate given by Eqn. 1, n=5 which is the number of feature habitats. The value of k ranges from 1which is the worst rank to 5 which is the best rank. The immigration rate curve for different ranks is illustrated in Fig.2.

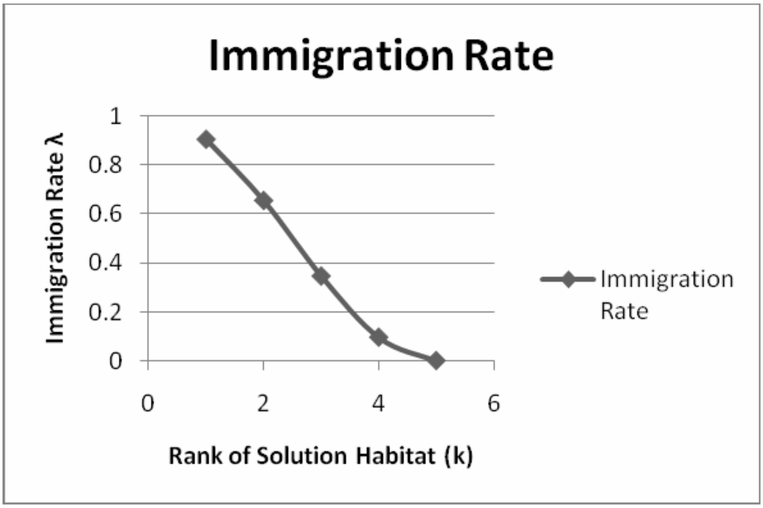


Fig. 2. Plot of the immigration rate with the rank of solution based on fitness

5 Results

The classified image of the Alwar region after applying the Extended Biogeography Based Optimization approach is shown in Fig.3 and the classified image after applying pure BBO [5] is shown in Fig.5. The terrain features- rocky, barren, water, vegetation and urban are represented by yellow, black, blue, green and red colors respectively.

The kappa coefficient, an index of assessing classification accuracy of the classifier, generally used in the remote sensing domain, is calculated using the method given by Lillesand and Kiefer [8]. The kappa coefficient is computed to explain proportional improvement of the classification over a random assignment of classes. The kappa coefficient for both the approaches is compared in Table 1. The Extended BBO has a higher kappa coefficient exhibiting better performance compared to the pure BBO approach. The results indicate that the water body feature is identified with maximum efficiency.

Table 1. Kappa Coefficients for Extended BBO and pure BBO approach

Extended BBO	pure BBO
0.6912	0.6715

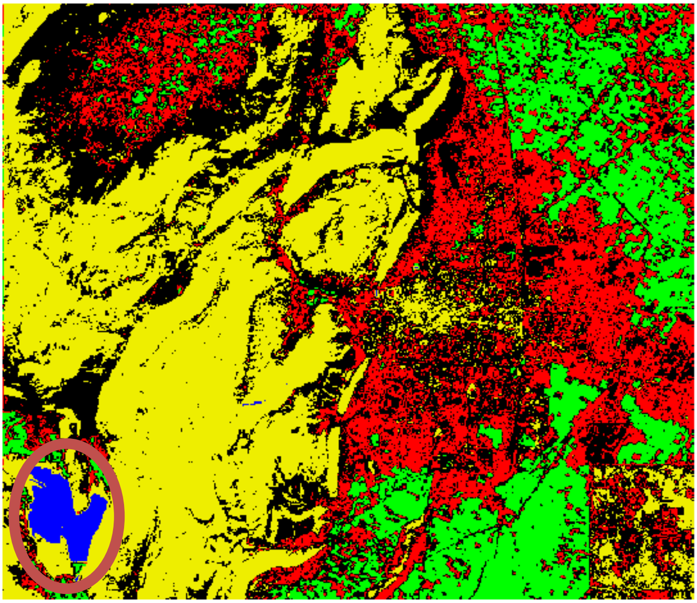


Fig. 3. Classified Image of Alwar using Extended BBO

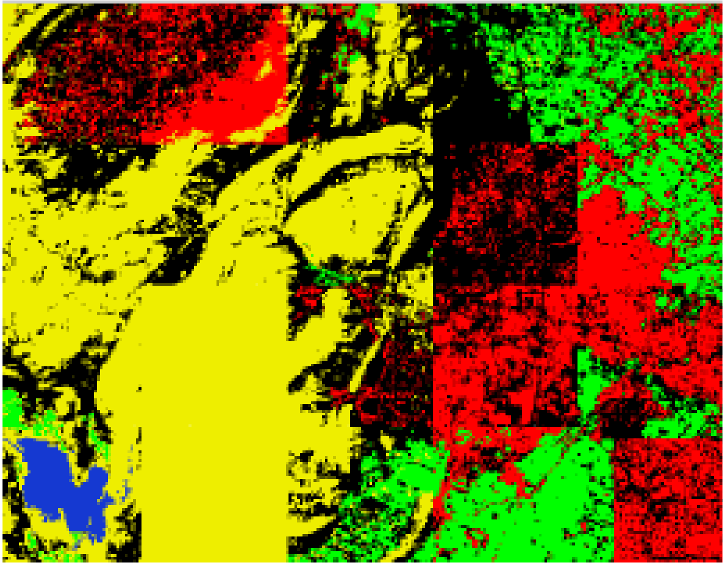


Fig. 4. Classified Image of Alwar using pure BBO

6 Conclusion and Future Work

As the results show, Extended Biogeography Based Optimization results in accurate classification of natural terrain features from satellite image. Rank-based fitness has proved to be sufficient criteria for migration of species between habitats. For Alwar region results show that water region is classified accurately and for regions like rocky, vegetation and barren it gives satisfactory result. But in case of urban region, accuracy is reduced. The reason for the Extended BBO not performing well for the urban region needs investigation. Also, the performance of Extended BBO can be enhanced by using better training sets.

References

1. Simon, D.: Biogeography-based optimization. IEEE Trans. on Evolutionary Computation 12(6), 702–713 (2008)
2. Kundra, H., Kaur, A., Panchal, V.K.: An Integrated Approach to Biogeography Based Optimization with Case Based Reasoning for Retrieving Groundwater Possibility. In: 8th Annual Asian Conference and Exhibition on Geospatial Information, Technology and Applications, Singapore (2009)
3. Rarick, R., Simon, D., Villaseca, F., Vyakaranam, B.: Biogeography-Based Optimization and the Solution of the Power Flow Problem. In: IEEE Conference on Systems, Man, and Cybernetics, pp. 1029–1034. IEEE, San Antonio (2009)
4. Ergezer, M., Simon, D., Du, D.: Oppositional biogeography-based optimization. In: IEEE Intl. Conf. on Systems, Man and Cybernetics, pp. 1009–1014 (2009)
5. Panchal, V.K., Goel, S., Bhatnagar, M.: Biogeography Based Land Cover Feature Extraction. In: World Congress on Nature and Biologically Inspired Computing, Coimbatore, India, pp. 1588–1591 (2009)
6. Ma, H.: An Analysis of the Equilibrium of Migration Models for Biogeography-Based Optimization. J. Information Sciences 180(18), 3444–3464 (2010)
7. Whittaker, R.: Island Biogeography.Oxford University Press, Oxford (1998)
8. Lillesand, T., Kiefer, R.W., Chipman, J.: Remote Sensing and Image Interpretation, 5th edn. Wiley India Pvt. Ltd, Chichester (2007)

DNA Based Molecular Electronics and Its Applications

Deep Kamal Kaur Randhawa[1], M.L. Singh[2], Inderpreet Kaur[3], and Lalit M. Bharadwaj[3]

[1] Department of Electronics & Communication Engineering, Guru Nanak Dev University Regional Campus, Jalandhar, India
[2] Department of Electronics Technology, Guru Nanak Dev University, Amritsar, India
[3] Biomolecular Electronics and Nanotechnology Division (BEND), Central Scientific Instruments Organization (CSIO), Sector-30C, Chandigarh, India
randhawadk@gmail.com

Abstract. Single Electron Transistor is the most prominent nanoelectronic device that will dominate the operations of nanoscaled integrated circuits. Molecules, especially DNA is prophesized to be integral part of the futuristic ICs. In this paper the current voltage characteristics of DNA base Cytosine are obtained by non-equilibrium Green's function combined with density functional theory. The pattern of current flow for an applied voltage sweep of 0-5 V is plotted. The phenomenon of tunneling is exhibited in the characteristics of molecules. The DNA base cytosine displays a typical surge of current in the voltage sweep section of 0.4V-0.6V, indicating single electron effects. The effect of gate voltage on the current-voltage characteristics of cytosine was studied in the gated two-probe setup. The typical section of characteristics of cytosine was re-drawn by varying the gate potential. The application of gate bias exhibits excellent ON/OFF switching for combinations of the two applied voltages- source voltage and gate voltage. Repetitive peaks are also observed in current when gate voltage is varied, fixing source potential. In this paper the cytosine molecule is proposed as a switch, AND gate and OR gate in this paper that can be used in DNA based molecular electronic devices.

Keywords: DNA based Molecular Electronics, Cytosine, tunneling, single electron effects, molecular gates.

1 Introduction

The continuous scaling down of electron devices to increase the speed of electronic systems that are supported by dense memories is pushing integrated circuits into the realm of nanoelectronics. For a nanoelectronic device, the element can be a carbon nanotube or a semiconducting nanowire. Molecules also provide an interesting option for use as an element in nanoelectronic devices. Molecular electronics tends to utilize the electronic properties of certain molecules as nature has been doing for millions of years. Since the molecules are very small, their functionality can be tuned. Development of molecular and nanoelectronic components such as wires, diodes, transistors, oscillators and switches [1],[2] as well as the conceptual discussion of

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 270–278, 2011.

their current-voltage properties have greatly enhanced the zeal for designing novel molecular systems with typical electronic properties. Several attempts have been made to theoretically explain the current-voltage characteristics of the molecular systems [3],[4] There are various candidates for molecular devices such as-organic polymers [5],[6] large bio-molecules [7],[8], nanotubes & fullerenes [9],[10]. DNA the backbone of life is is one of the promising candidates for molecular electronics. Self-replication property of DNA renders it as most suitable for creation of identical molecular electronic devices. A DNA chain consists of a long polymer composed of four subunits (nucleotide) containing the bases adenine (A), thymine (T), guanine (G) and cytosine (C) attached to the repetitive sugar-phosphate backbone almost like beads strung on a necklace. The four DNA bases form an interesting subject for use in single molecule electronics. The DNA bases exhibit excellent tunneling effects that suggest use of Adenine and Guanine as RTDs [11]. In this paper the current-voltage characteristics of DNA base Cytosine are the focus of interest.

2 Computational Method

The molecular structure of the DNA base molecule Cytosine was created using HYPERCEM 7 software. The structure of C were obtained was inclusive of backbone that was removed from the single strand structure and the open bond was terminated using a hydrogen atom. Virtual Nanolab software was used to perform calculations, which is an *ab initio* electronic structure program capable of simulating and modeling electrical properties of nanostructured systems coupled to semi-infinite electrodes. Non-equilibrium Green's functions (NEGF) and density functional theory (DFT) are combined in the software and the entire system was treated self-consistently under finite bias conditions [12].We used LDA-PZ, which is the local density approximation (LDA) with the Perdew-Zunger (PZ) parametrization [13] of the correlation energy of a homogeneous electron gas calculated by Ceperly-Alder [14]. In these *ab initio* electronic structure computations, we used the Double Zeta Polarization (DZP) basis set. Gold was used to construct thin electrodes as it is a practical choice and a promising monoatomic nanowire [15]. The current is calculated using Landauer's formula which expresses the conductance of a system at T=0 in terms of the quantum mechanical transmission coefficients [16].

$$\mathrm{I}= \int_{\mu_L}^{\mu_R} T\ (E, V_S) dE$$

where μ_L and μ_R are the left- and right-side metallic reservoirs electrochemical potentials and T (E, V_S) is the transmission probability for electrons incident at an energy E through a device under a potential bias VS. The Landauer equation based on the Green's function method relates the elastic conductance of a junction to the probability that an electron with energy E injected in one electrode will be transmitted to another electrode through a scattering region which in our case is the DNA bases. Using the above-mentioned procedure we calculated current-voltage (*I-V*) characteristics of the DNA base Cytosine. The effect of gate potential on the characteristics of DNA base Cytosine was also studied by enabling the gate and providing gate bias.

3 Current Flow through DNA Bases

To study the current voltage characteristics the DNA base is placed between two gold electrodes one as source and other as drain. The positive voltage V_S is applied at source fixing the drain potential V_D at 0V. This potential difference will maintain them at distinct potentials so that;

$$\mu_L - \mu_R = qV_S$$

thus giving rise to two different Fermi functions which are expressed as:

$$f_L(E) = f_0(E- \mu_L) = \{1+\exp[(E- \mu_L)/K_BT]\}^{-1}$$
$$f_R(E) = f_0(E- \mu_R) = \{1+\exp[(E- \mu_R)/K_BT]\}^{-1}$$

Each contact seeks to bring the channel into equilibrium with itself. The quest to achieve equilibrium causes the current to flow from source to drain.[17]

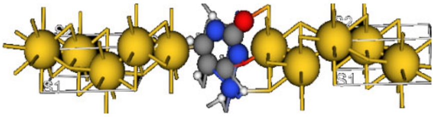


Fig. 1. DNA Base Cytosine inserted between gold terminals, Yellow balls are presenting Gold, Blue as nitrogen, Grey as Carbon and Red as Oxygen in electrode-molecule-electrode assembly

The molecular structure of DNA base created in HYPERCHEM 7 were inserted between two gold terminals using the two probe setup of Virtual Nanolab software forming metal-molecule-metal assembly as shown in Fig 1. The molecules are chemisorbed onto the electrodes, and the above orientation is fixed, although these molecules being asymmetric, the current-voltage characteristics will be varying greatly with orientation.

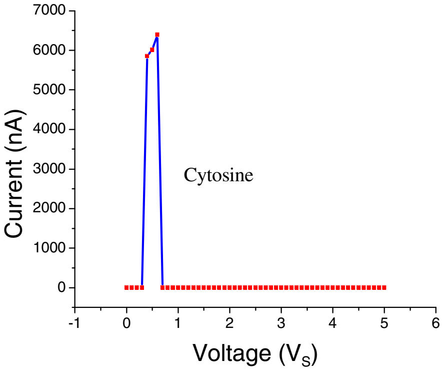


Fig. 2. Current-voltage plot for Cytosine

A voltage bias V_S varying from 0-5V was applied to the four respective bases and the corresponding current values were obtained. The current voltage characteristics for the Cytosine molecule are shown in Fig. 2. The curve illustrates flow of tunneling current through the molecule. The DNA base Cytosine shows a sudden surge of current in its characteristics. The value of peak current is 6.39 μA for an applied bias of 0.6 V. Though the magnitudes of current are very small, the current densities are going to be very large virtue the small size of molecules. These densities will be able to drive the electronic circuits. The typical surge of current motivated the study of effect of gate potential on the characteristics of the molecule.

4 Effect of Gate Potential on DNA Base Cytosine

The current voltage characteristics of the DNA base Cytosine display a very interesting pattern. When the voltage sweep of 0-5 V is applied at the source on gold-cytosine-gold assembly, a sudden surge of current is observed for applied source voltage of value 0.4-0.6 V. The typical pattern inspired the detailed probing of the characteristics of the molecule. To do so a third terminal gate was added to the two probe structure that is schematically represented in Fig.3 and is called Cytosine based Molecular Transistor.

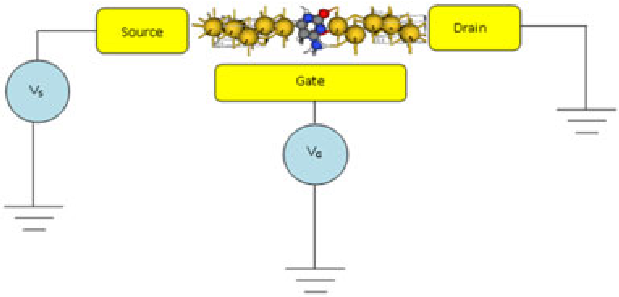


Fig. 3. Schematic representation of gated Gold-Cytosine-Gold structure

The current flowing between source and drain for a voltage sweep of -0.4V to 0.4 V was calculated while varying gate potential from -0.4V to 0.4V in incremental steps of 0.1V. From the values obtained, a set of current-voltage characteristics was plotted for Current I flowing between source and drain wrt voltage applied at source V_S; keeping gate potential V_G constant. Another set of current-voltage curves was plotted between current I and gate potential V_G maintaining constant source potential V_S. The analysis has been done on selective data that displayed typical values.

4.1 Current versus Source Voltage at V_G = constt

The set of characteristics between the source potential and the corresponding current were obtained for various gate potentials. Nine sets of current voltage values were obtained for gate potentials varying from -0.4V to +0.4V in steps of 0.1V. It is observed from the curves plotted that for V_G = 0V, the current suddenly rises from 4.84X 10^{-2} nA to 5.85 μA for change of source voltage from 0.3V to 0.4V displaying On/Off kind of behavior.

The typical behavior of sudden increase in conductivity of the molecule can be similarised to single electron effect. When the gate potential is applied, the energy levels of the molecule are shifted higher in case of negative gate potential and are lowered in case positive potential; thus altering the conductivity pattern. The current-voltage curve plotted for a source voltage sweep of -0.4 V to 0.4V with gate voltage V_G fixed at 0.2V is characteristically interesting. The curves are shown in Fig.4. It is seen that current changes from 1.62X 10^{-2} nA to 6.5 μA when V_S is increased from 0.1V to 0.2V and current stays in μA range for higher voltages of the sweep. This implies that the current goes high if the gate voltage V_G is varied from 0V to 0.2V fixing source at 0.2V.

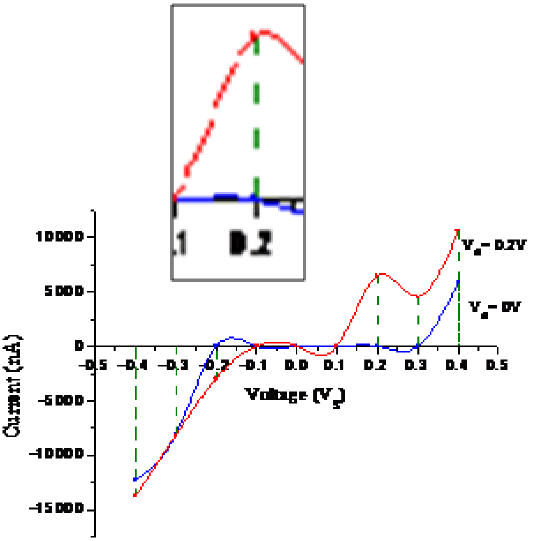


Fig. 4. Current v/s Source Voltage (V_S) for constt V_G = 0V and 0.2V

4.2 Current versus Gate Voltage at V_S =constt.

Another set of current-voltage characteristics as shown in Fig.5 was plotted between current and gate voltage for constant source voltage. The source voltage was varied from -0.4 to 0.4V in steps of 0.1V. Fixing the source at various source voltages, the current was obtained by sweeping gate voltage V_S from -0.4V to 0.4V. It is observed that the current flow through the assembly displays repetitive peak values when the gate voltage sweep is applied fixing the source potential at 0.2V and 0.3V. The pattern of current variation as shown in Fig. 5 can be related to the coulombic oscillations. This is due to resonant tunneling via molecular orbitals of cytosine that provide an open path for conduction whenever aligned with the Fermi level. For small source voltages of value ± 0.1V there is a constant flow of very small current of the 1.6X10^{-2} nA .

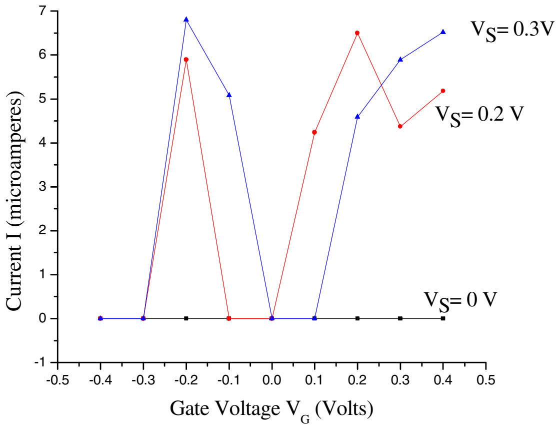


Fig. 5. Current v/s Gate Voltage (V_G) for constt V_S

The study of conductivity in cytosine molecule has created a very interesting pattern. The conductivity of the molecule is very high if we apply a gate potential of 0.2V and apply a source potential of 0.2 V or -0.2V, though the current flowing in the two cases is opposite in direction. Similar type of transition in conductivity is observed for fixed source voltage of 0.2V and gate potential varying from 0 to 0.2 V or 0 to -0.2V. The suppression of current around zero source voltage with zero gate potential is called coulomb blockade characteristics.

5 Molecular Gates

Logic gates are the backbone of digital electronics. They are building blocks of combinational and sequential circuits which can be used as counters, registers and memories. The logic gates can be designed using switches, relays, transistors etc. The switching behavior of the cytosine molecule can be used to design logic gates. In this paper we propose design for a switch which can be used in AND and OR gate using Cytosine based molecule transistor.

5.1 Cytosine Based Molecular Switch

The current-voltage data obtained for the cytosine molecule as shown in Table 1 exhibited excellent switching pattern.

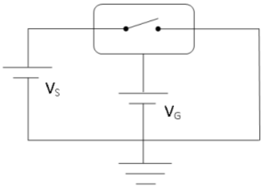


Fig. 6. Schematic of a switch

Table 1. Current flowing through gated gold-cytosine-gold assembly corresponding to various source and gate voltages

	V_G = -0.2 V	V_G =0 V	V_G = 0.2 V
V_S = -0.2 V	-3.239 e-011A (OFF)	-3.239e-011A (OFF)	-2.98e-006A (ON)
V_S = 0 V	0.0 (OFF)	0.0 (OFF)	0.0 (OFF)
V_S = 0.2 V	5.892e-006A (ON)	3.237e-011A (OFF)	6.501e-006A (ON)

It is observed that the switch will be always OFF (open) for V_G = 0V for all values V_S. Similarly it also remains OFF (open) for V_S = 0V for all values of V_G. A current in the range of 10^{-2} nA flows through the device if a voltage of 0.1V is applied to any of the terminals which is very small, hence can be approximated as zero and considered as logic LOW. The switch goes ON (closed) for voltage combinations as indicated in the Table 1, where the current is in μA range and can be considered as logic HIGH. So the switch can be operated in two ways; i) Fixing gate potential V_G at 0.2V and varying V_S in the Table, ii) Fixing the source potential V_S at 0.2 V and varying value of V_G as indicated in the Table 1.

5.2 Cytosine Based AND Gate

The AND gate is the "all or nothing gate" which can be expressed as a cascade of two switches as shown in Fig.7. The AND gate is a three terminal device with two inputs A and B and a single output labeled as Y. A and B inputs correspond to the source and gate terminal whereas output terminal Y is drain terminal. As the two switches are in series with the LED, it will light up when both the switches are closed. This means that for the LED to glow appropriate bias (0.2V) must be applied to the two

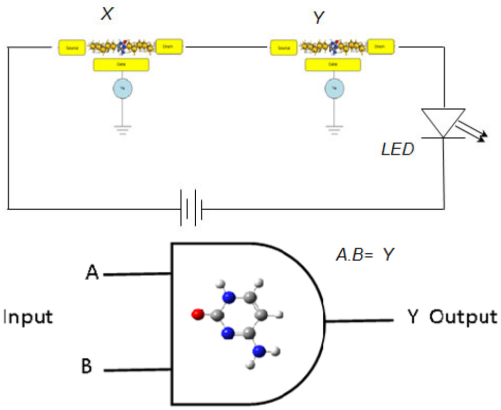


Fig. 7. Cytosine based AND Gate

switches through the battery. Further the gate bias of both the switches should be high (0.2V), so that right combination of two potentials take the cytosine based transistor into ON state and the LED glows which indicates completion of the circuit loop. The AND operator is denoted by a dot and the symbol of gate is as shown in the Fig. 7.

5.3 Cytosine Based OR Gate

The OR gate is also known as the "any or all gate". Fig.8 illustrates the basic concept of an OR gate using switches. As clear from the figure there are two paths for the current to complete the circuit. So the LED will low when either or both the switches are ON. As the voltage for source bias is same for both of the switches, the gate potentials will decide the status of switches, hence the glowing of LED. The OR operator is denoted by '+' sign between the two input variables and the symbol of OR gate is shown in Fig.8.

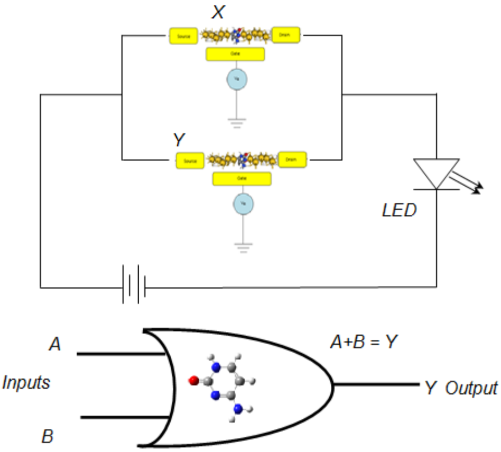


Fig. 8. Cytosine based OR Gate

6 Conclusions

The current- voltage characteristics have been plotted for the four DNA bases A, T, G and C in this paper using Virtual Nanolab. The comprehensive study of the characteristics displays flow of tunneling current through the four bases. The current flowing through purine DNA bases G and A is of nA range and is of μA range for pyrimidine DNA bases C & T. The typical surge of current in the i-v plot for cytosine implying single electron characteristics motivated detailed probing of the characteristics of the molecule by adding the third terminal 'gate' to the two probe structure of the gold-cytosine-gold assembly. The increase in value of current on application of gate potential confirmed the single electron effects in the molecule. Hence the three terminal assembly can be considered as a transistor. The suppression of current around zero source voltage with zero gate potential is called coulomb blockade. The detailed probing of gated current-voltage characteristics has shown that switching behavior which can be exploited to use the cytosine based transistor as a molecular switch. This switch is used to assemble the circuits acting as AND gate and OR gate. The cytosine based transistor can be used as a memory as you can control the ON/OFF pattern by applying adequate bias. Organic LED's can be used in the circuits to further improve the molecular circuits

References

1. Bauschlicher Jr., C.W., Lawson, J.W.: Current–voltage curves for molecular junctions: the issue of the basis set for the metal contacts. Phys. Rev. B. 75, 115406–115411 (2007)
2. Datta, S.: Electronic Transport in Mesoscopic Systems.Cambridge University Press, New York (1996)
3. Zhou, Y.-h., et al.: Current Rectification by asymmetric molecules: An ab initio study. J. Chem. Phys. 125, 244701–244705 (2006)
4. Di Ventra, M., et al.: First-Principles Calculation of Transport Properties of a Molecular Device. Phys. Rev. Lett. 84, 979–982 (2000)
5. Aviram, A., Ratner, M.A.: Molecular rectifiers. Chem. Phys. Lett. 29, 277–283 (1974)
6. Collier, C.P., et al.: A Catenane-Based Solid State Electronically Reconfigurable Switch. Science 289, 1172–1175 (2000)
7. Keren, K., et al.: Sequence-specific molecular lithography on single DNA molecules. Science 297, 72–75 (2002)
8. Porath, D., et al.: Direct measurement of electrical transport through dna molecules. Nature 403, 635–638 (2000)
9. Rinaldi, R., et al.: Transport in hybrid electronic devices based on a modified DNA nucleoside (deoxyguanosine). Annals of the New York Academy of Sciences 960, 184–192 (2002)
10. Benenson, Y., et al.: DNA molecule provides a computing machine with both data and fuel. Proc. Natl Acad. Sci. USA 100, 2191–2196 (2003)
11. Randhawa, D.K., et al.: Tunneling effects in DNA bases Adenine and Guanine. International Journal of Computer Applications (2011) (accepted)
12. Stokbro, K., et al.: TranSIESTA - A spice for molecular electronics. Ann. NY Acad. Sci. 1006, 212–226 (2003)
13. Perdew, J.P., Zunger, A.: Self-interaction correction to density-functional approximations for many-electron systems. Phys. Rev. B 23, 5048–5079 (1981)
14. Ceperley, D.M., Alder, B.J.: Ground State of the Electron Gas by a Stochastic Method. Phys. Rev. Lett. 45, 566–569 (1980)
15. Yong Xue, G., Mansoori, A.: Quantum Conductance and Electronic Properties of Lower Diamondoid Molecules and Derivatives. International Journal of Nanoscience 7, 63–72 (2008)
16. Ceperley, D.M., Alder, B.J.: Ground State of the Electron Gas by a Stochastic Method. Phys. Rev. Lett. 45, 566–569 (1980)
17. Datta, S.: Quantum Transport: Atom to Transistor. Cambridge University Press, Cambridge (2005)

An Efficient Fault Detection Algorithm in Wireless Sensor Network

Meenakshi Panda and P.M. Khilar

Department of Computer Science and Engineering,
National Institute of Technology Rourkela,
India-769008
{509cs106,pmkhilar}@nitrkl.ac.in

Abstract. In wireless sensor network (WSN) the accuracy of data is important to maintain the networks' performance. Therefore detecting nodes which either provide faulty readings or do not provide any information is an essential issue in sensor network management. As a whole the solution is to detect nodes with data and function faults, this paper proposes a novel method to detect nodes with both types of faults without assuming a particular sensing model. The performance of proposed fault model is more accurate and requires less communication compared to the existing methods.

Keywords: Wireless sensor network, data fault, function fault, detection algorithm.

1 Introduction

WSN is a self-organized network which consists of thousands of inexpensive low-powered devices, called sensor nodes. Each sensor node is capable of limited functionalities like computation, communication and sensing operations. These devices can be deployed for some specific applications like monitoring [1], detecting, and reporting of the occurrences of interesting events. The accuracy of individual nodes' data is crucial in these applications, e.g., in a surveillance network [8], the readings of sensor nodes must be accurate to avoid missed detections. Although some applications are designed to be fault tolerant up to some extent. The performance of the whole system significantly improved by removing nodes with fault readings from the system with some redundancy or replacing them with fault free sensor nodes. To conduct such type of maintenance extra deployment is required which is difficult to perform. Therefore it is essential to investigate an efficient methods for detecting faulty nodes in the network.

Wireless sensor network as a whole coming across an unexpected situation like the sensor nodes eventually misbehave, producing results not expected from the unit's specification, not getting result from some specific nodes etc [11]. This situation may occur due to the number of reasons, including typical hardware and software failures, malicious interference, battery depletion and natural calamities [2]. It is thus desirable to detect, locate the faulty sensor nodes and then exclude

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 279–288, 2011.

or replace them from the network for better Quality of Service (QoS) of the entire WSN. Thus fault tolerance is the major constraining factor on the functionalities of WSN. In this paper we put emphasize on fault detection to maintain the network quality.

Broadly there are two types of faults that can affect the performance of sensor network such as hard fault (permanent fault or function fault) and soft fault (transient fault, byzantine fault and intermittent fault)[3], [9], [4], [5], [11], [12]. In case of function fault, the faulty sensor nodes neither respond nor send any data to other nodes. But in transient fault, the sensor nodes can not perform their desired operation for a short duration and generally difficult for diagnosis. Where as the intermittent faulty sensor nodes [7] provide sometimes fault free information and thereby make the fusion center confuse to take a decision about the status of a sensor node. Byzantine faulty sensor nodes can behave in an arbitrary ways i.e. sends different data at different moments.

In literature [2] the authors have proposed a fault detection algorithm for WSN which is a sequence based soft fault detection method. In this approach, the entire terrine is partitioned into number of sub regions and each contains single node within itself. Each subregion is uniquely identified with an identifier which is calculated based on their distance to rest of the nodes present in the network. When an event occur each node sense it and then send their sensed data to the sink node. The sink node estimate a sequence based on the signals it receives from all its sensors, compared with corresponding stored sequences for some analysis to identify the faulty node. The major demerit of this approach is it requires a large memory at the fusion center to keep all identifiers of the sub regions. Along with this the approach needs more time for partitioning the entire terrine into no of sub regions which is in the order of $O(n^4)$.

In this paper a novel fault detection algorithm is proposed for detecting the soft and hard faulty nodes present in the wireless sensor network. The proposed method use less number of communications among the nodes and fusion center for the detection. Therefore the proposed centralized algorithm is more efficient compared to other existing algorithms [2].

The remaining part of the paper is organized as follows. In section 2, 3, 4 we presented the network model, fault model, detection model respectively required for developing the proposed fault detection algorithm. The proposed fault detection algorithm is described in section 5. The performance of the algorithm in terms of detection accuracy and false alarm rate has been described in section 6. Finally concludes the paper in Section 7.

2 Network Model

Let us assume that N number of sensor nodes are randomly deployed in an rectangular terrine located in two dimensional Euclidean Plane R^2. Each sensor node has an initial power source, a processing unit, memory, radio unit and sensors. The sensor nodes communicate wirelessly and employ one-to-many broadcast primitive in their basic transmission mode. The transmission range of each sensor node is fixed. The average degree of the network depends on the transmission range. The data sensed by the sensor node is stored locally on it's memory and sends their sensed data to their neighbors as well as to the fusion center regularly within a fixed interval of time. An algorithm is developed to construct an arbitrary network topology based on transmission range which is given in Algorithm-1. The notations used for the development of the algorithm-1 are given in table-1.

Table 1. The notations used for developing an arbitrary network topology

Symbol	**Meaning**
S	Set of sensor nodes in the sensor network.
N	Total number of sensor nodes deployed in rectangular terrine
$T_r(s_i)$	Transmission range of the sensor node $s_i \in S$
xco_i	x co-ordinate of $s_i \in S$
yco_i	y co-ordinate of $s_i \in S$
$E_d(s_i, s_j)$	Euclidean distance between the node s_i and s_j
$T_r(s_i)$	Fixed transmission power assigned to s_i

Algorithm 1. The algorithm for constructing an arbitrary network topology

```
Data: Size of the rectangular terrine
Transmission range of the sensor node $(x)$
Result: Arbitrary network topology
1. Compute a random position $p_i\,(xco_i, yco_i)$ such that $0 \leq xco_i \leq r$,
$0 \leq yco_i \leq r$.
2.Deploy the sensor node $s_i$ at $p_i$.
3.Assign the transmission range $x$ to $s_i$ i.e. $T_r(s_i) = x$.
4.Compute the communication link among the sensor nodes.
for $i = 1 \cdots N$ do
    for $j = 1 \cdots N$ do
        if $i \neq j$ then
            Compute $E_d\,(s_i, s_j)$
            if $E_d\,(s_i, s_j) \leq T_r\,(s_i)$ then
                $s_i$ is connected with $s_j$
            end
        end
    end
end
```

3 Fault Model

Generally fault occurs at different levels of the WSN, such as in physical layer, hardware system, system software and middle ware. As sensors are most prone to malfunction, so we focus on the sensor fault by assuming all software is already fault tolerant. That means the nodes are still able to communicate and process when their sensors are faulty. Faulty sensors may sense the environment or may not. When they sense the environment they send some data to its neighbors with zero mean and high variance, otherwise they send fixed data (zero or maximum value supported by sensor, any arbitrary value etc.) regularly to their neighbors. The proposed algorithm can handle all types of faults described above. In this scheme each sensor node receives data from its fusion center for diagnosing. After receiving the data they perform mean operation over the received data and then send the data back to fusion center. The fusion center analyses all the processed data again to decide which node is likely faulty nodes. All the likely faulty nodes work actively to decide the faulty nodes.

4 Detection Model

A diagnosis system has modeled as collection of independent sensors, denoted by S = $\{s_1, s_2, s_3, \cdots, s_n\}$ and these sensors communicate with each other via a communication medium. The interconnection model can be represented by an undirected graph $G(S, C)$ (without self loops and multi edges) where each $s_i \subset S$ represents a sensor and each undirected edge $c_{ij} \subset C$ represents undirected (possibly bidirectional) communication link between s_i and s_j. Two sensors interact with one another by sending messages over the communication link [6].

4.1 Assumptions

For the development of proposed algorithm the following assumption are taken into consideration.

- During fault diagnosis the sensor nodes are static.
- The network topology remains unchanged.
- Sensor node can send its sensed data to its neighbors and fusion center.
- synchronous mode of communication is used to send the data by all the sensor nodes to the fusion centre within a fixed interval of time.
- A faulty sensor node may receive accurate data from its neighbors as well as from fusion center but it may send arbitrary values as its functional unit does not work perfectly.
- The message forwarding or relay logic in each sensor node is fault-free but the processing logic of the sensor may be faulty.

5 Fault Detection Algorithm

This section provides details of the proposed fault detection algorithm. The entire algorithm is explained through i) initialization ii) computation iii) detection iv) confirm phases. All the notations used for developing the proposed fault detection model is summarized in table-2.

Table 2. The notations used for developing the algorithm

Symbol	**Meaning**
s_i	A sensor node deployed at $P_i(xco_i, yco_i)$
N	Total number of sensor nodes deployed in rectangular terrine of size $r \times r$ respectively
NT_i^t	Neighboring table of the sensor node s_i at the time instant t containing all the information about its neighbors and itself.
FS_i^t	Fault status of the sensor node s_i calculated by s_i at the time instant t
Neg_i^t	A set containing all the neighboring sensor nodes of s_i at the time instant t
LFS	Likely faulty sensors
AED_i^t	Average estimated data at s_i
RDB	A set containing the receive data at fusion center
DSB	Data send by the fusion center to all its sensor nodes coming across its transmission range
$Degree(s_i)$	Degree of the sensor node $s_i \in S$
GS_i	i^{th} good sensors, $1 \leq i \leq N$
RD_i^t	Received data of i^{th} sensor
$CRDN_i^t$	Cumulative sum of receive data of all the neighbors. This parameter is estimated by i^{th} sensor ,1 $\leq i \leq N$

5.1 Description of the Proposed Algorithm

It is assumed that the fusion center has detail description about the network topology at time t. After receiving information about the mobility of any sensor node s_i, the fusion center re estimate the topology of the network which will be required during fault diagnosis period.

Initialization Phase. In this phase of diagnosis, the fusion center broadcast a fixed data to all the sensor nodes $s_i \in S$ present in the sensor network. Each sensor node s_i is assumed to be fault free.

Computation Phase. Each sensor node s_i collects data from its neighbors' sensor nodes. It performs the mean operation over the collected data and its own received data at time t. The computted mean is send back to the fusion center.

Detection Phase. In this phase, the fusion center collects the calculated mean from each of the sensor nodes. After this it analyze the collected data to estimate the function and data faulty sensor nodes.

The fusion center extracts the sender Identifier from received data in order to identify which sensor nodes are able to send their data to fusion center. Those sensor nodes which are unable to send data to the fusion center are identified as function faulty sensor nodes. After identifying function faulty sensor nodes, the data faulty sensor nodes are diagnosed. For identifying the data fault sensor nodes, the fusion center compares the sensor node s_i's data d_i with its own data. If it is matched then fusion center concludes that all the neighboring sensor nodes of s_i including itself are fault free as d_i is calculated from the mean over the neighboring sensors data. Finally it finds some sensor nodes whose status can not be detected based on the above principle which are tagged under the likely faulty sensor nodes. The fusion center assign a task to all likely faulty sensor nodes for further diagnosis.

Confirm Phase. Each likely faulty sensor nodes perform the task assigned by fusion center to identify their own status. After completing the assigned task, some likely faulty nodes are detected as fault free. Then fault free nodes broadcasts their status over the sensor network. Now the fusion center has complete knowledge about the fault status of all the sensor node present in the network. Now the fusion center broadcast the information about the faulty node to all fault free node so that the fault free node neither send data nor receive data from the faulty node. By this the quality of the network can be maintained.

The detail of the algorithm is given in Algorithm-2. The notations used for developing the algorithm are summarized in table-2.

5.2 Time and Message Complexity of the Algorithm

The time and message complexity are the valuable parameter for evaluating an Algorithm. These parameters are evaluated by analyzing all the phases of the proposed algorithm. Initialization phase takes $O(1)$ time for initiating the fault detection algorithm. The computation and confirm phase takes $O(d)$ time as these phases are running by each of the sensor nodes over their neighbors data. In detection phase fusion center checks each of the sensor node's data to identify the good and likely faulty nodes which requires $O(N)$ time. So the total time required by the algorithm is $\max\{O(1), O(d), O(N), O(d)\} \approx O(N)$ which is much less than $O(N^4)$ reported in [2].

Similarly the message complexity of the algorithm is calculated by considering total number of messages exchanged by different phases over the network. In Initialization phase $N-1$ messages are exchanged over the network to keep the track of number of nodes are alive and functioning properly. Each sensor node collects data from their neighbors for which $(N-1)d$ messages are exchanged over the network where d is the average degree of the network. After collecting and computing the data each sensor node send their data to fusion center and needs $N-1$

Algorithm 2. Centralized Fault Detection in WSN

```
**Data**: Sensor node position $s_i^t\left(xco_i^t, yco_i^t\right)$
Transmission Range $(T_r)$
**Result**: Fault Status of Sensor node $s_i\left(FS_i^t\right)$
**Initialization Phase**
Fusion center sends the fixed data $x$ to all the sensors present on the network.
Each sensor node $s_i \in S$ receives data $R_i$ from their neighboring sensor node $Neg_i^t$.
Fault status of each sensor node is assigned to be fault free.
**Computation Phase**
Set $CRDN_i = RD_i$
for $j = 1 \cdots |Neg_i^t| and s_j \in Neg_i^t$ do
    Calculate $CRDN_i = CRDN_i + RD_j$
end
$AED_i = CRDN_i/(Degree(s_i) + 1)$
$s_i$ sends $AED_i$ to fusion center for further investigation.
**Detection Phase**
(a) Fusion center receives the data and store it on $RDB$ where $RDB_i$ keeps the $s_i$'s data, $s_i \in S$
(b) It checks the sender id to identify the function faulty sensor nodes.
(c)Here it identifies data fault sensor nodes
for $i = 1 \cdots N$ do
    if $RDB_i = DSB$ then
        Then $s_i$ and $s_j \in Neg_i^t$ is assigned with a status Good and add these sensors into the set $GS$
    else
        $s_i$ is added to the set $LFS$
    end
end
(d) Fusion center will send a query to each $s_i \in LFS$ for further processing.
**Confirm Phase**
for $j = 1 \cdots |Neg_i^t|$ do
    if $RD_i \neq RD_j$ then
        Add it to $FS_i$
    else
        Add it to $GS_i$
    end
end
Set $CRDN_i^t = RD_i^t$
for $j = 1 \cdots |GS_i|$ do
    Calculate $CRDN_i = CRDN_i + RD_j$
end
$AED_i = CRDN_i/(|GS_i|) + 1)$
if $AED_i = RD_i^t$ then
    $s_i$ broadcasts its status GOOD to fusion center.
else
    $s_i$ broadcasts its status as FAULTY to fusion center.
end
```

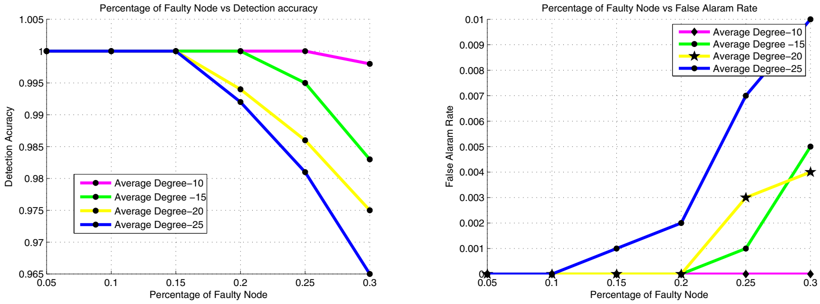
message exchange. After receiving data from different sensor node the fusion center identifies the hard faulty and likely soft faulty nodes. Then fusion center sends its query to likely faulty node to identify the soft faulty nodes available on the network which needs maximum of $N/2$ message exchange over the network. so total message exchange over the network is $(N-1)(1+d+1)+N/2 \approx O(Nd)$ respectively where N is the total number of nodes deployed in the target terrine.

6 Performance Analysis

The proposed fault detection algorithm is validated using MATLAB. The performance of the algorithm depends on number of sensor nodes deployed in a target area, the average degree of node, the probability that a sensor node is faulty. During simulation, we assumed that faults are independent of each other. The detection accuracy and false alarm rate [2] are being used for evaluating the performance of the algorithm. These parameters are defined below:

- Detection Accuracy (DA) is defined as the ratio of number of faulty sensor detected as faulty to the total no of faulty sensors introduced to the network.
- False Alarm Rate (FAR) is defined as ratio of the number of Non faulty sensor nodes diagnosed as faulty to the total number of Non faulty nodes present on the given network.

In our simulation 1024 sensor nodes are randomly deployed using normal distribution in a rectangular terrine of size 100 × 100 respectively. It is assumed that all the nodes have a equal transmission range. This transmission range is chosen for the sensor network to meet desired average degree of the network. The performance of the algorithm is evaluated for different percentage of faulty node. The percentages of faulty sensor nodes are introduced here are 0.05, 0.10, 0.15, 0.20, 0.25, 0.30, respectively.



(a) Percentage of faulty nodes Vs. detection accuracy for 1024 sensor nodes

(b) Percentage of faulty nodes Vs. false alarm rate for 1024 sensor nodes

Fig. 1. Performance analysis of fault detection algorithm

The performance of the proposed algorithm in terms of DA and FAR for different percentages of faulty nodes present in the network are deprecated in figure-1(a) and figure-1(b). The detection accuracy and false alarm rate are 100% when average degree of the network is 10 or less . As the average degree increases DA decreases and FAR increases which indicate that some of the faulty node detected as non faulty and some of the non faulty node detected as faulty respectively. In the worst case this algorithm can detect up to 97% of faulty.

The comparison result of the proposed algorithm with existing algorithm [2] in literature will be provided in the final paper.

7 Conclusion

In this paper a centralized co-operative based hybrid fault detection algorithm is proposed to detect hard and soft faulty nodes in wireless sensor networks. In this algorithm co-operative learning technique is used for identifying the data faulty nodes. Here the fusion center plays major role for identifying both function and data faulty sensor nodes. The overall communication overhead is reduced here as the likely faulty nodes detected by the fusion center are taking part for finding the actual faulty nodes. By doing this at least 50% of nodes are not taking part for the diagnosis which yeilds an efficient fault detection algorithm.

References

1. Szewczyk, R., Mainwaring, A., Anderson, J., Culler, D.: An Analysis of a Large Scale Habit Monitoring Application. In: Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems, SenSys 2004, pp. 214–226. ACM, New York (2004)
2. Guo, S., Zhong, Z., He, T.: FIND: Faulty Node Detection for Wireless Sensor networks. In: SenSys 2009, pp. 1–14. ACM, CA (2009)
3. Chessa, S., Santi, P.: Comparison Based System-Level Fault diagnosis in Ad-hoc Networks. In: Proceedings of the 20th IEEE Symposium on Reliable Distributed Systems, OCT (2001)
4. Ding, M., Chen, D., Xing, K., Cheng, X.: Localized fault-tolerant event boundary detection in sensor networks. In: IEEE Infocom, pp. 902–913 (2005)
5. Rangarajan, S., Fussell, D.: A Probabillistic Method for Fault Diagnosis of Multiprocessor Systems. In: Proceedings of the 18th International Symposium on Fault Tolerant Computing, pp. 278–283, Tokyo, Japan (1988)
6. Bond, J., Pyerat, C.: Diameter vulnerability in networks, Graph Theory with Applications to Algorithms and Computer Science, pp. 125–149. John Wiley and Sons, New York (1985)
7. Mallela, S., Masson, G.M.: Diagnosable Systems for Intermittent Faults. IEEE Trans. Comp. C-27(6), 560–566 (1978)
8. He, T., et al.: An Integrated Sensor Network System for Energy-Efficient Surveillance. ACM Trans. on Sensor Networks 2(1), 1–38 (2006)
9. Chessa, S., Santi, P.: Crash Fault Identification for wireless sensor networks, Jour. of Computer Communications 25(14), 1273–1282 (2002)

10. Lee, M.H., Choi, Y.: Fault detection of wireless sensor networks. Science direct Transaction on Computer Communications 31, 3469–3475 (2008)
11. Preparata, F.P., Metze, G., Chien, R.T.: On the Connection Assignment Problem of Diagnosable Systems. IEEE Trans. on Computers EC-16, 848–854 (1967)
12. Barborak, M., Malek, M., Dahbura, A.T.: The Consensus Problem in Fault-Tolerant Computing ACM computing surveys, vol. 25, pp. 171–220 (1993)

Modeling a Central Pattern Generator to Generate the Biped Locomotion of a Bipedal Robot Using Rayleigh Oscillators

Soumik Mondal, Anup Nandy, Chandrapal Verma, Shashwat Shukla, Neera Saxena,
Pavan Chakraborty, and G.C. Nandi

Robotics & AI Lab, Indian Institute of Information Technology, Allahabad, India
{mondal.soumik,nandy.anup,cverma.ro,shashwatshukla10aug,
neera.saxena}@gmail.com,
{pavan,gcnandi}@iiita.ac.in

Abstract. This paper mainly deals with designing a biological controller for biped robot to generate biped locomotion inspired from human gait oscillation. The nonlinear dynamics of the biological controller is modeled by designing a Central Pattern Generator (CPG) which is the coupling of the Relaxation Oscillators. In this work the CPG consists of four Two-Way coupled Rayleigh Oscillators. The four major leg joints (e.g. two knee joints and two hip joints) are being considered for this modeling. The CPG parameters are optimized using Genetic Algorithm (GA) to match an actual human locomotion captured by the Intelligent Gait Oscillation Detector (IGOD) biometric device. The Limit Cycle behavior and the dynamic analysis on the biped robot have been successfully simulated on Spring Flamingo robot in YOBOTICS environment.

Keywords: Rayleigh Oscillator, Central Pattern Generator, Intelligent Gait Oscillation Detector, Genetic Algorithm, Nonlinear Dynamics, YOBOTICS.

1 Introduction and Our Contribution

Humans started surviving on this beautiful planet since long decade. Then the invention of rock, wheel, fire vehicles etc. has been done by humans. And then a tremendous invention was done that was digital computer. To implement human thoughts, the added technology is invented. Then humans updated these technologies as per their need. Then they made some industrial robots, which can do the limited task. These types of robots are pre programmed robots. Then the technology took a new turn. So it has taken steps in the field of humanoid robotics. The humanoid robots are the robots which look and act like a human. These humanoid robots can adopt all the activities of human. It can perform the activities like walking, handshaking, running etc. The humans have much need of household robots, which can perform household task. The humanoid robot can act like a soldier on war. The most basic activity of humanoid robot is walking and performing this task in complex environments also. So make the robots walking pattern as efficient as humans is the first need of industry. After that it can easily perform the other task. This challenge to

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 289–300, 2011.

make humanoid robot's walking pattern as efficient as humans created much interest in me to work with it.

The main objective of this work is *"To build a CPG model by using Rayleigh Oscillators and train this CPG by human gait oscillation to generate the human like biped locomotion for biped robot."*

***Contribution made by this paper*:**

- ✓ Considered only four major joints for two legs in our work i.e. left hip, knee and Right hip, Knee.
- ✓ Two-Way coupling between four different Rayleigh Oscillators for four joints to design our CPG model.
- ✓ Capture the stable human walking data by a self made biometric suit called IGOD [1] and train the CPG model with that captured data.
- ✓ Find the optimized coupling parameters of CPG by using Genetic Algorithm.
- ✓ Simulate the generated human like biped locomotion by our designed CPG model into Spring Flamingo robot in YOBOTICS environment.

2 Related Work

The basic concept of Central Pattern Generator (CPG) is that a number of living species produces cyclic motor patterns. There is some sort of pattern generating systems or neural circuits are indicated that are able to generate cyclic movements [9][10][11]. Now as per biomechanical concept the CPG refers to a group made by the artificial neurons and these artificial neurons are oscillators, which are capable to produce an oscillatory signal output without any external periodic input. This concept of artificial neural network which is based on central pattern generator has been used in the field of human gait biomechanics and as well as in robotics [11].

In the robotics society, we are progressively using the C.P.G. models. The different views of CPG models are designed for robots including connectionist models (e.g. Lu, Ma, Li; Arena, 2000, & Wang, 2005), and some models created by coupled oscillators (e.g. Ijspeert et al.; Kimura et al.; Williamson et al.;) [16][17][18][19][20][21][22]. In some infrequent cases, some spiking neural models are used (e.g. Lewis et al.) [23]. Almost all implementations consist of some sets of Coupled Differential Equations which are integrated numerically on the processor or on a microcontroller. Most likely the only exceptions are CPGs. These CPGs are unswervingly realized in hardware, which is on a chip (e.g. Schimmel et al. 1997, DeWeerth et al.) [27] or with the analog electronics (Still & Tilden, 1998). It is associated to CPG research up to some scope which are quasi-cyclic movements governed by chaotic maps.

The CPG models have been widely used in the control of a variety of distinct robots and also in control of different modes of locomotion. The CPG models have already been used for hexapod and octopod robots. This has been inspired by pest locomotion like Arena, Frasca, etc.

Practical implementation of CPG in knee active prosthetic limb development was proposed by G. C. Nandi et al. [12][13]. Some CPG model simulation in Matlab was done by M. H. Kassim et al. and A. Carlos De Filho [14][15]. Behavior control of robot using Nonlinear Dynamics was proposed by Nakamura et al. [24][25][26].

3 Explanation of Relevant Components Participated in This Paper

3.1 Biped Locomotion

Biped locomotion is defined as the activities which are performed on two legs like walking, running and standing etc. Static stability upon two legs is extremely simple but at the same way maintaining the stability dynamically on two legs tends to be a critical task. It looks to be too simple but it implies an extremely nonlinear dynamical process. Fig. 1 describes the different phases of biped locomotion.

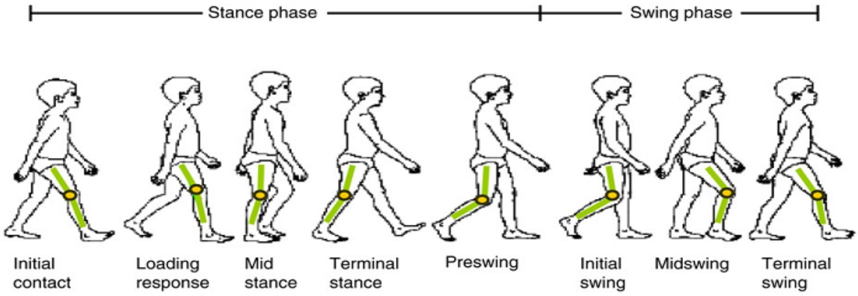


Fig. 1. Details of biped locomotion

3.2 Central Pattern Generator (CPG)

The concept of Central Pattern Generator is inherited from nature [3]. In this approach it is not mandatory to know the entire information about the robot dynamics. This method implies more adaptive to generate controllers for two leg walking. In this method there are some type of reflexes which are used to control the balance and the effect generated by the external force. These reflexes can also be used as the feedback for the system [2]. The designing of CPG based model is inherently based upon the concept of nonlinear dynamics which is being used for coupling of the relaxation oscillators. We have only considered four major leg joints (i.e. two hip joints and two knee joints) of a bipedal robot to generate the biped locomotion. The oscillators are coupled in the concept of Two-Way coupling technique. The completion of coupling technique is followed by the CPG which is able to produce the pattern of biped locomotion provided the proper selection of the different parameters.

The CPG are oscillator based controller. So the theory of limit cycle is used and this is very well-situated for the bipedal walking phenomenon. These oscillators can regenerate the stability against some weak external input. These can persist also in the stable state on the small disturbance in the preliminary circumstances. This method can be of two types, the open loop and the closed loop method.

The concept of limit cycle was taken from Nonlinear Dynamic System *"The Limit cycle is a cycle that is isolated and closed trajectory"* [5]. Fig. 2 shows the limit cycle according to the system stability.

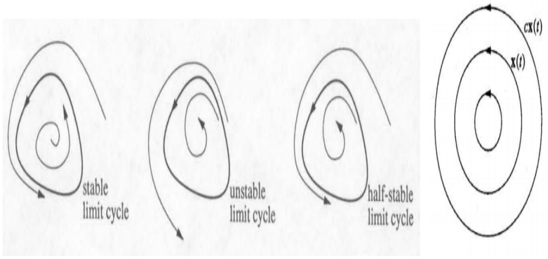


Fig. 2. Limit Cycle according to the stability

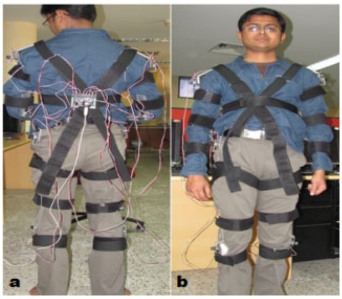


Fig. 3. (a) Rear (b) Front view of IGOD [1]

3.3 Intelligent Gait Oscillation Detector (IGOD)

Intelligent Gait Oscillation Detector (IGOD) is a self made rotation sensor based biometric suit which is used to capture different major joints [(Hip, Knee, Shoulder, Elbow) × 2] in terms of angle value oscillations involved in human locomotion [1]. In our work we have only considered two hip joints and two knee joints. Fig. 3 depicts the rear and front view of IGOD suit.

3.4 Rayleigh Oscillators

Rayleigh Oscillator is a Relaxation Oscillator. It means the oscillator is based upon performance of the physical system and with the condition of returning to the equilibrium position after being perturbed (small external force).

The second order differential equation of the Rayleigh oscillator is

$\ddot{a} - \alpha(1 - \dot{a}) + \mu^2 a = 0$ Without forced condition and

$\ddot{a} - \alpha(1 - \dot{a}) + \mu^2 a = \gamma \sin At$ For forced condition.

Here μ parameter controls the amount of voltage (energy) goes into our system. α is frequency controlling the technique in which voltage flows in the system.

Fig. 4 (A) represents the Matlab plot of **a** vs. time **t** and (B) represents the limit cycle of a Rayleigh Oscillator.

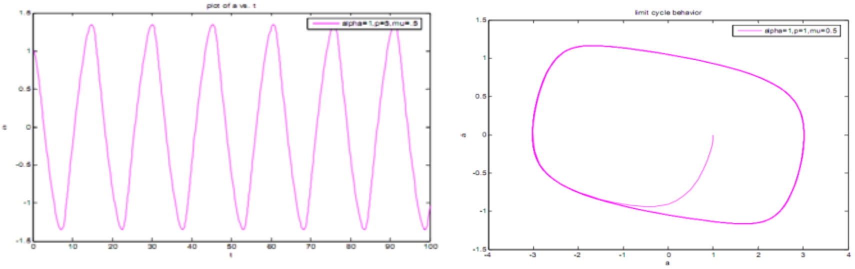
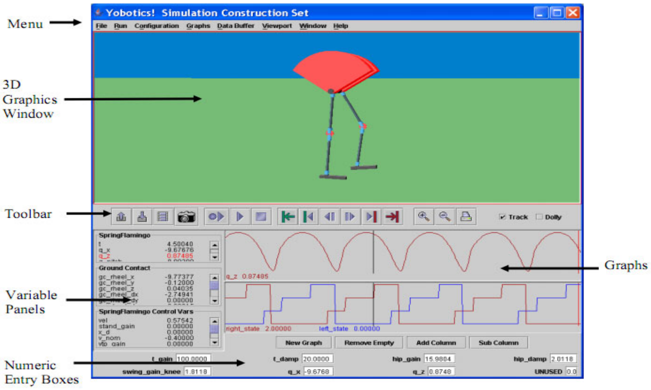


Fig. 4. (A)The graph represent plot of **a** vs. time **t** and (B) Limit Cycle of Rayleigh Oscillator Where α=1, μ=0.5

3.5 YOBOTICS

YOBOTICS is a simulation tool for robot simulation. The construction is the totally-featured software package to simple and rapidly generating simulations for mechanical system like biped locomotion, biomechanical model regarding robots [4]. Fig. 5 shows the different components of YOBOTICS robotics simulation tool.



SpringFlamingo

```
t                4.50040
q_x             -9.67676
q_z              0.87485
```

Ground Contact

```
gc_rheel_x      -9.77377
gc_rheel_y      -0.12000
gc_rheel_z       0.04035
gc_rheel_dx     -2.74941
gc_rheel_dy      0.00000
```

SpringFlamingo Control Vars

```
vel              0.57542
stand_gain       0.00000
x_d              0.00000
v_nom           -0.40000
vtp_gain         0.00000
```

```
t_gain 100.0000    t_damp 20.0000    hip_gain 15.9804    hip_damp 2.0118
swing_gain_knee 1.8118    q_x -9.6768    q_z 0.8748    UNUSED 0.0
```

Fig. 5. GUI window of YOBOTICS simulation software with a Spring Flamingo robot

4 CPG Modeling

In our work we modeled the CPG according to the concept of Nonlinear Dynamic System (NDS). According to the NDS concept if we can couple the relaxation oscillators then the system can be able to produce different rhythmic patterns and also we can be able to check the system stability according to this concept. Here we have used Two-Way coupling concept. The CPG model with all four Rayleigh oscillators is shown in Fig. 6 (a) and Fig. 6 (b) showing the different coupling parameters.

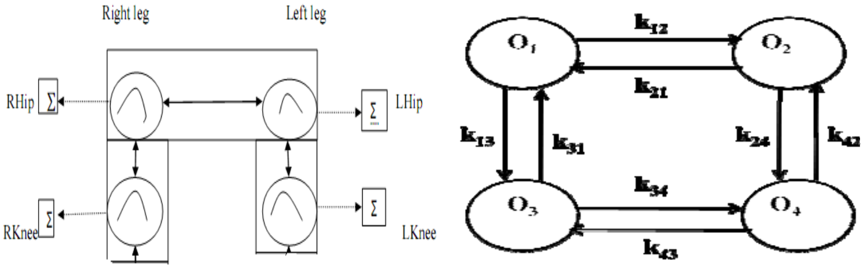


Fig. 6. CPG Model (a) Oscillators position with two way coupling (b) Different coupling parameters.

In this figure O_1, O_2, O_3, O_4 represent four Rayleigh oscillators. k_{12}, k_{21} are coupling parameters between oscillator O_1 and O_2. k_{24}, k_{42} are parameters between oscillator O_1 and O_4. The parameters between oscillator O_3 and O_4 are k_{34} and k_{43}, and k_{31}, k_{13} are parameter between oscillator O_1 and O_3.

4.1 Rayleigh Oscillator Coupling

As we already did the basic architecture of the modeling of the CPG then the implementation phase comes into under consideration. The implementations are categorized into two different parts.

First part in our model, we started placing the Rayleigh oscillators at the different rhythm generating position i.e. left side knee, right side knee, left side hip and right side hip location. These four Rayleigh oscillators are as follows that are in the form of second order differential equation.

For left side hip position's equation: $\mathbf{\ddot{a}_1 - \alpha_1 (1\text{-}d_1 \acute{a}_1^2) \acute{a}_1 + \mu_1^2 (a_1\text{-}a_{10}) = 0}$ **------- (A)**
For the right side hip position's equation: $\mathbf{\ddot{a}_2 - \alpha_2 (1\text{-}d\ {}_2\acute{a}_2^2) \acute{a}_2 + \mu_2^2 (a_2\text{-}a_{20}) = 0}$ **--- (B)**
For the left side knee position's equation: $\mathbf{\ddot{a}_3 - \alpha_3 (1\text{-}d_3\acute{a}_3^2) \acute{a}_3 + \mu_3^2 (a_3 - a_{30}) = 0}$ **- (C)**
For right side knee position's equation: $\mathbf{\ddot{a}_4 - \alpha_4 (1\text{-}d_4\acute{a}_4^2) \acute{a}_4 + \mu_4 (a_4 - a_{40}) = 0}$ **------ (D)**

Here these parameter $d_{1,}$ $d_{2,}$ $d_{3,}$ d_4, μ_1^2, μ_2^2 μ_3^2 μ_4^2, α_1, α_2, α_3, α_4 refer to positive constants in the Rayleigh oscillators. Changing these parameters permit the modification of the frequency of generated signal and amplitude of generated signal.

Simulation is done in matlab7.5 environment. Second order differential equations are complicated to solve it easily. We have applied matlab7.5 on it to get first order differential equations. Now representing the first order equation as A, B, C and D are written below:

Form equation (A) we found

$\mathbf{\acute{a}_1 = z_1}$ and $\mathbf{\acute{z}_1 = \alpha_1 (1\text{-}d_1 z_1^2) z_{1\ .}\ \mu_1^2 (a_1 - a_{10})}$ **------------------- (e)**

Form equation (B) we found

$\mathbf{\acute{a}_2 = z_2}$ and $\mathbf{\acute{z}_2 = \alpha_2 (1\text{-}d_2\ z_2^2) z_{2\ .}\ \mu\ {}_2^2 (a_{2\ \_}\ a_{20})}$ **------------------ (f)**

Form equation (C) we found

$\mathbf{\acute{a}_3 = z_3}$ and $\mathbf{\acute{z}_3 = \alpha_3 (1\text{-}d_3 z_3^2) z_{3\ \_}\ \mu_3^2 (a_3 - a_{30})}$ **------------------ (g)**

Form equation (D) we found

$\mathbf{\acute{a}_4 = z_4}$ and $\mathbf{\acute{z}_4 = \alpha_4 (1\text{-}d_4 z_4^2) z_{4\ \_}\ \mu_4^2 (a_4 - a_{40})}$ **------------------ (h)**

The four Rayleigh oscillators in our model will produce four output signals autonomously. Here, all oscillators are not affecting each other because there is no coupling. In order to produce the preferred rhythmical output patter next task is to be linked with all oscillators with each other or coupling them.

Secondly we did interconnection among all four oscillators with each other. In this work we have used coupling concept. Coupling basically implies two types, one is refereeing One-Way coupling and other is directing to Two-Way coupling. In this paper a two way coupling technique has been applied. In Two- way coupling type, if two or more oscillators are interrelated then all the oscillators effect on each other. It has been observed that first oscillator effects on second oscillator and second oscillator effects on first one for linking the all of four Rayleigh oscillators that are used for left side knee, right side knee, left side hip and right side hip location. In order to provide encouragement this idea came from the association among left side knee, right side knee, left side hip and right side hip joints of humans at the time of simple walking. If we talk about biped locomotion in human being a situation is arrived to locate one leg is in stance phase (on ground) the other side leg is in the situation of swing phase (in air) [refer to Fig. 1]. As a result, we can always exempt phase association stuck between the left side knee's joint angle & right side knee's joint angle the hip angle differently other is knee joint angles are synchronized. If we talk about hip difference angle then we can say that it gives an oscillatory performance throughout locomotion, angle difference oscillates in mean while positive value and then negative values.

Therefore all the four oscillators are interlinked to do so facts discussed in above section. These second order differential equation showing all four oscillators has considered only one term in account of feedback from one to other oscillator. Following are the equation for this system after coupling oscillators:

$$\ddot{a}_1 - \alpha_1 (1 - d_1\acute{a}_1^2)\, \acute{a}_1 + \mu_1^2 (a_1 - a_{10}) - k_{13} (\acute{a}_3 (a_3 - a_{30})) - k_{12} (\acute{a}_1 - \acute{a}_2) = 0 \text{ ---- (i)}$$
$$\ddot{a}_2 - \alpha_2 (1 - d_2\acute{a}_2^2)\, \acute{a}_2 + \mu_2^2 (a_2 - a_{20}) - k_{24} (\acute{a}_4 (a_4 - a_{40})) - k_{21} (\acute{a}_2 - \acute{a}_2) = 0 \text{ ---- (j)}$$
$$\ddot{a}_3 - \alpha_3 (1 - d_3\acute{a}_3^2)\, \acute{a}_3 + \mu_3^2 (a_3 - a_{30}) - k_{31} (\acute{a}_1 (a_1 - a_{10})) - k_{34} (\acute{a}_3 - \acute{a}_4) = 0 \text{ ---- (k)}$$
$$\ddot{a}_4 - \alpha_4 (1 - d_4\acute{a}_4^2)\, \acute{a}_4 + \mu_4^2 (a_4 - a_{40}) - k_{42} (\acute{a}_2 (a_2 - a_{20})) - k_{43} (\acute{a}_4 - \acute{a}_3) = 0 \text{ ---- (l)}$$

4.2 Optimization of CPG Parameters Using GA

Now we need to optimize the different parameters of CPG. In our work we choose Genetic Algorithm (GA) as an optimization technique. The fitness function for GA is the difference between angles that is joint angles generated by our CPG model and the joint angle captured by IGOD suit. Here e(t) is the difference between the angle value in time **t**. So the fitness function is

$$E_d(t) = \beta_1 e(t) + \beta_2 de(t) / dt + \beta_3 \int e(t)\, dt \text{ --------------- (p)}$$

β_1, β_2 and β_3 considered as Proportional Constant, Differential Constant and Integral Constant respectively. According to our fitness function reduce the function value means reduce the angle difference that means we are going towards the generation of natural human like walking pattern by our CPG model for our robot.

Now differentiating the equation (p) with respect to t:

$$\beta_1\ de(t)/dt + \beta_2\ d^2e(t)/dt^2 + \beta_3\, e(t) = dE_d(t)/dt \text{ --------------- (q)}$$

Now consider that the system is in steady state condition that means system within the virtual static state. In condition of steady state is $\frac{de(t)}{dt} \to 0$, $\frac{d^2e(t)}{dt^2} \to 0$. We know that $E_d(t)$ is constant and $\beta_3 e(t) = 0$, but β_3 is not equals to 0 because this is considered as positive constant, that means **e(t) → 0 ------- (r).**

Hence we can say that the fitness function reduces the fault. Therefore the fitness function (p) will decrease the steady state error to 0.

5 Analysis of Our CPG Model

In this part we will show the CPG parameters we obtain from GA and the walking pattern generated by our CPG model. In our work the fitness function (p) is converged to 0.001, that means **e(t) → 0.001 .** So the optimized value we get from GA is k_{12}=.2111, k_{13}=.1125, k_{24}=.1129, k_{21}=.3010, k_{31}=.1125, k_{34}=.2012, k_{42}=.1129, k_{43}=.2012, α_1=.0314, α_2=.0220, α_3=.0208 and α_4=.0308.

Fig. 7 shows the rhythmic patterns generated by our CPG model.

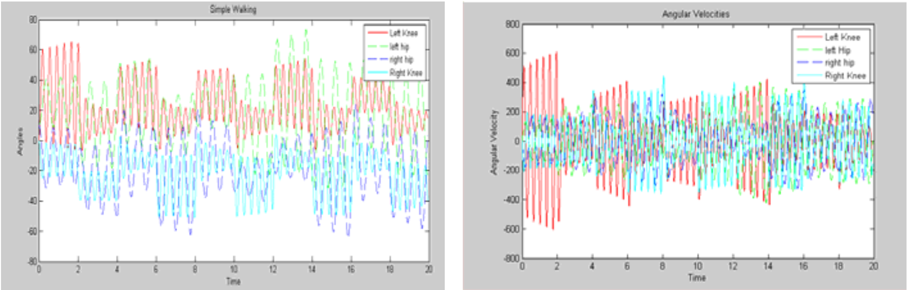
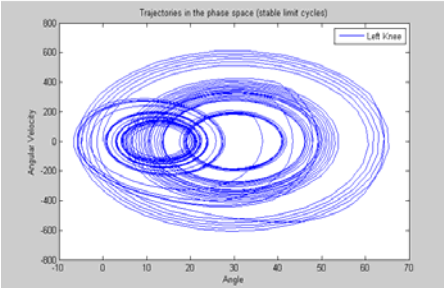


Fig. 7. The pattern generated by our CPG model of different joints (a) Angle vs. Time graph were angle is in degree and time is in Second. (b) Velocity vs. Time graph.



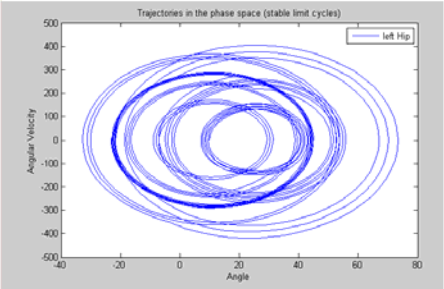
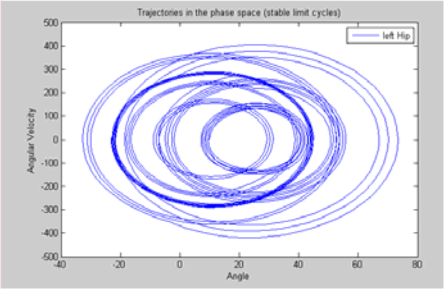


Fig. 8. Phase diagram of Left Knee joint

Fig. 9. Phase diagram of Left Hip joint



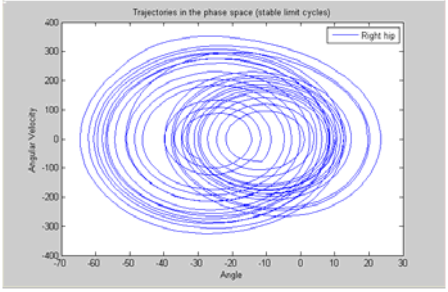


Fig. 10. Phase diagram of Right Knee joint

Fig. 11. Phase diagram of Right Hip joint

Now coming to the phase space trajectory graphs those are also known as limit cycle which should be in stable state for stable walking of a Robot. Fig. 8, 9, 10 and 11 shows the phase space trajectory graph for left knee, left hip, right knee and right hip respectively. All these phase diagram start from Origin and converged to constant oscillatory swinging action and have a stable limit cycle.

6 Simulation

In our work we have used Matlab 7.5 and YOBOTICS robotics simulation environment. The Differential equation solver presented in Matlab 7.5 is used for modeling the CPG. The implementation part of GA is also done in Matlab 7.5. This experiment provides us some patterns those are being tested on YOBOTICS simulator with a Spring Flamingo Robot. It also gives the oscillatory activity of the CPG where angle are considered in radian.

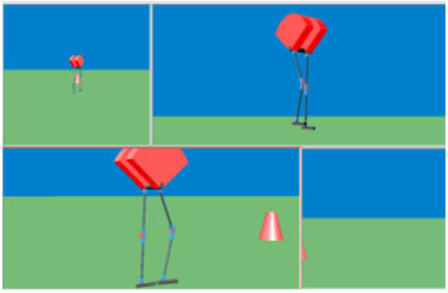


Fig. 12. Walking of a Spring Flamingo robot

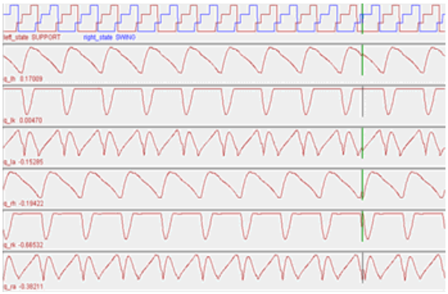


Fig. 13. Oscillation activity of each joint

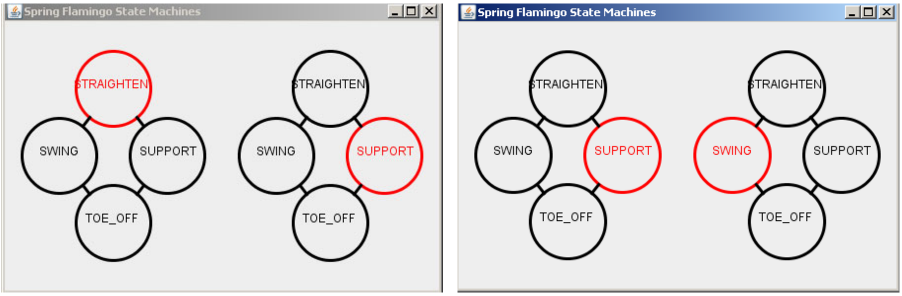


Fig. 14. Shows the state of left and right legs when the robot is walking. (A) Left leg is in straightening state while right is in support state. (B) Left leg is in support state while right is in swing state.

In this environment spring damper system is used for modeling the ground. The coefficient of the spring is 40000N/m and 100N/m for damping. The Ts is time interval having value 0.5ms. The pattern we have got from CPG given to this simulator is in the form of CSV (Comma Separated Value) file format. In this simulator we can export the CSV file and run it freely. Since CPG is matched to an actual human gait oscillation; the ratio of the limb dimension has been kept similar to that of a human. After running it we will get the pattern and intended to prove of our CPG model is working or not. Fig. 12 is the snap shot of a walking Spring Flamingo robot from three camera view in YOBOTICS environment. Fig. 13 shows the each joint oscillation activity when the Spring Flamingo robot is walking. Fig. 14 shows the state diagram of our robot within a particular gait cycle when the robot is walking. Fig. 15 shows the plot of the robot state diagram.

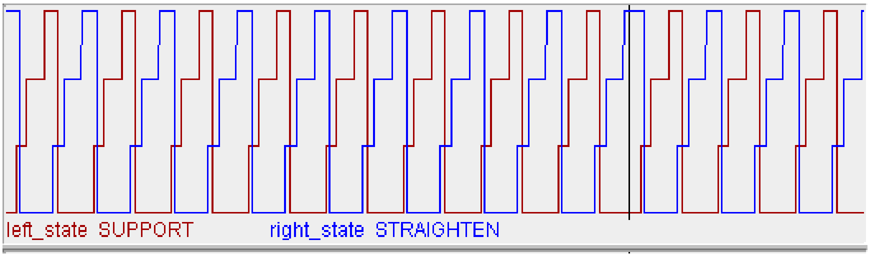


Fig. 15. Plot of the state diagram when the robot is walking

7 Conclusion and Future Work

In this work we have explored Rayleigh oscillator for modeling of CPG controller for biped locomotion. There are certain things needs to be done as future work related to this work. In this model of CPG we have used four joints only as compared to our desired result. Now we could look for humanoid robot HOAP 2 (Humanoid Open Architecture Platform 2) for considering of 26 joints having developed a CPG based model accordingly. Sensory feedback plays a critical role which could be incorporated to deal with perturbation like wind slopes etc. It includes the extension of sensory inputs which are needed in dealing with environment easily.

So far we have worked upon on Rayleigh oscillator for constructing biologically inspired CPG based model for generating rhythmic movement of bipedal robot. It has highlighted many drawbacks in comparison to MATSUOKA oscillator [6][7][8]. In future modeling scenario we could suggest MATSUOKA oscillator for generation of rhythmic pattern of bipedal robot. Irrespective of the simulation work of bipedal robot implemented on simulated software we would implement it on real humanoid robot in real world environment.

Acknowledgment. This work was supported by Indian Institute of Information Technology, Allahabad.

References

1. Mondal, S., Nandy, A., Chakrabarti, A., Chakraborty, P., Nandi, G.C.: A framework for synthesis of human gait oscillation using intelligent gait oscillation detector (IGOD). In: Ranka, S., Banerjee, A., Biswas, K.K., Dua, S., Mishra, P., Moona, R., Poon, S.-H., Wang, C.-L. (eds.) IC3 2010. CCIS, vol. 94, pp. 340–349. Springer, Heidelberg (2010)

2. Taga, G.: A model of the neuro- musculo-skeletal system for anticipatory adjustment of human locomotion during obstacle avoidance. Biological Cybernetics 78(1), 9–17 (1998)
3. QiDi, W., ChengJu, L., JiaQi, Z., QiJun, C.: Survey of locomotion control of legged robot inspired by biological concept. Information Science, Science in China Series F, 1715–1729 (2009)
4. Yobotics simulation software, `http://yobitics.com/simulation/simulation.htm`
5. Pikovsky, A., Rosenblum, M., Kurths, J.: Synchronization, A universal Concept in non-linear sciences. Cambridge University Press, Cambridge (2001)
6. Matsuoka, K.: Sustained oscillations generated by mutually inhibiting neurons with adaptation. Biological Cybernetics 52, 367–376 (1985)
7. Matsuoka, K.: Mechanisms of frequency and pattern control in the neural rhythm generators. Biological Cybernetics 56, 345–353 (1987)
8. Williamson, M.M.: Design Rhythmic Motions using Neural Oscillators. In: IEEE/RSJ IROS 1999, pp. 494–500 (1999)
9. Grillner, S.: Control of locomotion in bipeds, tetrapods and fish. In: Brooks, V.B. (ed.) Handbook of Physiology, Sect. I: The Nervous System II, Motor Control, American Physiological Society, Waverly Press, Bethesda, Maryland (1981)
10. Cohen, A.H., Rossignol, S., Grillner, S.: Neural Control of Rhythmic Movements in Vertebrates. John Wiley & Sons, New York (1998)
11. Abbas, J.J., Full, R.J.: Neuromechanical interaction in cyclic movements. In: Winters, J.M., Crago, P.E. (eds.) Biomechanics and Neural Control of Posture and Movement, Springer, New York (2000)
12. Nandi, G.C., Ijspeert, A.J., Nandi, A.: Biologically inspired CPG based above knee active prosthesis. In: IEEE/ RSJ IROS 2008, pp. 2368–2373 (2008)
13. Nandi, G.C., Ijspeert, A.J., Chakraborty, P., Nandi, A.: Development of Adaptive Modular Active Leg (AMAL) using bipedal robotics technology. Robotics and Autonomous Systems 57, 603–616 (2009)
14. kassim, M. H., Zainal, N., Arshad, M. R.: Central Pattern Generator in Bio-inspired Simulation using MATLAB. In: MEDINFO 1998 (1998)
15. Filho, A.C., De, P.: Simulating the Hip and Knee Behavior of a Biped by Means of Nonlinear Oscillators. The Open Cybernetics and Systemic Journal 2, 185–191 (2008)
16. Ijspeert, A.J.: Central Pattern Generators for locomotion control in animals and robots: A review. Neural Networks 21, 642–653 (2008)
17. Ijspeert, A.J., Crespi, A., Ryczko, D., Cabelguentics, J.M.: From swimming to walking with a salamander robot driven by a spinal cord model. Science 315, 1416–1420 (2007)
18. Ijspeert, A.J., Crespi, A., Cabelguentics, J.M.: Simulation and robotics studies of salamander locomotion: Applying neurobiological principles to the control of locomotion in robotics. Neuroinformatics 3, 171–195 (2005)
19. Kimura, H., Fukuoka, Y., Cohen, A.H.: Adaptive dynamic walking of a quadruped robot on natural ground based on biological concepts. International Journal of Robotics Research 26, 475–490 (2007)
20. Kimura, H., Akiyama, A., Sakurama, K.: Realization of dynamic walking and running of the quadruped using neural oscillators. Autonomous Robots 7, 247–258 (1999)
21. Kimura, H., Tsuchiya, K., Ishiguro, A., White, H.: Adaptive Motion of Animals and Machines. Springer, Heidelberg (2005)
22. Williamson, M.M.: Robot arm control exploiting neural dynamics. In: PhD Thesis, MIT, Cambridge, MA (June 1999)

23. Lewis, M. A., Tenore, F., Cummings, R. E.: CPG design using inhibitory neurons. In: IEEE/ RSJ ICRA 2005 (2005)
24. Sekiguchi, A., Nakamura, Y.: Behavior Control of Robot Using Orbits of Nonlinear Dynamics. In: IEEE/RSJ ICRA 2001, pp. 1647–1652 (2001)
25. Nakamura, Y., Yamazaki, T., Mixushima, N.: Synthesis, Learning and Abstraction of Skills through Parameterized Smooth Map from Sensors to Behaviors. In: IEEE International Conference on Robotics and Automation, pp. 2398–2405 (1999)
26. Sekiguchi, A., Nakaniura, Y.: The Chaotic Mobile Robot. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 172–178 (1999)
27. Williams, C.A., DeWeerth, S.P.: Resonance tuning of a neuromechanical system with two negative sensory feedback configurations. Neurocomputing 70(10-12), 1954–1959 (2007)

Opinion Based Trust Evaluation Model in MANETs

Poonam[1], K. Garg[2], and M. Misra[1]

[1] Dept. of Electronics & Computer Engineering
IIT Roorkee, Roorkee, India
[2] Manipal Institute of Technology, Manipal
{pgeradec,manojfec}@iitr.ernet.in, kumkum.garg@manipal.edu

Abstract. A secure routing mechanism is the basis of security in mobile ad hoc networks. Seeing that nodes have to share routing information in order to find the route to the destination, further an ad hoc network is an open setting where everyone can participate, trust is a key concept in secure routing mechanisms. In this paper, we propose a novel opinion based trust evaluation model to detect misbehaving nodes and further securing data transmission by avoiding such node from path selected. Misbehaving nodes are detected based on the node own direct observation, but in the circumstances of network failure as node congestion or collusion a well-behaving node can be misidentified. Therefore we require opinion of other nodes before proceeding to isolation of such misbehaving node. Source nodes maintain a path reliability index associated with each path to the intended destination. The most trustworthy and reliable path is selected by source node for forwarding data packets to the intended destination. We have evaluated the misbehaving node detection rate and the performance of our method along a number of parameters through simulation. Results show that our method increases the throughput of the network while also discovering a secure route.

Keywords: Trust, Opinion, misbehaving nodes, Attacks, Multipath routing.

1 Introduction

A mobile ad-hoc network (MANET) is a rapidly deployable, multi-hop wireless network. Neither pre-defined network infrastructure nor centralized network administration exists to assist in the communication in MANETs. Moreover, absence of a routing infrastructure that would assure connectivity of mobile nodes precludes using the traditional internet protocols for routing, name resolution, trust establishment etc. Therefore, in such networks, each node simultaneously acts as a host and as a router to forward packets for other peer nodes.

In MANETs nodes act both as hosts and routers, and thus cooperatively provide multi-hop strategy to communicate with other nodes outside their transmission range. All nodes in the network execute a pre-agreed routing protocol to pass packets from other nodes. The accurate execution of these protocols requires sustained and benevolent behavior by all participating nodes.

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 301–312, 2011.

The problem of all the current ad hoc routing protocols is that all the nodes are considered as trust worthy and assume that they behave properly. Therefore, they are vulnerable to attacks launched by misbehaving nodes. According to [1] misbehaving nodes can be categorized as *selfish* or *malicious*. Misbehaving nodes participate in the route generation process to sabotage the network. These nodes can agree to forward packets on behalf of other nodes but silently drop packets in an attempt to save their resources. Malicious nodes may try to sabotage other nodes or even the whole network. For example, one malicious node can advertise itself as having the shortest path to all nodes in the network. It can cause Denial of Service (DoS) by dropping all the received packets, in *Black hole attack*, or selectively dropping packets in *Gray hole attack*. This gives rise to the need for secure routing protocol to provide network services under the presence of misbehaving nodes. Seeing that nodes have to share routing information in order for each node to find the route to the destination, and that an ad hoc network is an open setting where everyone can participate, trust is a key concept in secure routing mechanisms.

Therefore in thus paper, we propose a novel opinion based trust evaluation model to detect misbehaving nodes and further securing data transmission by avoiding such node from path selected. Misbehaving nodes are detected based on the node own direct observation, but in the circumstances of network failure as node congestion or collusion a well-behaving node can be misidentified. Therefore we require opinion of other nodes before proceeding to isolation of such misbehaving node. Source nodes maintain a path reliability index associated with each path to the intended destination. The most trustworthy and reliable path is selected by source node for forwarding data packets to the intended destination.

The rest of this paper is organized as follows. In Section 2 the related work is given, followed by a detailed description of our solution in Section 3. In Section 4 we have described the process to selected optimized secure path free from misbehaving nodes. In section 5 we evaluate the efficiency of our method through exhaustive simulation. The last section concludes the paper and gives suggestions for further work in this area.

2 Related Work

A number of secure routing protocols have been developed that conform to mitigate the effect of misbehaving nodes in mobile ad hoc networks. These protocols employ a variety of cryptographic tools for protecting the vulnerabilities in different routing protocols. However, these protocols have been developed as a practical response to specific problems that arose due to attacks on ad hoc network routing protocols. Consequently, these protocols only cover a subset of all possible threats and are not flexible enough to be integrated with each other. Some of the related previous work that has been carried out in order to make ad-hoc networks more trustworthy is explained in this section.

Marti et al. [1] proposed a mechanism to identify the misbehaving node in the network. They introduced the concept of ‘‘watchdog” and ‘‘pathrater”. Watchdog by listening the next node’s transmission promiscuously identifies the misbehaving nodes and then pathrater uses this information to select the appropriate path. The authors identified watchdog's weakness in terms of receiver collisions, limited transmission power, false misbehaving, collusion and partial dropping. Pathrater uses the information of watchdog to efficiently calculate trust of a node. It calculates a path metric by averaging the node rating in the path and if there is more than one path to the destination, the path with the highest metric is selected. And if all nodes are of the same cost in the path to destination then the shortest path is selected.

Many cooperation enforcement mechanisms are based on Watchdog or similar mechanisms. CORE [2] detects and excludes both malicious and selfish nodes. With CORE, a node *N* maintains trust values for every neighbor, when the trust of a node *M* falls below a threshold; *N* stops to forward packets emitted by *M*. To spread the nodes reputation among the network, recommendations are sent. CONFIDANT [3], proposed by Buchegger and Le Boudec, only allows negative recommendations to advertise that a malicious node is detected. Blackmail is prevented since nodes take in consideration only ALARMs sent by trusted nodes. With OCEAN [4], a node directly bypasses distrusted nodes during route establishment appending their ID in a blacklist included in the route request they send.

Pirzada and McDonald [5][6] proposed a trust model in which every node has a trust agent to calculate the trust level of its neighboring node. The model does not demand any extra packets or buffers to assign trust level to other nodes. The model uses link-layer and network layer acknowledgments for trust calculation. It is a good attempt for security in pure ad hoc networks but there are many issues left unaddressed. In a passive acknowledgment situation there is a possibility that a node is unable to forward a packet due to heavy traffic load, low battery power or less processing power, this behavior will be treated as a misconduct of that specific node. Explicitly requested network layer acknowledgments can introduce unwanted delays in high density ad hoc networks. The model does not handle malicious colluding nodes, which is a very common attack.

In [7], Park et al. proposed a solution to identify black hole attack. The first part of solution requires finding more than one paths to the destination, the source node waits for RREP packet to arrive from more than two nodes. Any node that receives the first packet will not drop the second one if it exists in both paths. From shared hops the source node can recognize safe route but the problem with this mechanism becomes non-trivial when more than two paths are not available. This solution can introduce unwanted delays by storing packets.

Patcha and Mishra [8] extended the concept of watchdog [1] to handle colluding malicious nodes. Six extra control packets were created to implement security extension in AODV. The watchdog node is selected on the basis of node energy, node storage capacity and node computing power. The induction of six extra packets degrades network performance and watchdog idea is vulnerable because watchdog nodes can be crashed and impersonated. This creates a single point of failure in the monitoring area of that specific watchdog node.

3 Opinion Based Trust Evaluation Model in MANETs

In almost all previous existing work, trust is quantized based on direct and indirect observation. Indirect observation or reputation gives higher chances for the false information propagation and it also consume a good part of network bandwidth. By keeping all such points in mind we have quantized trust between nodes based on direct observation to detect misbehaving nodes. The trust value of a node is calculated based on the events directly experienced by the node. The trust calculation is continuous process so it is performed in both route discovery and data delivery phase. Because it is possible a node may show a good behaviour in route discovery phase to setup a path around it and may launch a black hole or grey hole attack by dropping the packet.

As the first step toward the solution to the problem, we model the network into hierarchical network between a group of neighbor nodes (NN) monitored node and the supervisor node (SN). This hierarchical network can also be considered as a distributed information aggregation system. In this network supervisor node is the node monitoring the behavior of intermediate node or monitored node, further NN are neighbor of either MN or SN or both. The information provided by neighbor node (NN) is considered if it's immediate neighbor of either MN or both MN and SN. Because the immediate neighbor of MN are the direct observer of it and they provide the most accurate information. We have also given consideration to the distance between SN and NN. As the distance increases the chances of information modification also increases.

Our solution is able to detect misbehaving nodes that drop or modify packets. We deal all type of packets i.e. data, routing and control packets. Our model is able to capture independent packet forwarding and node recommendation propagation misbehaviors. Therefore, it is able to withstand against the presence of misbehaving nodes and bad recommenders. Moreover bad recommenders launch bad mouthing attack by constantly reporting incorrect information about other node. This leads supervisor node (SN) to make a wrong decision and black list a well behaving node, this result due to the effect of the malicious node incorrect reporting. It is therefore an important issue in mobile ad hoc networks to detect malicious nodes in spite of such problems.

3.1 Trust Assignment

Trust is assigned by SN based on observed behavior of MN. It is continuous process so it is performed in both route discovery and data delivery phase. Because it is possible a node may show a good behaviour in route discovery phase to setup a path around it and may launch a black hole or grey hole attack by dropping the packet.

We present a realistic opinion based trust model that takes into account not only the packet forwarding events but also verifies a number of other parameters including the integrity of forwarded traffic and sincerity in execution of the routing protocol. We have fine-tuned our trust model for the DSR protocol and makes use of effective mechanisms, which facilitate trust derivation and its subsequent application to the routing process. Each node is given an integer trust value lyingbetween -1to1.If a new node joins the network, it sends a hello packet to its neighbors. Each NN would move the node to well-behaved list and assign trust value as 1. All the nodes in well behaved list are having trust value as 1. The nodes in suspect list are assigned trust value

0 and nodes existing in black list are assigned trust value as -1. The trustworthiness of thenode can be increased if the node shows benevolent behavior.We do not consider physical layer and link layer attacks, like jamming attacks, in this paper.

3.2 Direct Monitoring Based Node Categorization

It is responsible for evaluating a node behavior and categorizing it as well-behaved, misbehaving or suspect node. Every node is responsible for monitoring the behavior of its neighbor and then discriminate misbehaving nodes from well behaving nodes. It makes sure that its neighbors forward packets relayed through them as expected. An observation that a neighbor forwards a packet routed through it is interpreted as an indication of cooperative behavior. An observation that a neighbor does not forward a packet routed through it is interpreted as an indication of misbehavior. Supervisor node classifies its neighbor nodes based on the behavior observed. Each supervisor node in the network maintains three lists which are as well behaved, suspect list and black list. When a new node join the network its neighbor node assign it to well behaved list, afterwards based on the behavior observed node either stay in that list or move in any of the other list. All the neighbor nodes which are well behaved are maintained in the *well-behaved* list by the supervisor node. *Suspect table* contain set of nodes which are being suspected as misbehaving as shown in table 1. In this list supervisor node maintain the node id and the number of times the node has been detected as misbehaving. Finally in the *black list*, the nodes confirmed as misbehaving are moved to this list.

Table 1. Suspect table

Node ID	Suspect Count
B	2
D	1

Each node monitors and evaluates its immediate next node for each transmitted packet through snooping the channel in the forwarding phase. Each node act as supervisor node while forwarding the packet to its neighbour node. It makes sure that its neighbours forward packets relayed through them as expected. An observation that a neighbour forwards a packet routed through it is interpreted as an indication of cooperative behaviour. An observation that a neighbour does not forward a packet routed through it is interpreted as an indication of misbehaviour.

Therefore in the promiscuous mode after forwarding a packet each node store the packet, and listens to packet retransmission by its neighbor node. When it gets a packet back from its neighbor in certain time limit which is equal to two way propagation delay then it checks for the integrity of the message. If SN neither receive the overhear packet nor any link failure notification in certain time limit it conclude a packet drop attack. The MN is suspected as misbehaving and suspect list is modified accordingly.

Each suspected node is moved to suspected list but if the node exists in the list then the count is incremented. If its detection count (DC) crosses a pre-defined threshold then it call detector module to discriminate node misbehavior from network fault.

But these observations are performed using promiscuous mode overhearing. Therefore the trust models are vulnerable to the limitation of promiscuous mode hearing [1]. Moreover in circumstances of network failure as node congestion or collusion a well-behaving node will be identified as misbehaving due to packet drop. Therefore we have chosen threshold = 3 to withstand against wrong detection. This value is decided by extensive simulation with aim to achieve maximum efficiency. It helps to differentiate between false detection due to network congestion or collusion and genuine detection of misbehaving nodes. When the supervisor node overhear the packet transmitted but the packet fails integrity check then such monitored node is detected as misbehaving and moved to black list. In figure 1 provide flowchart for the above module.

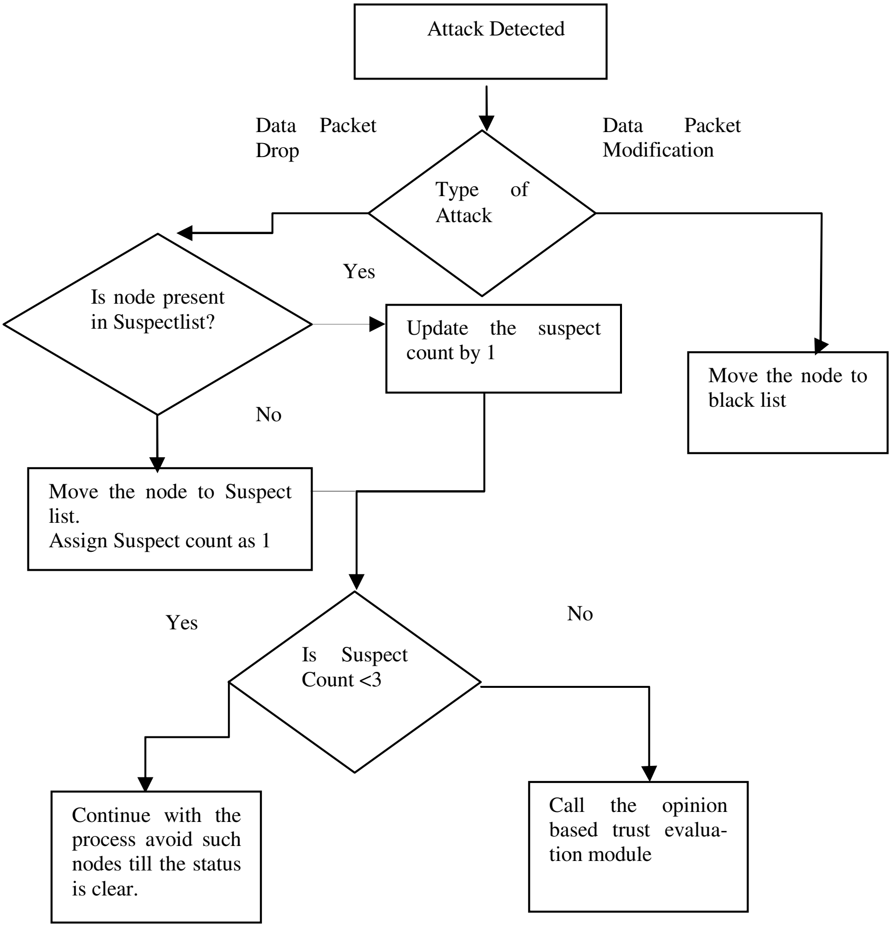

Attack Detected
Data Packet Drop
Data Packet Modification
Type of Attack
Yes
Is node present in Suspectlist?
Update the suspect count by 1
Move the node to black list
No
Move the node to Suspect list.
Assign Suspect count as 1
Yes
No
Is Suspect Count <3
Continue with the process avoid such nodes till the status is clear.
Call the opinion based trust evaluation module

Fig. 1. Flowchart of Direct Monitoring based Node categorization

3.3 Misbehaving Node Detection

When the supervisor node suspect a monitored node (MN) as misbehaving then it calls this module, seeking for the opinion provided by other nodes for the detection and isolation of monitored node. SN gathers information about MN reported by NN in form of opinion. Based on the information collected from NNs, SN computes the aggregation result and move the MN to either well behaved or black list based on it. It is possible that NN providing opinion to SN may be compromised and report fake information, it is important for SN to verify the correctness of the information collected from NNs.

Therefore importance given to the opinion provided by NN or opinion weightage is proportional to the trust value of NN and reciprocal to the distance of NN as given in equation1 and equation 2.

$$Opinion_{weightage} \propto Trust_{NN} \tag{1}$$

$$Opinion_{weightage}\ 1/\propto\ \ Distance_{NN} \tag{2}$$

SN discards the opinion provided by NN having trust value < 1. The opinion regarding suspected node is influenced by majority neighbor node's opinion. As it is not possible that majority of nodes act as a malicious in one hop neighborhood.

For gathering the opinion about the suspected node SN broadcast the opinion request (OREQ) packet. It contains the address of opinion seeking and opinion required node address i.e. SN and MN. To limit the network overhead we have to inserts a hop count field with value as 2, which constrain the opinion reply only by the nodes which are one hop neighbor of either SN or MN. We have constrained as the one hop neighbors of MN are the direct neighbor of MN and the information they provide is more significant. Similarly the one hop neighbors are direct monitor of SN and it induces very less network overhead.

SN collects the opinion provided by the neighbor nodes for the monitored node. SN drops the opinion received by black list node as there is high probability that these nodes may be bad recommenders. Moreover bad recommenders launch bad mouthing attack by constantly reporting incorrecting information about other node. This leads supervisor node (SN) to make a wrong decision and black list a well behaving node.

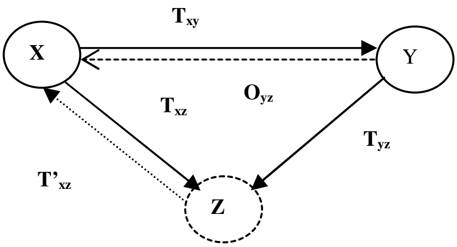


Fig. 2. Opinion Network

Supervisor node categories monitored node as misbehaving or as well behaved based on the majority opinion. The information provided by the nodes is consolidated and weighted opinion is considered for categorizing suspected node as well–behaved or misbehaved node. Figure 2 shows the opinion network where node Z is suspected node, node X is supervisor node or the node seeking opinion and node Y is neighbor of node Y and node Z, further it is node providing opinion regarding node Z to node X. The opinion given by NN is the trust value assigned by node y to node Z. Opinion is the discrete value that recognizes the recommendation for the monitored node by neighbor. It is 1 if the NN provide a positive recommendation, i.e. a node is well behaved and 0 if it negative recommendation or the monitor node is misbehaving.

The weighted opinion ($\mathrm{WO}^{\mathrm{Y}}{}_{\mathrm{XZ}}$) is computed as average sum of opinion provided by node Y for node X and trust value of node Y assigned by node X divided by number of nodes providing opinion.

$$WO_{XZ}^{Y} = \frac{\sum T_{XY} * O_{YZ}}{\sum Nodes\ providing\ Opinion}$$

If the opinion providing NN is in the suspect list of SN, then there will be no impact of their opinion on weighted opinion ($\mathrm{WO}^{\mathrm{Y}}{}_{\mathrm{XZ}}$) as T_{XY} is zero. A MN node having weighted opinion >=0.5 is assigned to well behaved list with trust value as 1otherwise it is assigned to black list with trust value of -1.

3.4 Opinion Distribution Module and Isolation

The misbehavior detection module receives trust value from nodes concerning their belief in other nodes. Similarly it sends its own opinion in form of trust value to other nodes. The exchange of this information between nodes is done through piggybacking it with the packets which are unicasted. These packets could be RREP packets or the data packets. In order to avoid misbehaving nodes to modify or falsify the recommendation sent by other nodes, we use HashCash [9] tokens along with recommendations forwarded. The token is appended with the recommendation sent along the path.

Each node can easily verify the token by performing a single hash operation on it. Each node on the source route of DSR receives information from nodes regarding their belief in other nodes. Similarly it appends its own opinion about other nodes in the path. A node appends the information about the nodes whose trust information has changed drastically or we can say the node which is detected as misbehaving.

Likewise, in order to cheat the trust model, any node can initially behave in a benevolent manner so as to carry out malicious activity later on. However, once it starts the malicious activity, its corresponding trust value that is being maintained by other nodes start decaying. This results in circumventing of the malicious node in subsequent route discoveries. The precise impact of such attacks is very difficult to gauge, especially if the malicious node operates at the threshold between benevolent and malevolent behavior.

4 Secure Path Selection

In DSR [10] protocol, whenever a new route is required it is obtained from the link cache. The route with the least number of hops and highest stability is selected using the Dijkstra [11] shortest-path algorithm. We modify this rule in the opinion based trusted evaluated DSR protocol and associate the trust value of nodes with the default link costs. The different possible link cost values are shown in Table 2.

As mentioned earlier, for optimization of the route discovery process, DSR supports two types of route cache structures: path cache and link cache. In the path cache each route that is received through a RREQ packet is stored as an independent path to the destination. This scheme is easy to implement and also ensures that all paths are free of routing loops. A disadvantage of such a caching scheme is that in case of failure of any single node in the path, the complete route has to be discarded. In contrast, in the link cache scheme individual links are added to a unified graph data structure that represents a node's view of the network topology. The link cache scheme of DSR, allows better utilization of the received routes in comparison to the path cache scheme as it offers alternate routes through the network upon failure of intermediate nodes. The caching nodes also maintain a stability table, which preserves the permanence of the links based upon their last usage in form of path reliability.

Table 2. Link Cost with respect to Trust value Nodes

Node x	Link Cost
1	0
0	1
-1	2

The opinion based trust model associate trust values of nodes with link cost. This allows us to make a decision regarding the selection of optimized route from source to destination. Each time a new route is required source node launch route discovery process. After receiving RREP packets it computes the link cost of the path. The route having the minimum cost is selected for routing process. By associating each link with its respective trust level, we are able to distinguish the level of dependability of nodes in the network. As these costs are realistically computed based upon the events experienced by an individual node, they give an accurate representation of the cost associated with a particular link. Consequently, the routes retrieved from the link cache, during subsequent requests for routes, are found to be more accurate and reliable.

5 Performance Analysis

We have used the QUALNET network simulator (version 4.5) developed by Scalable Network Technologies Inc. [12] to evaluate the effectiveness of the proposed method. Different scenarios are defined in a 1000 * 1000 m square area with 50 nodes. The source and destination nodes are randomly selected. The IEEE 802.11 Distributed Coordination Function (DCF) [13] is used as the medium access control protocol. A traffic generator was developed to simulate constant bit rate (CBR) sources. In every scenario, the nodes move in a random direction using the random waypoint model with a speed randomly chosen within the range of 0–20 m/s. The transmission range of each node is 100 m. We assume that there are 0-40% malicious nodes in the network.

5.1 Metrics

To evaluate the performance of the proposed scheme, we use the following metrics: *Route Discovery Time*: It is defined as the total time required for selecting a path set for routing.

Average Latency: It is defined as the mean time in seconds, taken by the data packets to reach their respective destinations.

Throughput: It is the ratio of the number of data packets received by the destination node to the number of packets sent by the source node.

5.2 Discussion of Results

The results obtained from exhaustive simulation are shown as graphs in fig.3 to fig.6. The proposed opinion based trust evaluation model is examined for its efficiency in presence of misbehaving nodes. To test the efficiency of our method we have varied number of misbehaving nodes from 0 to 40%. Figure 3shows that as the presence of malicious nodes increases the throughput of DSR is decreased and opinion based trust evaluation model works well to identify and isolate the misbehaving nodes and improves throughput by providing secure routing path free of misbehaving nodes. As percentage of misbehaving nodes increases from 0 to 40% throughput is significantly decreased in DSR. But the impact of this decline is appreciably less on the proposed opinion based node evaluation method.

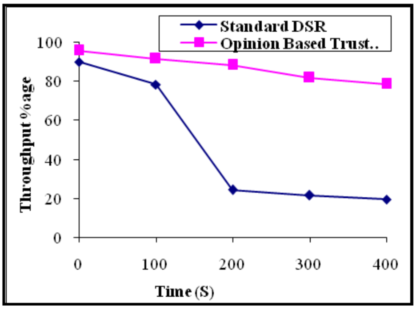


Fig. 3. Throughput in Presence of Black Hole Attack

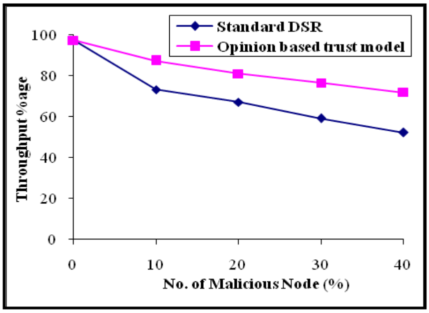


Fig. 4. Throughput

The proposed scheme has been examined for black hole attack as well. Figure 4 shows that throughput of DSR decreases to the minimum level in black hole attack. The black hole node is identified and removed from the path to provide maximum throughput.

The *average latency* of the packets also increases with the number of misbehaving nodes, as the trusted paths are not always the shortest in terms of number of hops. The increase in the path lengths leads to higher latency than that occurring with the Standard DSR as in figure 5. But as number of misbehaving nodes increases in the network it simultaneously increases the rate of route recovery due to the attack launched by misbehaving nodes. So, average latency in DSR increases significantly with respect to increase in misbehaving nodes in the network. This route recovery is delayed in our trust model as path discovered are trust-worthy.

Routing overhead of our trust model is always less than standard DSR, as we accomplish trustworthy routes instead of the standard shortest routes. This leads to decrease in control packets generated due to route breakage by misbehaving nodes, as shown in figure 6.

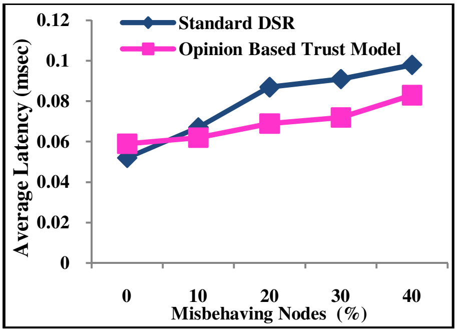


Fig. 5. Average Latency

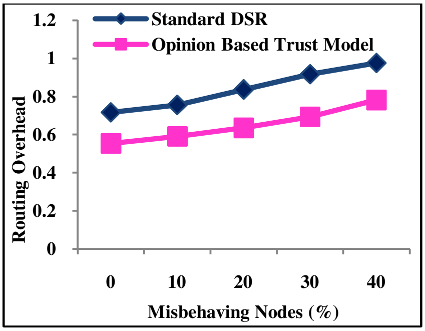


Fig. 6. Routing Overhead

6 Conclusions and Future Work

In this paper we have developed an opinion based trust evaluation model. In this model each node is responsible for evaluating a node behavior and categorizing it as well-behaved, misbehaving or suspect node. Every node is responsible for monitoring the behavior of its neighbor and then discriminating misbehaving nodes from well behaving nodes. Each node passively receives a lot of information about the network. This information is used to build trust levels for different nodes.

Our method is a novel method to identify and isolate the malicious nodes in DSR. It detects and isolates malicious nodes by directly monitoring the behavior of nodes. Any suspicious behavior is confirmed through the opinion of its intermediate neighbor node. False accusation is nullified based on the opinion of other nodes, which are neighbor of the accused node. If any of the node is confirmed as misbehaving then such detection is propagated by monitoring node or supervisor node. The source node contains a list of well-behaved and misbehaving nodes. Therefore, source node makes sure that selected path is secure and free from misbehaving nodes. This process of identification of malicious nodes is dynamic and efficient as the categorization of node changes with respect to their behavior. The simulation results show that the proposed scheme provides better throughput, less routing overhead and less average latency in the presence of misbehaving nodes. The proposed scheme is able to detect black hole attacks as well.

In future we will look at further issues that have not been addressed in this paper, including trust decay over time, trust acquirement through malicious behavior and a security analysis of the proposed model against attacks.

References

1. Marti, S., Giuli, T.J., Lai, K., Baker, M.: Mitigating routing misbehavior in mobile ad hoc networks. In: Proceeding of Sixth Annual International Conference Mobile Computing and Networking (MobiCom), pp. 255–265. ACM Press, New York (2000)
2. Michiardi, R., Molva, C.: Core: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks. In: Communication and Multimedia Security Conference, Portoroz, Slovenia (2002)
3. Buchegger, S., Boudec, J.: Performance Analysis of the CONFIDANT Protocol: Cooperation Of Nodes—Fairnes. In: Distributed Ad hoc NeTworks. In: Proceeding ACM Workshop Mobile Ad Hoc Networking and Computing, pp. 226–236 (2002, 2006)
4. Bansal, S., Baker, M.: Observation-based cooperation enforcement in ad hoc networks, Research Report cs. NI/0307012, Stanford University
5. Pirzada, A.A., Datta, A., McDonald, C.: Propagating trust in ad-hoc networks for reliable routing. In: Proceeding of IEEE International Workshop Wireless Ad Hoc Networks, pp. 58–62 (2004)
6. Pirzada, A.A., Datta, A., McDonald, C.: Trust-based routing for ad-hoc wireless networks. In: Proceeding of. IEEE International Conference Networks, pp. 326–330 (2004)
7. Park, S., Al-Shurman, M., Yoo, S.: Black Hole Attack in Mobile Ad hoc Network. In: ACMSE 2004, Huntsville, AL, USA (2004)
8. Patcha, A., Mishra, A.: Collaborative security architecture for black hole attack prevention in mobile ad hoc networks. In: Proceedings of the 2003 IEEE Radio and Wireless Conference, Boston, MA, USA, pp. 75–78 (2003)
9. Back, A.: Hashcash: A Denial of Service Counter-Measure (2002), `http://www.hashcash.org`
10. Johnson, D.B., Maltz, D.A., Hu, Y.C., Jetcheva, J.G.: The dynamic source routing protocol for mobile ad hoc networks (DSR). Internet draft IETFRFC 3561 (2003), `http://www.ietf.org/rfc/rfc3561.txt`
11. Dijkstra, E.W.: A note on two problems in connection with graphs. Numer. Math., 83–89 (1959)
12. QUALNET simulator, `http://www.scalable-networks.com`
13. IEEE Computer Society LAN MAN Standards Committee, Wireless LANMedium Access Protocol (MAC) and Physical Layer (PHY) Specification,IEEE Std 802.11-1997. The Institute of Electrical and Electronics Engineers, New York, NY (1997)

Sentiment and Mood Analysis of Weblogs Using POS Tagging Based Approach

Vivek Kumar Singh, Mousumi Mukherjee, and Ghanshyam Kumar Mehta

Department of Computer Science, Banaras Hindu University, Varanasi-221005, India
vivek@bhu.ac.in, mou.sonai@gmail.com, ghanshyam4u2000@gmail.com

Abstract. This paper presents our experimental work on analysis of sentiments and mood from a large number of Weblogs (blog posts) on two interesting topics namely ‘Women’s Reservation in India’ and ‘Regionalism’. The experimental work involves transforming the collected blog data into vector space representation, doing Parts of Speech Tagging to extract opinionated words and then applying semantic orientation approach based SO-PMI-IR algorithm for mining the sentiment and mood information contained in the blog text. We obtained interesting results, which have been successfully evaluated for correctness through both manual tagging and by cross-validating the outcomes with other machine learning techniques. The results demonstrate that these analytical schemes can be successfully used for blog post analysis in addition to the review texts. The paper concludes with a short discussion of relevance of the work and its applied perspective.

Keywords: Blogosphere, Mood Analysis, Opinion Mining, POS Tagging, Sentiment Analysis.

1 Introduction

Opinion Mining and Sentiment Analysis has traditionally been an area of exploration by Linguists. However, developments in Information Retrieval, Computational Linguistics and Statistical Text Processing during the last decade have turned it into an interdisciplinary area. It is now a more attractive area of research with noticeable advances in a very short span of time. The transformation of the World Wide Web into a new more participative web (often termed as Web 2.0) has led to a manifold increase in its value. In today’s Web users are no longer only consumers but they have become co-creators. Users are now generating content on the Web in a variety of forms such as blogs, reviews, forums, wikis etc. The huge amount of user posted data has attracted people and companies to exploit it for useful & productive purposes. However, the unstructured nature of data presents computational challenges that require sophisticated search and mining techniques.

During the last few years there have been interesting works on sentiment analysis of the user posted data on the Web. Experiments to identify sentiments have been performed with product reviews, movie reviews, student feedback, user posts about various products & services and prospects of candidates in elections etc. Broadly two

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 313–324, 2011.

approaches have been used by researchers: (a) classifying documents into positive and negative sentiment categories using sophisticated text classifiers; and (b) using an unsupervised semantic orientation computation from selected parts of speech tags. The goal in both the approaches is to mine the opinions and classify them as positive or negative (or favorable and unfavorable). A related goal to sentiment analysis is to identify the mood of the writer through mood analysis.

The opinion mining and sentiment analysis work can have very useful applications. For example, a person planning to go on a vacation at a particular place or stay in a particular hotel there can browse through the reviews about that place and if the majority opinion is negative, he may plan for another destination. Similarly, a person planning to buy a particular product (say a digital camera) may search for user reviews about it and in the event of majority opinion being negative; he may decide not to buy it. Opinion analysis is useful not only for end users but may also be used by companies to know the performance of a product, or by advertisers to decide where to place their advertisements. A page expressing a positive sentiment about a product could be a good choice to place the advertisement of the product, whereas a page having negative reviews may be a good page to place the advertisement of its competitor. Taking this a level further, a semantic orientation feature can be added to search engines to further filter/ group their search results into positive and negative categories. A user can thus quickly find the positive and negative aspects of a topic/ product easily.

Weblogs (also termed Blogs) provide an important platform for users to express themselves. The ease of creating blog posts, low barrier to publication, open standards of content generation and the free-form writing style allow large number of people to create their own blogs or post their contributions on community blog sites. People express their opinions, ideas, experiences, thoughts, and wishes in blog posts. The Blogosphere (universe of all blog sites) is now a huge source of user posted data. According to a statistics by a blog tracking company, Technorati, there were more than 112 million blogs as of September 2008 [1], with two blog posts created per second. The tremendous amount of valuable information contained in blogs has attracted the attention of people in academics as well in industry. Whether it be the much talked about strategy- to get early feedback about the newly released operating system Vista by Microsoft- by contacting influential bloggers in early 2007 and persuading them to share their experiences in return for Laptops as free gifts, or tacit marketing strategies by a number of companies; blogosphere is now a widely acknowledged platform for commercial exploitation as well.

In this paper, we have presented our work on sentiment and mood analysis on two blog data sets. The first data set is comprised of a good number of blog posts about ‘Women’s Reservation in India’. The second data set is comprised of blog posts on “Regionalism’. Both the data sets collected include only recent and moderately sized posts. We chose the topics due to the highly opinionated and emotion laden nature of the prospective posts. We have employed Parts Of Speech tagging (POS tagging) based approach to identify opinionated words and classified them as having positive and negative semantic orientations. The aggregated set of semantic orientations of the relevant POS tags in a post is then used to classify the entire post as positive or negative. We also performed both sentiment and mood analysis on the collected blog data. The rest of the paper is organized as follows. Section 2 of the paper describes the text processing infrastructure built for collecting, pre-processing and representing the blog post data. Section 3 discusses the Sentiment Analysis approach used. Section 4 summarizes the mood analysis task performed, followed by a snapshot of the results in Section 5. The paper concludes with a short discussion (Section 6) of the experimental work and the relevance & usefulness of this kind of experimental work.

2 Building Text Processing Infrastructure

Since the blog data is textual in nature, we need an appropriate text processing infrastructure to perform the analysis. The collected blog data often needs to be pre-processed, both to make the analytical task easier and to reduce the memory & computing requirements of the algorithm. The processed data is then represented into a vector space model with every document being represented as a collection of distinct terms. Once the documents are represented appropriately, the analysis task becomes domain spec?fic. If the goal is to identify opinionated words for further determination of semantic orientation, parts of speech tagging is performed. On the other hand, if the goal is to classify or cluster the document using machine learning, then term-document matrix representation comprising of *tf-idf* measures of terms is to be built.

2.1 Collecting Blog Data

Searching for relevant blog posts on a topic has been made simple by availability of several blog tracking companies. These blog tracking companies provide free tracking service which can be used by a blog search program to find high authority score data. One thus needs to devise a blog search program which can accept user queries and send it to a blog tracking provider through HTTP Get/Post. The blog search program will translate the query into a format acceptable to the blog tracking provider before sending. The blog tracking provider processes the query and sends back a response, usually in XML. The XML response received back is then parsed by the blog search program and the retrieved results are displayed. We collected full blogs from Google Blog Search [2] through a Java program. The collected blog data was stored as separate text files. Every blog data entry comprised of name of the blog site, permalink of the blog post, author’s name, title of the blog post, its text and user comments.

2.2 Pre-processing the Data

The collected blog text data is transformed into a term vector structure with frequency of occurrence of different terms. Tokenization is the first step towards this end and involves identifying terms contained in the document. Tokenization also includes managing hyphens and sometimes converting the tokens into lowercase as well. Many text analysis tasks require removal of stop words (such as to, or, and, the, are, their) etc. before subjecting the data to further phases of analysis. Though the inverse document frequency (idf) measure, described ahead, reduces the weight of widely occurring terms (such as stop words), but removal of stop words nevertheless reduces the time and space complexity requirements. Another technique, stemming, which removes occurrences of the same word in multiple forms (for example ‘computing’, ‘computer’, ‘computes’ etc. reduced to ‘comput’); may also be useful in some applications. Multi-word phrases are often important and convey exact and relevant meaning and therefore few applications takes special care to preserve them. We have done stop word removal but no stemming and synonym injection has been done.

2.3 Vector Space Model Representation

We have used the Vector Space Model [3], [4] to represent each document. Every blog post is represented in the form of a term vector. A term vector consists of the distinct terms appearing in a blog post and their relative weights. There are a number of ways to represent the term vectors. Commonly used ones are *tf*, *tf.idf* and *Boolean presence*. Term Frequency (*tf*) is a count of how often a term appears within the document. If the total number of documents of interest be *N*, and df_t be the number of documents that contain a term *t*, then the *idf* for the term *t* is computed as $\text{idf}_t = \log (\text{N}/\text{df}_t)$. The *idf* of a rare term is high, whereas the *idf* of a frequent term is likely to be low. The *tf* and *idf* values are used to produce a composite weight for each term in each document (say a term t in document d), defined as $\text{tf-idf}_{t,d} = \text{tf}_{t,d} \text{ X } \text{idf}_t$. The vector V(d) derived from the document *d* thus contain one component for each distinct term. Boolean presence uses ‘0’ and ‘1’ values to represent absence and presence of a term in a document vector. Once we have the entire set of posts represented as document vectors, their degree of similarity can also be computed using *cosine similarity* measure as in equation 1 below.

$$\text{Cosine Similarity } (d_1, d_2) = \{V(d_1).V(d_2)\} \,/\, \{|V(d_1)||V(d_2)|\} \qquad (1)$$

The numerator represents the dot product of the vectors $V(d_1)$ and $V(d_2)$, and the denominator is product of their *Euclidean lengths*. The denominator thus length-normalizes the vectors $V(d_1)$ and $V(d_2)$ to unit vectors $v(d_1) = V(d_1) / |V(d_1)|$ and $v(d_2) = V(d_2) / |V(d_2)|$ respectively. The Cosine Similarity measure is used for clustering and classification tasks of text documents.

2.4 Parts of Speech Tagging

POS tagging refers to assigning a linguistic category (often termed as POS tag) to every term in the document based on its syntactic and morphological behavior. Common POS categories in English language are: noun, verb, adjective, adverb, pronoun, preposition, conjunction and interjection. There are other categories also that arise from different forms of these categories, such as a verb can be in its base form or in its past tense. We have used Penn Treebank POS Tags [5] as shown in the Table 1 below. Identifying POS tags is the first step in term based semantic orientation computation. Most of the opinion mining tasks employing this approach, extract phrases containing adjectives or adverbs, as they are good indicators of subjectivity and opinions. Different experiments have used extracting two to four consecutive words satisfying a particular pattern for opinion mining tasks. We have extracted two consecutive words provided their POS tags conform to any of the patterns given in Table 2 below. For example, the pattern in line 2 means that two consecutive words are extracted if the first word is an adverb and the second word is an adjective, but the third word (not extracted) cannot be a noun.

Table 1. Penn Treebank Parts of Speech Tags (excluding punctuations)

Tag	Description	Tag	Description
CC	Coordinating Conjunction	PRP$	Possessive pronoun
CD	Cardinal Number	RB	Adverb
DT	Determiner	RBR	Adverb, comparative
EX	Existential there	RBS	Adverb, superlative
FW	Foreign word	RP	Particle
IN	Preposition or subordinating conjunction	SYM	Symbol
JJ	Adjective	TO	To
JJR	Adjective, Comparative	UH	Interjection
JJS	Adjective, Superlative	VB	Verb, base form
LS	List item marker	VBD	Verb, past tense
MD	Modal	VBG	Verb, gerund or present participle
NN	Noun, singular or mass	VBN	Verb, past participle
NNS	Noun, plural	VBP	Verb, non-3rd person singular present
NNP	Proper noun, singular	VBZ	Verb, 3rd person singular present
NNPS	Proper noun, plural	WDT	Wh-determiner
PDT	Predeterminer	WP	Wh-pronoun
POS	Possessive ending	WP$	Possessive wh-pronoun
PRP	Personal pronoun	WRB	Wh-adverb

Table 2. Patterns of tags for extracting two-word phrases

	First Word	Second Word	Third Word (Not Extracted)
1	JJ	NN or NNS	anything
2	RB, RBR or RBS	JJ	not NN nor NNS
3	JJ	JJ	not NN nor NNS
4	NN or NNS	JJ	not NN nor NNS
5	RB, RBR or RBS	VB, VBD, VBN or VBG	anything

3 Sentiment Analysis

The sentiment analysis problem can be formally defined as follows: Given a set of documents D, a sentiment classifier classifies each document $d \epsilon D$ into one of the two classes, **positive** and **negative**. Positive means that *d* expresses a positive opinion and negative means that *d* expresses a negative opinion. There have been numerous approaches to opinion mining and sentiment analysis in texts such as movie reviews, product ratings and political campaigns in the past. Most of the experiments performed so far however employed one of the following two approaches: (a) using a text classifier (such as Naïve Bayes, SVM or kNN) that takes a machine learning approach to categorize the documents in positive and negative groups; and (b) computing semantic orientation of documents based on aggregated semantic orientation values of selected opinionated POS tags extracted from the document. Some of the past works on sentiment analysis have also attempted to determine the strengths of positive and negative orientations. Few prominent researches works on opinion mining and sentiment analysis around these themes can be found in [6], [7], [8], [9], [10], [11], [12] & [13]. We have classified the blog posts using the semantic orientation scheme that employs POS tag extraction. The results have been verified using a machine learning classifier.

3.1 Semantic Orientation Approach

In semantic orientation approach we first extract bi-grams (consecutive two words) that conform to the pattern described in Table 2. Thereafter the semantic orientation of extracted phrases is computed using the **Pointwise Mutual Information** (PMI) measure given in equation 2 below,

$$\text{PMI}(\text{term}_1, \text{term}_2) = \log_2 \{ \Pr(\text{term}_1 \blacktriangle \text{term}_2) / \Pr(\text{term}_1).\Pr(\text{term}_2) \} \quad (2)$$

where, $\Pr(\text{term}_1 \blacktriangle \text{term}_2)$ is the co-occurrence probability of term_1 and term_2 and $\Pr(\text{term}_1).\Pr(\text{term}_2)$ gives the probability that two terms co-occur if they are statistically independent. The ratio between $\Pr(\text{term}_1 \blacktriangle \text{term}_2)$ and $\Pr(\text{term}_1).\Pr(\text{term}_2)$ measures the degree of statistical independence between them. The log of this ratio is the amount of information that we acquire about the presence of one word when we observe the other. The semantic orientation (SO) of a phrase can thus be computed by using the equation 3,

$$\text{SO (phrase)} = \text{PMI (phrase, “excellent”)} - \text{PMI (phrase, “poor”)} \quad (3)$$

where, PMI (phrase, “excellent”) measures the association of the phrase with positive reference word “excellent” and PMI (phrase, “poor”) measures the association of phrase with negative word “poor”. These probabilities are calculated by issuing search query of the form “phrase NEAR Excellent” and “phrase NEAR poor” to search engine. The number of hits obtained is used as a measure of probability value. The SO value for all the extracted bi-grams is computed using this scheme. To determine the semantic orientation of the entire document, the SO values of the opinionated phrases in it is aggregated, and if the average SO is a positive value (or above a threshold) the document is labeled as positive, and negative otherwise. This algorithm is often referred to as SO-PMI-IR. A variation of this scheme is SO-PMI-LSA [14], which uses Latent Semantic Analysis. In this scheme, the term-document matrix is first reduced using Singular Value Decomposition (SVD) and then LSA(word_1, word_2) is computed by measuring the cosine similarity (as described in Eq. 1) of the compressed row vectors corresponding to word_1 and word_2. Thereafter, the semantic orientation of a word is computed by equation 4,

$$\text{SO (word)} = \text{LSA (word, \{positive terms\})} - \text{LSA (word, \{Negative terms\})} \quad (4)$$

where, positive terms refer to words like ‘good’, ‘superior’, ‘excellent’ etc. and negative terms refer to words like ‘bad’, ‘poor’, ‘inferior’ etc. The LSA of a word is computed with term vectors of positive and negative words occurring in the document set. Experimental results [14] have shown that SO-PMI-IR and SO-LSA have approximately the same accuracy on large dataset.

3.2 Machine Learning Approach

Another scheme, a machine learning approach, which has been used in the past for sentiment analysis is the SentiWordNet based approach [15]. It computes three numerical scores Obj(s), Pos(s) and Neg(s), describing how objective, positive and negative the terms contained in the synset are. The method used in SentiWordNet is based on qualitative analysis of the glosses associated to synsets, and on the use of resulting vectorial term representations for semi-supervised synset classification. The Obj(s), Pos(s) and Neg(s) scores for a synset s are derived by a committee of eight ternary classifiers. The scores of a synset s are determined by the normalized proportion of ternary classifiers that have assigned the corresponding labels to it. These scores associate PN polarity with every term of interest. It is important to note that sum of the three scores for a synset s is always 1. Once the scores of all POS tags of interest for a document are obtained, the aggregate score is used to compute objective, positive and negative scores of the whole document and label them accordingly. Naïve Bayes and Support Vector Machine (SVM) text classification schemes have also been used for sentiment analysis. We have used the machine learning approach to verify the accuracy of our semantic orientation approach.

3.3 Advantages and Limitations

The main advantage of sentiment analysis is that it provides the prevailing opinion about an object, topic or event. One can know the prevailing/ majority opinion about a topic without going through the text of a large number of documents on that topic. Several interesting applications, such as a movie review system, a product rating system and even a search engine for a particular aspect of a topic may be designed by incorporating sentiment analysis approach. The document level sentiment analysis however has several limitations. First is that it simply attributes a document as positive and negative and does not tell about the positive and negative aspects. Second limitation is that it may not produce good accuracy for non-review documents. Further, the sentiment analysis process itself has inherent limitation. It is not always easy to extract subjective/ opinionated phrases to be used for POS tagging based analysis. And the same is true for supervised text classifiers, as they require a good amount of training data before they can produce correct classification.

4 Mood and Gender Analysis

The task of classifying blog posts by mood involve predicting the most likely state of mind with which the post was written i.e., whether the author was depressed, cheerful, bored, upset etc. Like sentiment classification, mood classification may also be used for productive purposes such as filtering results of a search engine by mood, identifying communities and possibly even to assist behavioral scientists in behavioral research and training. A similar classification goal of categorizing the blog posts is to perform gender analysis based on their authorship, i.e. 'male' and 'female'. Both the tasks of mood and gender classification focus on stylistic features [16]. Stylometric research particularly that concerned with emotion and mood analysis in text is becoming more common nowadays. This may be primarily due to availability of subjective information on the Web. Most of the past research on mood analysis used style-related as well as topic-related features in the text for identifying the mood of the author. Usually 'bag of words' or POS tags are extracted along with their frequency counts for subsequent use in mood classification. There has been several interesting works on mood analysis [17], [18]. In an important experiment with mood classification in blog posts [19], Mishne used a Mood PMI-IR based approach. This approach is conceptually similar to SO-PMI-IR scheme. However, instead of using "excellent" and "poor" as reference terms, he used terms corresponding to various moods (such great, annoyed, cheerful, sleepy etc.) for calculating PMI measure.

Gender classification also uses extracted POS tags to attributed label to a blog post. In addition to POS tags certain other stylistic features, such as patterns of certain words have also been experimented with. However, work on gender analysis is still far from being accurate. We have used uClassify [20] to classify blogger's mood and gender. uClassify, as a online mood analysis system associates with each blog post happy and upset scores corresponding to the their moods. As a gender analyzer, it associates with each post a label of either 'male' or 'female' corresponding to the usage patterns of certain POS tags. We did mood analysis for both the blog data sets on 'Women's reservation in India' and on 'Regionalism'. Gender analysis has been performed on blog posts related to 'Women's reservation in India'.

5 Experimental Results

We have performed sentiment classification, mood analysis and gender association experiment on two large blog data sets. As discussed earlier, first data set comprised of moderately sized blog posts on 'Women's reservation in India' and the second on posts about 'Regionalism'. Though we were not selective in choosing posts on 'Regionalism' with Indian perspective only, but we got good number of posts in the context of India. We have used both the raw and pre-processed blog data for the analysis. The collected data was transformed into vector space model representation and POS tag labeling was done for different terms in the blog data. In order to evaluate PMI values for various bi-grams, we first extracted all the relevant tags in the entire dataset and cached the results of offline queries issued for computing PMI and SO values. We have used aggregated semantic orientation values of the terms contained in a post to classify it as positive or negative. This aggregation was done in two ways. In one scheme we associated '+1' and '-1' values for every positive and negative reference word. Here positive and negative is determined by setting a threshold for the SO value of the term. If the SO value of a term is greater than the chosen threshold, it gets '+1' score and '-1' otherwise. In the other scheme we simply added the SO values of all the extracted terms in a blog post and then divided the sum by the total number of terms extracted. For example, if the sum of SO values of n terms is x, then the aggregate SO value of that blog post is x/n. This aggregate value is then compared with a threshold to label a blog post as positive or negative. We also performed Mood and Gender analysis tasks, to identify blogger's mood and gender information. Tables 3 & 4 show a snapshot of semantic orientation, mood and gender analysis task results for a small subset of blog data on 'Women's Reservation in India' and 'Regionalism', respectively. Table 5 presents a summary of accuracy of a subset of 50 blogs of both the datasets.

Table 3. A snapshot of results for a subset of the first data set. The thresholds for two SO aggregation schemes are 0 and 0.70 respectively.

Title of the Blog	Mood	Gender	Aggregate SO Value	Semantic Orientation
Do women need reservation?	happy (70.1 %) upset (29.9 %)	female (74.8 %) male (25.2 %)	+3	Positive
			0.76653093	Positive
Reservation for Women: The icing on the cake	happy (54.7 %) upset (45.3 %)	female (52.8 %) male (47.2 %)	-7	Negative
			0.73098755	Positive
Woman's Quota ~another step to take India back	happy (63.1 %) upset (36.9 %)	female (51.4 %) male (48.6 %)	-2	Negative
			0.83751	Positive
No pride in Women's Reservation Bill	upset (54.9 %) happy (45.1 %)	male (62.1 %) female (37.9 %)	-1	Negative
			0.95041317	Positive
Reservation to power for Indian women	happy (76.8 %) upset (23.2 %)	female (64.9 %) male (35.1 %)	+4	Positive
			0.91197723	Positive
Reservation by custom and tradition is acceptable	happy (50.1 %) upset (49.9 %)	female (65.0 %) male (35.0 %)	+12	Positive
			0.9886512	Positive
Women's bill should lead on to real political reform	upset (81.4 %) happy (18.6 %)	male (80.3 %) female (19.7 %)	-6	Negative
			0.6081681	Negative

Table 4. A snapshot of results for a subset of the second data set. The thresholds for two SO aggregation schemes are 0 and 0.70 respectively.

Title of the Blog	Mood	Aggregate SO Value	Semantic Orientation
Is Your Region More Important than your Nation?	upset (66.8 %) happy (33.2 %)	1	Positive
		0.76783	Positive
Is it legitimate to give J&K the status of a special state?	upset (81.2 %) happy (18.8 %)	-2	Negative
		0.68771	Negative
The whole of India does not belong to all Indians	upset (75.9 %) happy (24.1 %)	-3	Negative
		0.62420	Negative
Do we need to fear Regionalism?	upset (93.9 %) happy (6.1 %)	+2	Positive
		0.68932	Negative
Regionalism takes over from religion in India	upset (85.1 %) happy (14.9 %)	+5	Positive
		0.95209	Positive
Freedom from Regionalism	happy (54.2 %) upset (45.8 %)	+15	Positive
		1.20138	Positive
A Sense of where you are!	happy (74.1 %) upset (25.9 %)	+10	Positive
		1.09861	Positive

Table 5. Classification accuracy of a sample subset of 50 blog posts of both the datasets

50 Blog Data subset on	**Classification accuracy using first SO aggregation scheme**	**Classification accuracy using second SO aggregation scheme**	**Classification accuracy using Naïve Bayes Machine Learning approach**
Women's Reservation in India	71%	74%	69%
Regionalism	67%	69%	67%

The entries in the last two columns of the tables 3 and 4 present the aggregate SO values obtained using the two schemes described above. It is clear that none of the two SO aggregation schemes are fail proof. In some cases first scheme performs better whereas in others the second performs better. The results of semantic classification achieved a reasonable degree of accuracy vis-à-vis manual labeling, as well as the machine learning classification approach implemented. Relatively better accuracy is obtained for the first dataset, primarily due to more subjectivity present in the data. One interesting thing to see was that the classification of a blog post as 'positive' sentiment holder does not necessarily mean that it is also assigned higher 'happy' score in mood analysis. Though a very small proportion, but some of the blog posts classified as 'positive' have relatively higher scores on 'upset' mood. This may be due to use of different reference words in semantic and mood classification (excellent & poor in sentiment classification and mood-related words like happy, cheerful, annoyed, sleepy etc. in mood classification). Similarly, gender analysis of blog posts of the first data set also resulted in few misclassifications vis-à-vis author information stored with blog entries. This was primarily due to the topic of analysis. Since all blog posts talk about women's reservation, there is a high degree of use of terms & pronouns belonging to female gender class. Overall the results obtained achieve a reasonable degree of accuracy.

6 Discussion

The sentiment and mood classification experiments performed obtained interesting results and also achieved a reasonable degree of accuracy. The accuracy of results was verified through both manual labeling and by comparing outcomes of multiple techniques implemented using many different ways. The results of sentiment and mood analysis were compared with results of machine learning, manual labeling and also with each other. The sentiment and mood labels did not agree in all the cases but there was a high degree of similarity of classifications done both by semantic orientation approach and the machine learning approach implementations. They both performed almost to the same level of accuracy vis-à-vis manual labeling. The results obtained are important from the point of view of demonstrating the applicability of semantic orientation approach on non-review data. It is often stated that the POS tagging based semantic orientation and mood analysis approach work well only with reviews and highly subjective data of sufficiently long size. However, the results clearly establish that it can be successfully applied to sentiment and mood classification of weblogs as well. We have verified the results with a naïve bayes machine learning based classifier. Verification using SVM is yet to be done.

Meanwhile, it would be very much in order to evaluate the applicability and usefulness of the semantic orientation based approach to sentiment and mood classification. Unlike machine learning approach which requires a good amount of training data and sufficient time to train the classifier; POS tag based semantic orientation approach is a purely unsupervised approach to sentiment classification. It has been used earlier with high accuracy for review kind of data. However, the current experiment and few others analytical experiments on non-review data have demonstrated that it can be used with diverse kinds of data. The only requirement is that the data should contain sufficient subjectivity. POS tag based sentiment classification approach is therefore an advisable approach (and perhaps the only approach if training set is difficult to build) for sentiment classification.

Sentiment and mood classification techniques are of immense value and can have very important and productive applications. The applications may range from (a) going through sentiment classified data about a product before deciding to buy it; (b) evaluation of sentiment and mood information of a movie review by a movie recommendation engine before recommending it to a user or possibly improve the recommendations obtained through a traditional collaborative filtering scheme; (c) use of sentiment classification data by advertisers to decide which product and where to place its advertisement; (d) designing an improved search engine by using sentiment analysis as a filter for the retrieved results to classify them into separate categories; to (e) using opinion mining and sentiment classification for sociological experiments on the new social Web [21]. Sentiment and mood classification techniques have just begun to demonstrate their usefulness.

References

1. Technorati Blogosphere Statistics, `http://technorati.com/blogging/state-of-the-blogosphere/`
2. Google Blog Search, `http://google.com/help/about_blogsearch.html`
3. Alag, S.: Collective Intelligence in Action. Manning, New York (2009)
4. Manning, C.D., Raghavan, P., Schutze, H.: Introduction to Information Retrieval. Cambridge University Press, New York (2008)
5. Penn Treebank Project, `http://www.cis.upenn.edu/~treebank/home.html`
6. Dave, K., Lawerence, S., Pennock, D.: Mining the Peanut Gallery-Opinion Extraction and Semantic Classification of Product Reviews. In: 12th International World Wide Web Conference, pp. 519–528. ACM Press, New York (2003)
7. Turney, P.: Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews. In: ACL 2002, 40th Annual Meeting of the Association for Computational Linguistics, Philadelphia, US, pp. 417–424 (2002)
8. Esuli, A., Sebastiani, F.: Determing the Semantic Orientation of Terms Through Gloss Analysis. In: CIKM 2005, 14th ACM International Conference on Information and Knowledge Management, Bremen, DE, pp. 617–624 (2005)

9. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up? Sentiment Classification using Machine Learning Techniques. In: Conference on Empirical Methods in Natural Language Processing, Philadelphia, US, pp. 79–86 (2002)
10. Kim., S.M., Hovy, E.: Determining Sentiment of Opinions. In: COLING Conference, Geneva (2004)
11. Durant, K.T., Smith, M.D.: Mining Sentiment Classification from Political Web Logs. In: WebKDD 2006, ACM Press, New York (2006)
12. Liu, B.: Web Data Mining: Exploring Hyperlinks, Contents and Usage Data. Springer, Berlin (2008)
13. SentiWordNet, `http://patty.isti.cnr.it/~esuli/software/SentiWordNet`
14. Turney, P., Littman, M.L.: Unsupervised Learning of Semantic Orientation from a Hundred-Billion-Word corpus. NRC Publications Archive (2002)
15. Esuli, A., Sebastiani, F.: SentiWordNet: A Publicly available Lexical Resource for Opinion Mining. In: Fifth Conference on Language Resources and Evaluation (LREC 2006), Geneva (2006)
16. Sebastiani, F.: Machine Learning in Automated Text Categorization. ACM Computing Surveys 34(1), 1–47 (2002)
17. Mishne, G., Rijke, M.D.: MoodViews: Tools for Blog Mood Analysis. In: AAAI 2006 Spring Symposium on Computational Approaches to Analyzing Weblogs (AAAI-CAAW 2006) (March 2006)
18. Balog, K., Rijke, M.D.: Decomposing Bloggers' Moods. In: 3rd Annual Workshop on the Web blogging Ecosystem, At WWW 2006 (2006)
19. Mishne, G.: Experiments with Mood Classification in Blog Posts. In: 2005 Stylistic Analysis of Text for Information Access Conference (2005)
20. Uclassify Mood Analysis Tool, `http://www.uclassify.com/browse/prfekt/Mood (retrieved April 2009)`
21. Singh, V.K.: Mining the blogosphere for sociological inferences. In: Ranka, S., Banerjee, A., Biswas, K.K., Dua, S., Mishra, P., Moona, R., Poon, S.-H., Wang, C.-L. (eds.) IC3 2010. CCIS, vol. 94, pp. 547–558. Springer, Heidelberg (2010)

Search Results Optimization

Divakar Yadav, Apeksha Singh, and Vinita Jain

Jaypee Institute of Information Technology,
A-10, Sector-62, Noida (India)
dsy99@rediffmail.com, apeksha0701@gmail.com,
vinita7103517cse@gmail.com

Abstract. In this paper, we put forward a technique for optimization of the search results obtained in response to an end user's query. With the enormous volume of data present on the web, it is relatively easy to find matched documents containing the given query terms. The difficult part is to select the best from the possible myriad of matching pages. Moreover, most Web search engines perform very well for a single keyword query but fail to do so in case of multiple terms. In this paper by using the concept of Meta search engines we propose a suitable query processing and optimization algorithm for giving the best possible result for multiple term keywords in the ranked order.

Keywords: Search Optimization, Query processing, Re-ranking, tf-idf, inverted indexing.

1 Introduction

The simple concept of search results optimization is to pick precisely the matched documents and produce it to the end user in the order of relevance with respect to the user query. It aims at giving the best suitable optimized results according to the users query, benefiting them in the structure of time and ease. According to [1] optimization involves comparison of a variety of parameters such as keyword density, link analysis, anchor text content, Meta tags and so on. A web crawler, which takes a seed URL and crawls the web pages accordingly, is used to retrieve the web pages [2]. Re-ranking involves calculation of the overall scores of the documents based on their weighing schemes and positions them in the order of their ranks [5].

The aim in this work is to develop a suitable weighing scheme by taking care of the factors mentioned in [1] which help to optimize the matched web pages when multiple terms query is provided as search input by the users. While optimizing the result various other factors such as page structure, the frequency of keywords and their position in a document are also to be considered.

The paper is organized in following ways: In the current section we discussed about the introduction of search results optimization and the key problems. Section 2 describes the research work that has been already been done in the field. Various research papers have been studied and examined for this purpose. Section 3 contains the algorithm that has been proposed for multi keyword based search results optimization. Implementation and results have been discussed in section 4. Finally in

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 325–334, 2011.

section 5 we have concluded the work with some future aspects followed by references of section 6 which have been the source of our knowledge.

2 Literature Survey

Extensive research and study has been done in the field of search results optimization in the past. Various research papers have been studied by us for the purpose of gaining knowledge in the field which motivated us to begin the work on this topic. Summarization of few selected research papers helpful on the topic is given hereafter.

The impact of search engine ranking and result optimization for the websites has been discussed in [10]. The fundamental methods involved in the optimization of the websites have been described in [1]. It talks about the relevance of Page Title in deciding the rank of web pages. Keyword Optimization and Link Analysis have also been stated as important criteria for making a web page search engine friendly and optimized.

The page rank algorithm given by Google founder which is based on link structure is discussed in [2]. It also describes the initial architecture behind the Google search engine. This paper helped us to know about the components and system structure involved in search engines, such as the web crawlers, repositories, web servers etc. A detailed explanation of the page ranking scheme has also been given in [6] and it describes the importance of the ranking scheme over citation counting method.

The inefficiency of web search engines for multiple term keywords has been discussed in [3]. It states that while most web search engines perform very well for a single-keyword query, their precisions is not so good for queries involving two or more keywords. This is primarily because the search results usually contain a large number of pages with weak relevance to the query keywords. It proposes a method to improve the precision of Web retrieval based on proximity and density of keyword for two-keyword queries.

The algorithm for better optimized search results has been discussed in [4]. It introduces a query processing scheme which forms the basis of our algorithm. Also a suitable ranking criterion, the weighing scheme and clustering method have been proposed. The re-ranking method which filters the unrelated documents by employing document comparison method, link extraction technique and by comparing the contents of the anchor text has been described in [5].

The various crawling algorithms, such as the shark algorithm etc have been discussed in [7], [8] while the inverted index scheme and its application have been discussed in detail in [9].

3 Proposed Algorithm

In this paper, we propose a new algorithm for optimization of the search results in response to a user's query. The major challenge while doing this was to devise an efficient method to handle multiple query terms and to incorporate the various optimization strategies together with it.

The retrieval of the web pages for the algorithm has been done by a web crawler which works on the principle of seed URL. For query processing we broke the query term into different subsets, processed it and calculated a probability factor which is associated with each of the subsets depending on their relevance, in ascending order. Further, we applied the optimization strategies such as Keyword density, Inverse Document Frequency (idf) and some others to calculate the scores of all the documents. Finally the documents are re-ranked based on their respective scores. Given below is the sequential description of the proposed technique.

1. Retrieve various web pages for the query term with the help of a WebCrawler from various search engines and store them.
2. For handling multiple query terms to ensure better optimization, every query is broken down into sets which contains all possible combinations of its terms such as single terms, double terms and so on, up to the maximum length of the original query. Afterward, a probability factor is attached with these sets which assign highest probability to the set having maximum term combinations. The sole reason to do this is based on the fact that web pages which contain most of the query terms would be more relevant as compared to those having just one or two terms.
3. Normalize the query terms. For this a weight is associated to all the sets of the query terms. In our case the probability becomes the weight. This weight serves as the main multiplying factors while calculating the ranks of the web pages. The other factors that have been taken into consideration are as follows:
   (a) Different weights have been assigned to the different tags according to their relative importance in a web page. For example if title and Meta tags are present in the header of the page they are considered more important than the other contents of the page and hence, they are assigned more weights. Therefore, the query term occurring in them will fetch the most score.
   (b) Another factor taken into account is the relative position of keywords in a web page. A web page which has the query term in the abstract or beginning of the web page will carry higher relevance than the pages which have the same query term towards the end in web page. Thus query terms found in different paragraphs will contain different scores based on different weights assigned to the different paragraphs.
   (c) Anchor tags have been assigned a weight lower than title and Meta tags. Score is calculated by calculating the frequency of all *n* grade query terms in anchor tags and multiplying the frequencies with the weight of respective terms.
   (d) Term frequency and inverse document frequency of the query terms is computed and tf-idf is also assigned a relevant weight. The inverse document frequency takes into account the anomaly which arises when certain pages are small while others are pretty long.

Based on the above factors a score for each web page is calculated using following formula.

Score (q, d) = frequency of the term (in each section)*weight (same section) ----- (1)

Incorporating the term frequency and inverse document frequency in above equation the final score is computed using the following formula.

Final Score =score (q, d) + tf-idf (q, d) ----------(2)

Where, q = query, d = document, tf-idf = term frequency * inverse document frequency.

Using the above formula the scores for each web document is computed and then they are arranged in their decreasing order of scores. The web page having the highest score is the most optimized result for a user's query.

4 Implementation and Results

We implemented a Meta search engine to optimize the web pages for a user's query. While implementing Meta search engine following procedures have been followed.

4.1 Retrieval of the Web Pages

As soon as the user enters the query term and hits the search button a web crawler crawls the search results of various existing search engines, retrieves the links of web pages and stores them in our database. For every keyword around 10-25 links are retrieved and stored randomly. Then the crawler iterates through the links and retrieves their corresponding web pages and stores it in the database.

4.2 Ranking Procedure

The web pages are then ranked on the basis of the following factors:

Instead of searching for the entire query entered by the user, for better optimization results, we broke the query term into a number of subsets. For example if a 5 word query is there then it will be broken down into sets containing all possible combinations of single terms, double terms and so on , till the length of the query.

E.g. Query term: I like icecream

Single phrases

- I
- Like
- Icecream

Double phrases

- I like
- Like icecream
- I icecream

Triple phrase

- I like icecream

The main reason for this was the fact that the pages that contain the entire query phrase are often more relevant than those containing single terms of the query.

The combinations of all the words are stored in 2D array for better access and also because for different sets different weights have to be assigned. For example, a query term "how to construct a triangle" will be broken down and all the combinations will be stored in array as shown in fig. 1.

Array of combinations of 5 words
Array ([0] => how to construct a triangle)

Array of combinations of 4 words
Array([0] => to construct a triangle [1] => how construct a triangle [2] => how to a triangle[3] => how to construct triangle[4] => how to construct a)

Array of combinations of 3 words
Array([0] => construct a triangle [1] => to a triangle [2] => to construct triangle [3] => to construct a [4] => how a triangle [5] => how construct triangle [6] => how construct a [7] => how to triangle [8] => how to a [9] => how to construct)

Array of combinations of 2 words
Array([0] => a triangle [1] => construct triangle [2] => construct a [3] => to triangle [4] => to a [5] => to construct [6] => how triangle [7] => how a [8] => how construct [9] => how to)

Array of combinations of 1 words
Array ([0] => triangle [1] => a [2] => construct [3] => to [4] => how)

Fig. 1. Combinations of query term phrases

4.3 Calculation of Relevancy Factor

The relevancy factor (probability) that has been assigned to each query set is given by

$$p(i) = p_i/p \qquad \text{----------------(3)}$$

Where pi is the number of permutations for the nth grade query term and p is the total number of permutations (summation of number of permutations for all the grades).

This probability 'p(i)' is the main multiplying factor while calculating the ranks. It serves as a kind of weight, thus having the highest value for the set containing maximum terms combinations and lowest for single terms combinations. This is in compliance with the fact that there is always more probability of existence of single word terms in a document. The frequency of occurrence for single term combinations in a document is higher in comparison to n term combinations. Hence least weight or relevance is assigned to single term combinations. On the same basis the set that has all the words of the query term will have the highest weight (as document containing the full query term will be always is more relevant). In this way score for more relevant page is higher even if it is shorter in length. This transfigures the probability factor into the relevancy factor for each query term subset.

To perform more accurate ranking of the document we even calculate the Inverse Document Frequency. This takes into account the problem that often documents vary in size from each other. The tf-idf which is calculated for each document thus gives us a rough idea about the relevancy of each of them.

$$tf_{i,j} = n_{i,j} / \sum_k n_{k,j}$$

$n_{i,j}$ is the number of occurrences of the considered term (t_i) in document d_j and denominator is the sum of number of occurrences of all terms in document d_j

$$idf_i = \log(|D| / |\{d : t_i \in d\}|)$$

D, is the total number of documents in the corpus $|\{d : t_i \in d\}|$, in which the term t_i appears.

Values of tf-idf are calculated for every query term subset that has been derived from query term during normalization using equation (4).

$$(tf\text{-}idf)_{i,j} = tf_{i,j} * idf \qquad \text{---------------(4)}$$

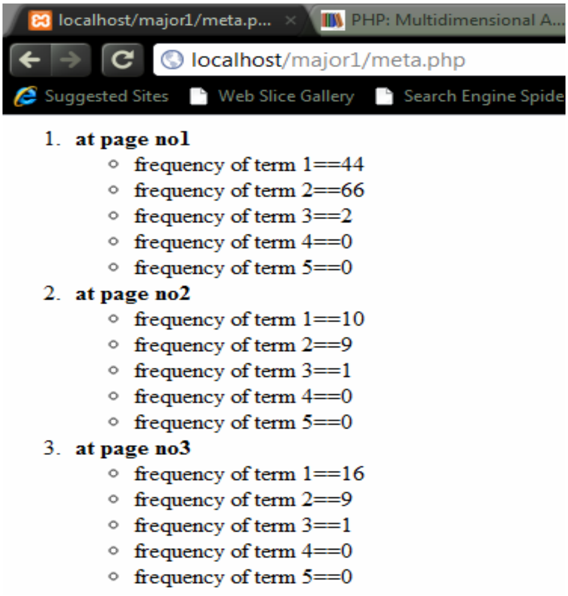
4.4 Location in a Document

The third step for optimization was to find the term frequency of each query term set in the different parts of the documents like in title, Meta tag, and anchor tag. The term frequency of the query terms have been calculated in all the tags and the respective tags are assigned weights.

Next, for each set of query term we multiply these weights with the relevancy factor calculated above as shown in table 1.

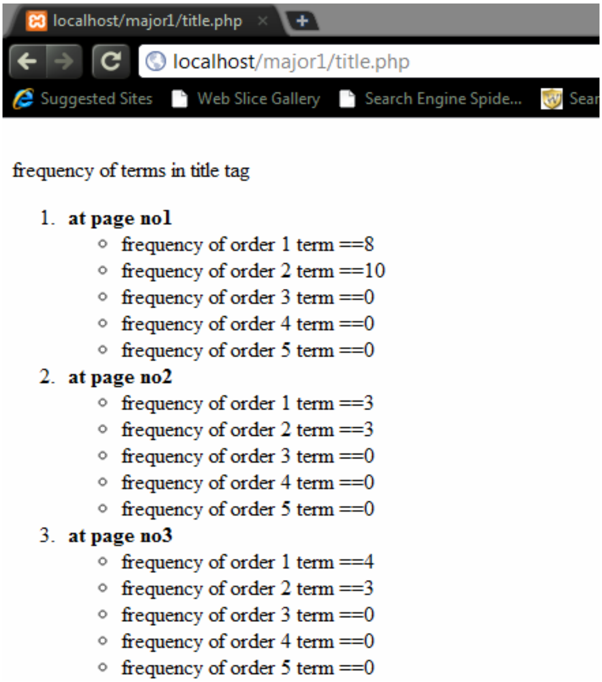
Table 1. Parameters and Weights

ID	ITEM	WEIGHT	RELEVANCY PARAMETER(λ)
1	Title tag	W1	$\Sigma F_{it} * p_i *$
2	Meta tag	W2	$\Sigma F_{im} * p_i *$
3	Anchor tag	W3	$\Sigma F_{ia} * p_i *$


localhost/major1/meta.p...
PHP: Multidimensional A...
localhost/major1/meta.php
Suggested Sites
Web Slice Gallery
Search Engine Spide

```
1. at page no1
   ○ frequency of term 1==44
   ○ frequency of term 2==66
   ○ frequency of term 3==2
   ○ frequency of term 4==0
   ○ frequency of term 5==0
2. at page no2
   ○ frequency of term 1==10
   ○ frequency of term 2==9
   ○ frequency of term 3==1
   ○ frequency of term 4==0
   ○ frequency of term 5==0
3. at page no3
   ○ frequency of term 1==16
   ○ frequency of term 2==9
   ○ frequency of term 3==1
   ○ frequency of term 4==0
   ○ frequency of term 5==0
```

Fig. 2. Keyword Frequency in Meta tags



```
frequency of terms in title tag

1. at page no1
   ○ frequency of order 1 term ==8
   ○ frequency of order 2 term ==10
   ○ frequency of order 3 term ==0
   ○ frequency of order 4 term ==0
   ○ frequency of order 5 term ==0
2. at page no2
   ○ frequency of order 1 term ==3
   ○ frequency of order 2 term ==3
   ○ frequency of order 3 term ==0
   ○ frequency of order 4 term ==0
   ○ frequency of order 5 term ==0
3. at page no3
   ○ frequency of order 1 term ==4
   ○ frequency of order 2 term ==3
   ○ frequency of order 3 term ==0
   ○ frequency of order 4 term ==0
   ○ frequency of order 5 term ==0
```

Fig. 3. Keyword Frequency in Title tags

4.5 Position in a Document

Another factor that has been kept in mind is the position of a particular query term inside a document, like the paragraph position. For example, the documents whose abstract or introduction contains the query term are often more relevant as compared to the documents in which query terms are appearing at the end.

Terms in paragraphs are assigned weights in the decreasing order according to the order of paragraph they are found in. The term frequency and scores are calculated for every subset of the query term.

Finally the relevancy factor calculated through the above ways is multiplied with the weights and the final score is decided by adding the tf-idf score to each of the documents as shown in equation (5).

$$\text{Score final} = \text{score}(q, d) + \text{tf-idf}(q, d) \qquad \text{----------(5)}$$

Where, q = query, d = document

4.6 Re-ranking

Scores of all the pages are calculated and arranged in the descending order and the pages are re-ranked.

4.7 Comparison Analysis

We perform the optimization algorithm on the query "taj mahal india" and calculated various optimization factors. The results obtained thereof are depicted in the following graphs (fig. 4).

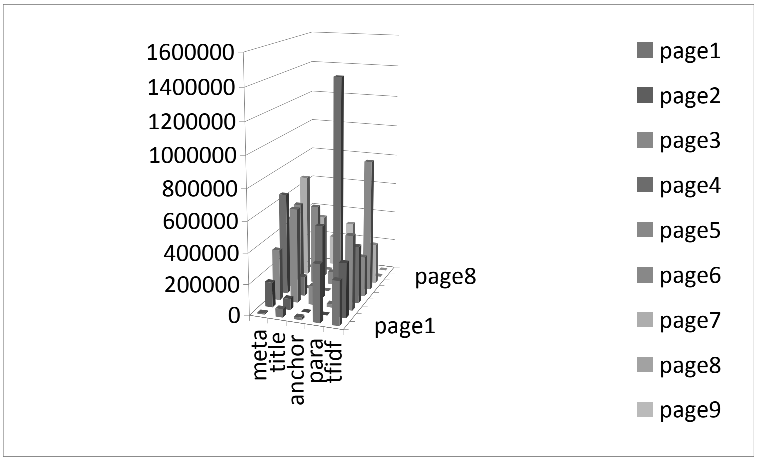


Fig. 4. Comparison of pages for each relevance factor

The above graph depicts the frequencies of the various tags in the pages obtained using yahoo search engine. Here page 1, 2 and so on are actually yahoo ranked pages 1, 2 etc.

According to this, the highest frequency accounts for page 4 and the second highest is of page 3. Fig. 5 is a pie chart representation of the final scores of all these pages based on our algorithm.

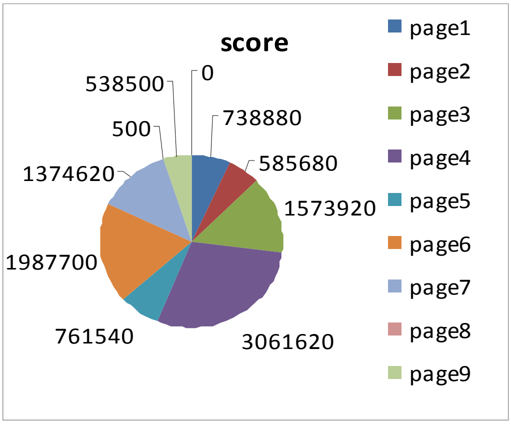


Fig. 5. Full score for every page

According to the results obtained by us the highest rank will be that of page 4 and second highest of page 3 and so on. Given below is a brief look on the first five ranks of our results.

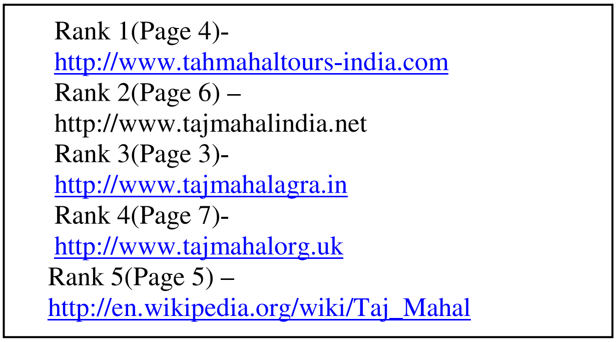

Rank 1(Page 4)-
http://www.tahmahaltours-india.com
Rank 2(Page 6) –
http://www.tajmahalindia.net
Rank 3(Page 3)-
http://www.tajmahalagra.in
Rank 4(Page 7)-
http://www.tajmahalorg.uk
Rank 5(Page 5) –
http://en.wikipedia.org/wiki/Taj_Mahal

Fig. 6. Yahoo links ranked by the implemented algorithm

In the bracket it is listed the yahoo ranking order on these pages. Our ranking order will change depending on the relevance of context in the page contents.

5 Conclusions and Future Work

The above stated algorithm has been successfully implemented. We took certain web pages based on the user's query and optimized the result on the basis of the proposed algorithm. Optimization of the search results can be taken a step further if the concept of semantic search, which takes a user's thought process into consideration is also taken into account. The future scope would be to merge the above slated algorithm and semantic search process to give the highest possible optimized search results.

References

1. Chengling, Z., Jiaojiao, L., Fengfeng, D.: Application and Research of SEO in the Development of Web2.0 Site. In: Second International Symposium on Knowledge Acquisition and Modeling, pp. 236–238 (2009)
2. Brin, S., Page, L.: The Anatomy of a large-scale hypertextual web search engine. In: WWW7 Proceedings of the seventh international conference on World Wide Web, vol. 7, pp. 491–495 (1998)
3. Tian, C., Tezuka, T., Oyama, S., Tajima, K., Tanaka, K.: Web Search Improvement Based on Proximity and Density of Multiple Keywords. In: Proceedings of the 22nd International Conference on Data Engineering Workshops, pp. x133 (April 2006)
4. Zhang, Y.: Result Optimization Returned by Multiple Chinese Search Engines Based on XML. In: International Conference of Computational Intelligence and Software Engineering, pp. 1–3 (2009)
5. Kumar, S., Madhan, R.P., Vijayalakshmi, K.: Implementation of two-tier link extractor in optimized search engine filtering system. In: IEEE International Conference on Internet Multimedia Services Architecture and Applications (IMSAA), pp. 1–4 (2009)
6. Lawrence, P., Sergey, B., Motwani, R., Terry, W.: The Page Rank Citation Ranking: Bringing Order to the Web. In: Technical Report, Stanford University InfoLab (1999)
7. Hersovici, M., Jacovi, M., Maarek, Y., Pelleg, D., Shtalheim, M.: Ur Sigalit.: The Shark-Search Algorithm – an application: tailored web site mapping. In: Computer Networks and ISDN systems, Special Issue on 7th WWW Conference, Brisbane, Australia, vol. 30(1-7) (1998)
8. Ozel, S.A., Sarac, E.: Focused crawler for finding professional events based on user interests. In: 23rd International Symposium on Computer and Information Sciences, pp.1–4 (2008)
9. Taeho Jo.: Clustering news groups using inverted index based NTSO. In: First International Conference on Networked Digital Technologies, pp. 1–7 (2009)
10. Ozel, S.A., Sarac, E.: Search engine marketing as key factor for generating quality online visitors. In: Proceedings of the 33rd International Convention, pp.1193 (2010)

MobiLim: An Agent Based License Management for Cloud Computing

Pankaj B. Thorat and Anil K. Sarje

Department of Electronics and Computer Engineering
Indian Institute of Technology, Roorkee Roorkee, India
{Pankspec,sarjepec}@iitr.ernet.in

Abstract. Cloud computing is on-demand computing in which the computing resources are owned and managed by a service provider and the users access the resources via the Internet. But cloud computing potential doesn't begin and end with the personal computer's transformation into a thin client. The mobile platform is going to be heavily impacted by this technology as well. License management is a major issue faced by mobile cloud computing paradigm. This paper presents an agent-based license management approach, the MobiLim, for mobile cloud computing. The mobile devices access the services of cloud and pay for the usage to service provider. Independent software vendors (ISVs) control the access to the resource provider resources. MobiLim provides a secure and robust license management solution to the service provider.

Keywords: mobile agent, cloud computing, cryptography, digital signature.

1 Introduction

Cloud computing is a new paradigm in which computing resources such as processor, memory, software applications and storage are not physically present at the user's location. Service provider owns and manages these resources, and users access them via the Internet [3] [4]. Mobile Cloud Computing refers to an infrastructure where both the data storage and the data processing are done outside the mobile device. Mobile cloud applications move the computing power and data storage away from mobile phones, and put them into the cloud, which brings applications and mobile computing to a much broader range of mobile users. Unlike applications that are downloaded and installed onto the end user's devices, these applications run inside the cloud and can be accessed by any mobile device running a browser. In mobile cloud computing, users access the resources through their mobile devices only when needed and are charged for that by the ISV.

The mobile user shall benefit from mobile cloud computing services. Mobile computing is a computing paradigm which enables code mobility in the network. Simple form of code mobility can be transferring codes to a remote machine for evaluation and collecting the results. In [2], a mobile agent is defined as a software entity autonomously migrating in network during its execution. In mobile cloud computing, the security problems of mobile computation can be effectively reduced

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 335–346, 2011.

as both mobile codes and hosts in cloud can be restricted by trusted companies through certain authentication mechanisms. Therefore, deploying mobile computation, especially mobile agent technology, in cloud computing environment is a promising and feasible approach to enhancing the overall performance of cloud. End users can experience a huge number of new features by enhancing their phones through mobile cloud computing. Users can share resources and applications without a high level of expenditure on hardware and software resources. A recent report by M. Armbrust et al. [4] states that license management is one of ten issues which needs to be tackled for the success of cloud computing. License management is the most important factor for the successful adoption of mobile cloud computing for service provider. The licensing scheme should be beneficial and compatible for both user as well as service provider.

In this paper, we have proposed and implemented MobiLim, an agent based license management framework for service provider, which issues licenses to mobile user and provides a secure mechanism to manage the digital identity and authorization mechanisms required to describe the users, license issuers, and the cloud server by using cryptographic techniques. The licenses are issued depending on the request of the users.

In the next section, we have stated the technological goals of licensing for mobile cloud computing. Then, we have stated the requirement of licensing for mobile cloud computing. Then, we have introduced MobiLim approach and described the design of MobiLim with its implementation in the next section. Finally, we discuss related work and conclude with a discussion of our work.

2 Goals for Building Licensing Mechanism for Mobile Cloud Computing

The various technological goals for building a robust and secure licensing mechanism for mobile cloud computing includes the following:

- Establishing reliable interconnected network of digital authorization and identification among various software vendors, cloud infrastructures, licensing servers, and cloud provisioning servers.
- Software vendor must be able to describe their course of action in a license expression language which differentiates between the end client of the resources, independent of the hosting organization, or the organization that is configuring and installing the software.
- Enabling vendor to choose, preserve their existing licensing approach on the cloud, or adopt a usage- based or subscription-based model by allowing flexible pricing models, and assign the task of billing and pricing to some third party providers that are specialized in that field.
- Enabling the agencies to aggregate, resell and provide integrated software on the cloud, along with custom licenses and prices for reselling agreements.

- Billing should be usage based i.e. pay per use.
- The architecture must be flexible enough to incorporate with existing software modules as well as improved framework.

In the earlier licensing techniques, the contracts were made between the service provider and its customers in which customer buys the software and installs it on local resources in order to use it. But in mobile cloud computing environment, this type of licensing is not possible and valid. In mobile cloud computing, mobile device users use the services provided by service providers as needed. In order to preserve this business as a practice and requirement, licenses should be mobile i.e. a mobile license must be location independent and the user who has acquired the license must be able to use it from any device while ensuring that all legal restrictions are fulfilled.

3 Requirement of Mobile Cloud Licensing

The following are the currently most commonly used licensing techniques which are:

- In system fingerprinting solution [5], a fingerprint of the user's system, i.e., the system on which user is going to access the resources composed of CPU identification numbers and other hardware specific identifiers, is sent to the ISV and the ISV creates a license key and sends it to the user which unlocks the software or utility for the requested hardware.
- A new approach is networked license servers, which are currently used in cluster installations. Licenses are issued on demand to the users. Random policies can be implemented within the server such as the total number of available licenses must not exceed the certain threshold at all times.
- In dongle solution, during the execution of the program hardware, a token is presented. Depending on the communication with this hardware token, the access to particular feature is granted and checks which feature is licensed.

The entire above mentioned license management schemes are not suitable for Cloud environment because the above license management mechanisms licenses are restricted to small IP ranges or single machines. These mentioned mechanisms do not support usage based licensing on an arbitrary machine. The Hardware dongle solution is not applicable for distributed environments of cloud computing. Sever based licensing mechanism uses IP address for authentication which is not useful for cloud environment because licenses should be mobile.

Our proposed license management scheme satisfies the requirements which are necessary for execution of licensing mechanism in distributed environment like cloud computing. MobiLim supports requirements like license mobility i.e. user can access the services of cloud from anywhere; usage based licensing, access control, Billing and accounting, user interface for easy access and request submission. In this paper we describe complete software license management architecture and its implementation.

We are using mobile agents for communication between platforms. There are various advantages of using mobile agents in a distributed environment where mobile devices communicate via wireless network with other platforms. Some of them are stated below [1]:

- *Asynchronous and autonomous execution:* Most of the tasks require seamless connectivity between the mobile device and network to execute. But mobile devices frequently depend on expensive or fragile network connections so it is not economically or technically feasible. To overcome this difficulty, mobile agents are useful because task can be encapsulated within them and then can be dispatched into network. After being dispatched, the agents can operate asynchronously and autonomously and become independent of the process that created them. The mobile device after disconnection can reconnect at a later time to collect the agent.
- *Network load reduction*: Mobile agents enable users to enclose a conversation and transmit it to a destination host where communications take place locally. To reduce raw data in the network mobile agents are useful. When very large volumes of data are stored at remote hosts, that data should be processed in its locality rather than transferred over the network. The motto for agent-based data processing is simple: Move the computation to the data rather than the data to the computation.
- *They overcome network latency:* For critical real- time systems, latencies are not acceptable. Mobile agents can be dispatched from a central controller to act locally and execute the controller‘s directions directly. Because of this property mobile agents offer a solution to reduce network latency.
- *Adapt dynamically:* Mobile agents has ability to sense their execution environment and react autonomously to changes. Multiple mobile agents have the unique ability of distributing themselves among the hosts in the network to maintain the optimal configuration for solving a particular problem.
- *Heterogeneous platforms:* Network computing is fundamentally heterogeneous, often from both hardware and software perspectives. Because mobile agents are generally computer and transport layer independent (dependent on only their execution environments), they provide optimal conditions for seamless system integration.
- *Robust and fault-tolerant:* Mobile agent's ability to react dynamically to unfavorable situations and events makes it easier to build robust and fault tolerant distributed systems. If a host is being shut down, all agents executing on that machine are warned and given time to dispatch and continue their operation on another host in the network.
- *Distributed information retrieval:* Instead of moving large amounts of data to the search engine so it can create search indexes, agent creators can dispatch their agents to remote information sources where they locally create search indexes that can later be shipped back to the system of origin.
- *Parallel processing:* Mobile agents can create a cascade of clones in the network. Another potential use of mobile agent technology is to administer parallel processing tasks. If a computation requires so much processor power that it must be distributed among multiple processors, an infrastructure of mobile agent hosts can be a plausible way to allocate the related processes.

In our proposed solution, we have considered three participants: users, resource providers and ISVs. Mobile agents are used to communicate for the communication between these three components. The use of mobile agents addresses the various problems which are associated with mobile devices. The above mentioned properties of mobile agents helped in designing the secure, robust and efficient licensing mechanism.

4 The Mobilim Approach

In MobiLim there are three components: users/customer, resource providers, ISV that we have outlined above. The MobiLim client is used by the user for acquiring license before accessing the resources. MobiLim server is managed by ISV whose job is to issue license and third one is resource provider who is the owner of the resources. These three entities are explained below.

1. *Resource provider:* Resource provider submits his resource information like resource name, resource prices, types of service and available number of copies of resource to the ISV. Different service providers own different kind of resources like software, hardware and platform which they want provide as a service to the customer. After issuing the license to customer, the service provider handles the request and allows the customer to access the resources.
2. *MobiLim server:* MobiLim server is managed by the ISV whose job is to issue new licenses to new customers and check the validity of the license which was acquired earlier for pending jobs. MobiLim sever maintains a database of various resources which belongs to different resource providers along with their attributes like its price, available number of copies and type of service. The MobiLim server is an entity that can legally issue software licenses, such as a vendor or reseller. Since MobiLim server has near-complete knowledge of the design of a provisioned software application, the MobiLim server is able to enforce a license's thresholds on unique usage metrics instead of the typical per-CPU or per-seat constraints.
3. *User/ customer:* User is at the client side who accesses the services of cloud when needed and pay for that according to pricing policies of the MobiLim server. User, who is a customer, has access to the client side of the MobiLim graphical user interface. User can choose the service by selecting it from the user interface. Depending on the selected service, the request is redirected to the MobiLim server which handles it.

The central idea of MobiLim is to accept the request of users for the service and according to the requirements of the user specified in the license request token and the availability of the resources at that time, MobiLim server attach the license to the customer license request agent. A mobile agent is generated by the client program at the users site which carries license request token for a given set of input data which consist of user requirements like name of the name of service, time for which he wants to use the service or the size of job in number of instructions which user wants to execute by using the resources. The license request token is a file that can be transferred together with the input data to the compute site with the help of mobile agent. We call this agent as license agent who has all information that the license verifier needs in order to check the identity of user and validity of the software license. The lifecycle of license request agent is explained in details below.

4.1 License Request Agent Lifecycle

In order to clarify the concept of our license agent we will shortly outline the lifecycle (Fig.1) of it. An agent consists of a set of hashes of the input files and a license terms specification which is software-specific. When a user submits his requirements, a request agent is generated to encapsulate the submitted requirement by the MobiLim client and it migrates to the MobiLim server. The requirements inside the agent are then extracted at the MobiLim server site. After performing various checks depending upon the requirement, MobiLim server licenses the job which enables it to transfer at the service provider site. During job startup the agent is evaluated again.

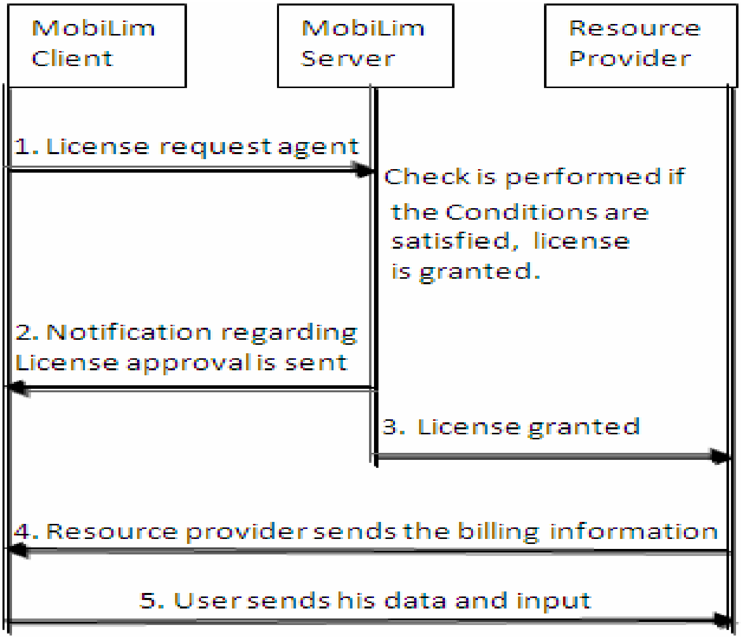

MobiLim Client
MobiLim Server
Resource Provider
1. License request agent
Check is performed if the Conditions are satisfied, license is granted.
2. Notification regarding License approval is sent
3. License granted
4. Resource provider sends the billing information
5. User sends his data and input

Fig. 1. License Agent Migration in MobiLim

When a user wants to use the services of the cloud, he needs to acquire a valid license for the requested services. If the user is new then he has to first create account before requesting services of cloud else he can login into his existing account. After log in, user specifies his requirement called as license terms which are MobiLim server specific. It includes name of resource, time for which user wants to access it or the number of instructions users want to execute on the resource providers hardware. Basically depending on the type of service, specifications are set. In cloud computing, services are categorized into mainly three types,

1. SaaS - Software-as-a-Service [10] [11] is a model of software deployment whereby a provider licenses an application to customers for use as a service on demand. Software or an application is hosted as a service and provided to customers across the Internet. This mode eliminates the need to install and run the application on the customer's local computers. In MobiLim for Software-as-a-Service, the subscription is based time for which user wants to use the software and depending on the specified usage time the user is charged.
2. IaaS - Infrastructure-as-a-Service [11] is the delivery of computer infrastructure as a service. Rather than purchasing servers, software, data center space or network equipment, clients instead buy those resources as a fully outsourced service. In MobiLim for Infrastructure-as-a-Service, the subscription is based on the number of instructions user has executed and depending on that the user is charged.
3. PaaS - Platform-as a-Service [11] [12] is the delivery of a computing platform and solution stack as a service. It facilitates the deployment of applications without the cost and complexity of buying and managing the underlying hardware and software layers. The consumer uses a hosting environment for their applications. In MobiLim for platform-as-a-Service, the subscription is based time for which user wants to use the software and depending on the specified usage time the user is charged.

After submitting the requirements, the MobiLim client encrypts the requirement which includes the current IP address of the machine on which the user working, the customer accounts ID and license term specifications. License terms specifications are used by the ISV to decide the type of license to be issued to the user for this particular job. The request token contains license term specifications which are signed with the user's X.509 certificate. MobiLim client generates one agent which encapsulates the request token and migrates to MobiLim server. The server agent who is a stationary agent extracts the license terms specifications and the user's identity from the license agent. The MobiLim server which is implemented by the ISV uses this information to enforce the ISVs business policies. For that MobiLim server maintains database that holds 3 tables:

- Resource information table: This table stores the information about the resources. Resource information table has various fields which includes resource name, resource provider name, resource provider IP address, resource price, and number of instances or copies available of a particular resource.
- Customer information table: This table stores the information about the requests arrived for license. Fields of this table are resource name, resource provider name, current IP address of the machine on which the user working, the customer accounts ID and usage time.
- Granted license table: This table stores the information about the license requests which are granted by the MobiLim server. Fields of this table are resource price, resource name, resource provider name, current IP address of the machine on which the user working, the customer accounts ID and usage time.

When a request for the license arrives at MobiLim server, it checks the identity of user and compares it with the customer database. If the MobiLim server is able to fulfill all the requirements of user then only it grants license to the customer else a notification is sent to the customer regarding unavailability of the resource. Depending upon the requirements of the user, he is charged and the billing record is stored in the database. After issuing the license to the user, MobiLim server uses its own certificate to sign the request which is then redirected to service provider site. This signed request agent is act as a license which enables resource provider to identify the customer on receiving input data from customer. After arrival of request agent, service provider allocates the resources according to specifications of request and notifies customer to send his input and data with the help of license agent. The MobiLim servers signed request is considered as a valid license.

User transfers the license together with the input data through mobile agent to the service provider's site which is in encrypted form. On receiving the job, the license is inspected by extracting the data inside license agent. The application decrypts the data and compares the hash of license with the locally calculated hash of the user's license. If locally computed hashes are identical to the hashes stored in the license then service provider grants the resources requested by the user and computation starts.

4.2 Cryptographic Algorithms

We have used hashing and signature algorithms to make MobiLim more secure and robust. In this section, we describe the cryptographic techniques in details that we have used in the MobiLim. MobiLim server licenses the customer and sends a secret key Share SK1 to the service provider. After getting notification about the license approval from the MobiLim server customer sends his license along with the data encapsulated within the license request agent in encrypted form. Let N be the license file, with a collision resistant one-way hash function hF, the hashes are generated. For license n Є N we compute Hn = hF(n). A request token which is to be encapsulated inside license agent has two tuples, first one is hashes of license and the second is input data. We get a unique identification of the contents of license request agent by construction of the hash functions. At the service provider side, firstly the content of license request agent is extracted and decrypted then the hashes of license are compared with locally computed hashes of the previously stored same license. If they are equal, a computation starts.

When the license agent arrives at service provider, service provider extracts the token from it. Applying public-key cryptography such as the RSA cryptosystem [6] in combination with secret key sharing [7] can help us to implement dual control over the encrypted data. Each service provider is assigned a unique public and secret key pair (PK/SK). The ISV can autonomously encrypt the token by using the public encryption key PK. The ISV and the service provider share the corresponding decryption secret key SK, using secret-sharing techniques [8]. A basic secret-sharing scheme is used. Initially, the customer uses the PK for the data encryption and MobiLim server sends SK1 which is share of SK along with encrypted token by encapsulating it within license agent. After receiving agent, the service provider uses his SK2 key and computes the two key shares SK1 and SK2 of the same bit length as SK, such that SK = SK1 $\oplus$ SK2, where $\oplus$ denotes the exclusive or XOR) bit operator. Then using SK, resource provider decrypts the token. In this process, each party will possess only one of the two key shares. In this way, decrypting a target data item requires the cooperation of two entities, as Figure 2 shows.

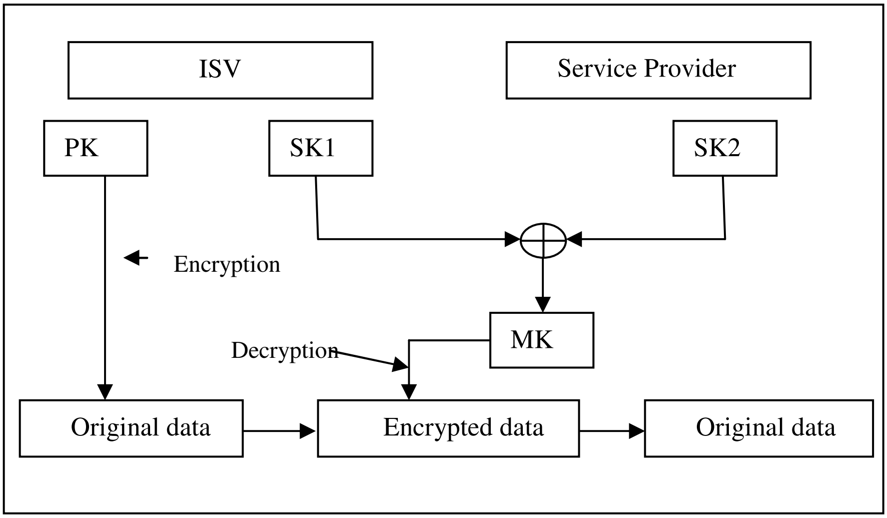


Fig. 2. Sharing the decryption key of the encrypted retained data between the service provider and the ISV

When the license agent arrives at ISVs MobiLim server, server extracts the token from it. Applying public-key cryptography such as the RSA cryptosystem [6] in combination with secret key sharing [7] can help us to implement dual control over the encrypted data. Each service provider is assigned a unique public and secret skey pair (PK/SK). The ISV can autonomously encrypt the token by using the public encryption key PK. The ISV and the service provider share the corresponding decryption secret key SK, using secret-sharing techniques [8].

Standard cryptographic hashing technique SHA-256[5] helps maintain the retained data's integrity. Each data item is hashed and the hash value is then encrypted along with the corresponding data item, by using the encryption mechanisms described previously. Thus, each encrypted entry also contains the corresponding data item's integrity check. This lets providers detect alterations of the stored data because an attacker won't be able to successfully modify or replace the hash value without knowledge of the encryption key. The signatures are generated using X.509 certificates with the RSA algorithm [6] in the Digital Signature Standard (DSS) [7] form. The design of MobiLim is modular because of which replacing the existing modules and algorithm is easier.

5 Implementation

The proposed system model of MobiLim has been designed for implementation on Windows as well as Linux based mobile devices (laptops). For implementation we have used Java as a programming language. In order to study the efficiency of the method presented in this paper, we developed a cloud simulator using Aglets Software Development Kit [9]. Aglet is a middleware aimed at developing multiagent systems and applications conforming to FIPA standards for intelligent agents. Aglet is a java-based framework for mobile agents. With aglets, one can construct objects that can move from one host to another. When the aglet moves, it takes along its program code as well as the states of the objects it is carrying. Aglets make use of a communication system based on message passing.

MobiLim has graphical user interface at the client side which is executed at client/user side when user want to access the service. After executing the program, the window appears where user enters his user name and password to login into his account. Before requesting for the license the user is authenticated. User can access his existing services from anywhere, this property made the licenses mobile in nature.

After log in account another service option window appears, which has a menu bar on that the services are classified according to their type. User can choose the type of service he wants to access. According to the type of the service selected, the list of available utilities appears then user can click on the utility. After selecting utility the "service selected" frame pops out. User can select the type of resource/service by just clicking on that particular resource name and which invokes an internal frame. For software as a service, "service selected" internal frame has two text fields one of them is name of resource/utility that is requested which automatically appears there after selecting on the resource/utility and the other is time for which user wants use the resource/utility.

After clicking on OK button, MobiLim client performs its assigned operation and send this information using mobile agents to independent software vendor. Independent software vendor searches the appropriate service provider in his database who has the resources demanded by the client and forward the request to that particular service provider. Depending on the billing policies of the service provider customer is charged and informed whether the request is successfully granted or not.

6 Related Works

There are various commercial products available in the market. But they do not reveal their implementation details. So it is not easy to compare these products with our approach. IBM's LUM [13], Acresso's FLEXnet [14], HP's iFOR/LS [15] are some the famous licensing software's in the market. GenLM [5] is a license management technique developed for grid and cloud environment. GenLM, also allows ISVs to manage their license usage in a distributed world. The main idea of GenLm is to attach the license not to a person or a node or but to issue licenses for the input or data. The disadvantage of the GenLM solution is that it licenses the whole data or input that user wants executes which requires transfer of data to independent software vendor as well as to the resource provider site. Li et al. [16] argued in their review work about license management requirements for grid and cloud environment that licenses should be managed in terms of a service level agreement. But their approach is not beneficial for the independent software vendor to follow. MobiLim is much easier to follow and implement for the independent software vendors. Because of the modular design of MobiLim, it is easier to upgrade and change the existing modules as per the needs of the application and distributed environment.

7 Conclusion and Future Work

In this paper we have proposed and implemented MobiLim, an agent-based license management framework for mobile cloud computing environments. MobiLim is developed for mobile devices by considering their limitations and ensures security for both the customer and resource provider. MobiLim provides an agent- based licensing solution to manage the increased complexity of software licensing in the mobile cloud computing. We showed how the submission of the job via the interface portal takes place in MobiLim and how MobiLim can monitor the availability of licenses. An important feature of our license management architecture is the accounting and billing part, which allows a flexible license cost model. In the future work, we are trying to implement licenses scheduling depending on the priority of the jobs. Priority will depend on the quality of service demanded by the customer.

References

1. Lange, D., Oshima, M.: Seven Good Reasons for Mobile Agents. Communications of ACM 42(3), 82–89 (1999)
2. Kotz, D., Gray, et al.: Mobile Agents and the Future of the Internet. ACM SIGOPS Operating Systems Review 33(3), 7–13 (1999)
3. Rajkumar, B., Chee, Y.: Srikumar v.: Market- Oriented Cloud Computing: Vision, Hype, and Reality for Delivering IT Services as Computing Utilities. In: Proceedings of the s10th IEEE International Conference on High Performance Computing and Communications, Dalian, China (2008)
4. Armbrust, M., Fox, A., Griffith, R.: Above the Clouds: A Berkeley View of Cloud Computing. EECS Department, University of California Berkeley,Tech.Rep. UCB/EECS 2009-28 (2009)
5. Dalheimer, M., Pfreundt, M.: GenLM: License Management for Grid and Cloud Computing Environments. In: 9th IEEE/ACM International Symposium on Cluster Computing and the Grid, pp.132–139 (2009)
6. Rivest, R., Shamir, A., Adleman, L.: A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. Communication of ACM 21(2), 120–126 (1978)
7. Shamir, A.: How to Share a Secret. Comm. of ACM 22(11), 612–613 (1979)
8. Panayiotis, K.: Data Retention and Privacy in Electronic Communications. IEEE Security & Privacy, 46–52 (September/October 2008)
9. Aglets mobile agent platform, `http://aglets.sourceforge.net/`
10. Foster, I., Zhao, Y., Raicu, I., Lu, S.: Cloud Computing and Grid Computing 360-Degree Compared. In: Grid Computing Environments Workshop, GCE 2008, pp. 1–10 (November 2008)
11. Maggiani, R.: Cloud computing is changing how we communicate. In: International Professional Communication Conference, IPCC 2009, pp. 1–4. IEEE, Los Alamitos (2009)
12. Lizhe, W., Jie, A., Kunze, M., Castellanos, A.C., Kramer, D., Karl, et al.: Scientific Cloud Computing: Early Definition and Experience. In: 10th IEEE International Conference on High Performance Computing and Communications, HPCC 2008, pp. 825–830 (2008)

13. IBM License Use Management, http://www-01.ibm.com/software/tivoli/products/license-use-mgmt/
14. Acresso FLEXnet Overview, http://www.acresso.com/products/software-hardware.htm
15. iFOR/LS Quick Start Guide, http://docs.hp.com/en/B2355-90108/index.html
16. Li, J., Waeldrich, O., Ziegler, W.: Towards SLA-Based Software Licenses and License Management in Grid Computing. In: Wu, S., Yang, L.T., Xu, T.L. (eds.) GPC 2008. LNCS, vol. 5036, pp. 139–152. Springer, Heidelberg (2008)

Data Mining on Grids

Shampa Chakraverty, Ankuj Gupta, Akhil Goyal, and Ashish Singal

Department of Computer Engineering, Netaji Subhas Institute of Technology,
Azad Hind Fauz Marg, Sector3, Dwarka, New Delhi 110078
apmahs@rediffmail.com, ankuj2004@gmail.com, akhil587@gmail.com,
ashish.singal.coe@gmail.com

Abstract. Data mining algorithms are widely used today for the analysis of large corporate and scientific datasets stored in databases and data archives. Industry, science and commerce fields often need to analyze very large datasets maintained over geographically distributed sites by using the computational power of distributed and parallel systems. Grid computing emerged as an important new field of distributed computing, which could support the distributed knowledge discovery applications. In this paper, we have proposed a method to perform Data Mining on Grids. The Grid has been setup using Foster and Kesselman's Globus Toolkit, which is the most widely used middleware in scientific and data intensive grid applications. For the development of data mining applications on grids we have used Weka4WS. Weka4WS is an open source framework extended from the Weka toolkit for distributed data mining on Grid, which deploys many of machine learning algorithms provided by Weka Toolkit. To evaluate the efficiency of the proposed system, a performance analysis of Weka4WS by executing distributed data mining tasks, namely clustering and classification, in grid scenario has been performed. At last, a study on the speed up obtained by doing data mining on grids is done.

Keywords: Weka4WS, Distributed Data Mining, Grid Computing, Classification, Clustering Grid.

1 Introduction

Data Mining (DM) or Knowledge Discovery in Database (KDD) is the process of extracting useful and hidden pattern from the massive datasets, which is the combination of database management, information system, statistical, mathematics, visualization, etc. [1][2]. In order to achieve the extraction of models, the phase of DM includes data preprocessing, data mining, and patterns' evaluation.

With the rapid development of information technology, massive data collections in terabyte and petabyte scale located on geographically distributed sites need to be maintained and analyzed in many areas such as scientific, commercial, financial, and so on. The conventional technology used for data mining is incapable of dealing with this problem. So, Distributed Data Mining (DDM) [3][4][5][6][7], which is the process of analyzing geographically dispersed large

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 347–358, 2011.

datasets for extracting novel and interesting pattern and model, becomes increasingly essential.

Grid computing emerged as an important new field of distributed computing, distinguished from conventional distributed technologies by its focus on large scale resource sharing, innovative applications, and in some cases, high performance orientation. At the same time, with the broad application of the web services in the industries, WSRF (Web Service Resource Framework) [8], has born as the production of Grid computing and web services combination, which is a family of technical specification concerned with the resource properties, addressing, service group, and lifetime management of stateful resources. WSRF describes the WS–Resource definition and association with the description of a Web Service interface, and describes how to make the properties of a WS–Resource accessible through a Web Service interface.

Weka4WS [9] is a framework that was developed by the University of Calabria, which is a new approach by WSRF compliant toolkit towards distributed data mining on grid environments and supports remote grid execution of the data mining algorithms through grid services. It is a parallel and distributed software architecture that integrates data mining techniques and WSRF enabled grid technologies. In the Weka4WS architecture data mining tools are integrated with the WSRF compliant web service. Thus the Weka4WS can be exploited to perform data mining on very large datasets available over grids, make scientific discoveries, improve industrial processes and organization models, and uncover business valuable information [10].

In the following sections we describe the design and implementation of Weka4WS. To evaluate the overhead introduced by the service invocation mechanisms and its effects on the efficiency of the proposed system, we also present a performance analysis of Weka4WS executing distributed data mining tasks in different network scenarios.

The outline of the paper is as follows. Section 2 describes the architecture of the Weka4WS. Section 3 introduces an execution of a distributed data mining task. Section 4 studies the performance of data mining on grids and Section 5 concludes the paper.

2 The Architecture of Weka4WS

Weka4WS is thought as an extension of Weka, which is a WSRF compliant toolkit for distributed data mining on grid. It is based on the Client/Server architecture. Figure 1 shows the hierarchy architecture of Weka4WS. Just as we could see that the four layers of Weka4WS framework include: Globus Toolkit (GT) [11], Weka Library, Weka4WS Client (Graphic User Interface) and Weka4WS Server (Data Mining WS–Resource).

- Globus Toolkit is a software toolkit, developed by the Globus Alliance and implemented the emerging of WSRF specification. The toolkit includes quite a few high–level services that we could use to build Grid applications. At

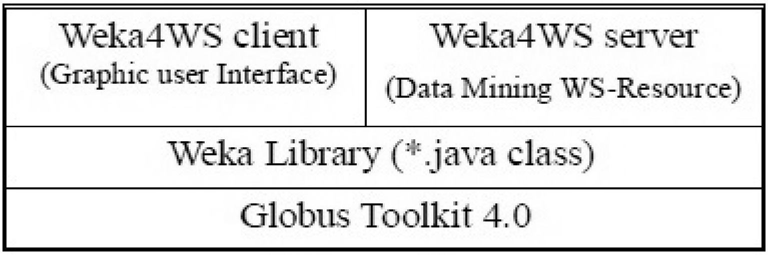


Fig. 1. Architecture of Weka4WS

the same time, it also includes a resource monitoring and discovery service (MDS) [12], a job submission infrastructure (GRAM) [13], a security infrastructure (GSI) [14], and data management services (GridFTP) [15], which could achieve the management of all kinds of resources.

- Weka Library is a large collection of machine learning algorithms, for data pre–processing, classification, clustering, association rules and visualization, which provides the basic functions of distributed data mining.
- Weka4WS server provides the Weka4WS Web services allowing for the execution of remote data mining tasks. It is responsible for the creation, destruction and lifetime management of the instances of grid services by using a Factory/Instance pattern. These grid services implement the functionality of the different algorithms and stages of the data mining process. What‘s more, the server side adopts the concept of WS–Resource, which is the combination of stateless web services and stateful resources, to manage all kinds of resources (computer, algorithms, etc.) efficiently.
- The Weka4WS client are the local machines providing the Weka4WS client software. They are responsible for interacting with the servers WS–Resources, which provides the graphic user interface of data mining task for users to implement the local or remote data mining jobs. At the same time, the client side takes charge of assigning the jobs and coordinating the workload balance of different sites.

Figure 2 shows the software components of client and server in the Weka4WS framework.

Server nodes include two components: a Web Service (WS) and the Weka Library (WL). The WS exposes all the data mining algorithms provided by the underlying WL. Therefore, requests to the WS are executed by invoking the corresponding WL algorithms.

Client nodes include three components: Graphical User Interface (GUI), Client Module (CM), and Weka Library (WL). The GUI is an extended version of the Weka Explorer environment to support the execution of both local and remote data mining tasks. Local tasks are executed by directly invoking the local WL, whereas remote tasks are executed through the CM, which operates as an intermediary between the GUI and Web services on remote server nodes.

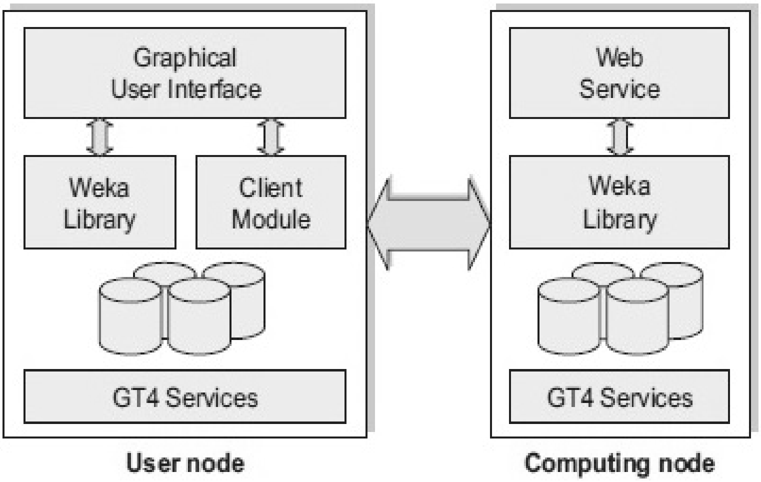


Fig. 2. Software Components of Client and Server in Weka4WS

3 Task Execution on Weka4WS

This section describes the steps performed to execute a data mining task on a remote Web service in the Weka4WS framework. Figure 3 shows a Client Module (CM) that interacts with a remote Web Service (WS) to execute a data mining task. In particular, this example assumes that the CM is requesting the execution of a classification task on a dataset located on the user node.

The following steps are executed in order to perform this task, (see Figure 3):

1. **Resource creation**. The CM invokes the createResource operation to create a new resource that will maintain the state of the subsequent classification analysis. The state is stored as the property of the resource. The WS returns the Endpoint Reference (EPR) of the created resource. The EPR is globally unique and distinguishes this resource from all other resources over the Grid. Subsequent requests from the CM will be directed to the resource identified by that EPR.
2. **Notification subscription**. The CM invokes the subscribe operation that subscribes to notifications about changes that will occur to the model resource property. As soon as this property will change its value (i.e., as soon as the model has been computed), the CM will receive a notification containing that value, which represents the result of the classification task.
3. **Task submission**. The CM invokes the classification operation requiring the execution of the classification task. This operation receives a set of parameters, which are the name of the classification algorithm to be used, the URL of the dataset to be mined and its checksum.
4. **File transfer**. If dataset is not already available on the computing node, the CM requests to transfer it to the URL specified as a return value by

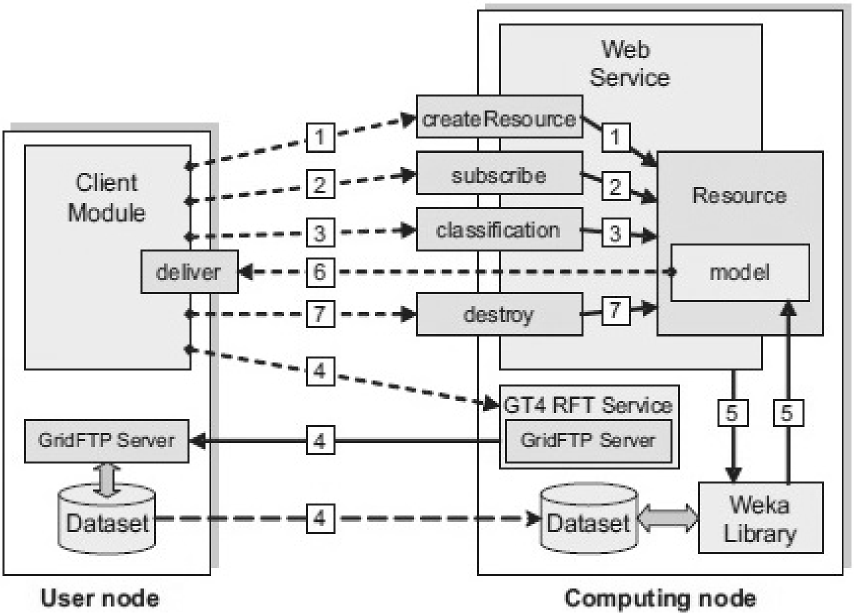


Fig. 3. Client Module Interactions

the classification operation. The transfer request is managed by the GT's Reliable File Transfer (RFT) service running on the computing node, which in turn invokes the GridFTP servers [16] running on the user and computing nodes. If the size of the dataset is over a given threshold, it is compressed before the transfer and decompressed at destination.

5. **Data mining**. The classification analysis is started by the WS through the invocation of the appropriate Java class in the Weka library. The result of the computation (i.e., the inferred model) is stored in the model property of the resource created on Step 1.
6. **Results notification**. As soon as the model property has been changed, its new value is notified to the CM by invoking its implicit deliver operation. This mechanism allows for the asynchronous delivery of the execution results as soon as they are generated.
7. **Resource destruction**. The CM invokes the destroy operation, which eliminates the resource created on Step 1.

4 Performance Analysis

To evaluate the performance of the system, we carried out some experiments where we used Weka4WS for executing data mining tasks in Local Area Grid (LAG), where the computing node and the user node are connected by a wireless local area network. In the following, we discuss performance results obtained by executing clustering and classification data mining tasks on publicly available datasets. The main goal of our analysis is to evaluate the overhead introduced by the WSRF mechanisms and the distributed scenario with respect to the overall execution time.

4.1 Clustering

For our clustering experiments, we used the USCensus1990 dataset from the UCI KDD Archive [17]. We extracted from it ten datasets containing a number of instances ranging from 1450 to 14500, with a size ranging from 200 KB to 2 MB. We used Weka4WS to execute the Expectation Maximization (EM) clustering algorithm on each of these datasets and asked the system to group data in 5 clusters on the basis of 10 selected attributes. For each dataset size we had run 20 independent executions. The measures are reported in the following graphs, resulting from the average values of all the executions.

Figure 4 shows the execution times of the different steps for a dataset size ranging from 200 KB to 2 MB. As shown in the figure, the execution times of the WSRF specific steps are independent of the dataset size, namely: resource creation (0.01 s, on average), notification subscription (0.389 s, on average), task submission (0.236 s, on average) and resource destruction (0.01 s, on average).

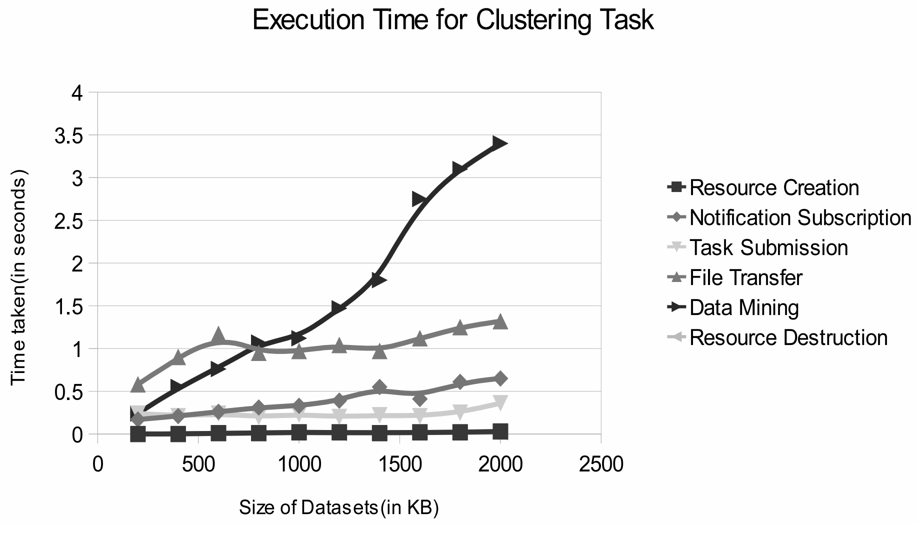


Fig. 4. Execution Time for Clustering Task

On the contrary, the execution times of the file transfer and data mining steps are proportional to the dataset size. In particular, the execution time of the file transfer ranges from 0.58 s for 200 KB to 1.32 s for 2 MB, while the data mining execution time ranges from 9.6 s for the dataset of 200 KB, to 136 s for the dataset of 2 MB. The total execution time (not shown in the Figure 4) ranges from 10.59 s for 200 KB, to 138.36 s for 2MB.

4.2 Classification

For data classification experiments, we used the kddcup99 dataset available at the UCI archive [18]. As before, we extracted ten datasets from it, with a number of instances ranging from 1900 to 19000 and a size ranging from 200 KB to 2 MB, and then we used Weka4WS to perform a classification analysis on each of those datasets. In particular, we employed the J48 classification algorithm, using 5–folds cross–validation based on 10 attributes.

Figure 5 shows the execution times of the different steps of the classification task in the LAG scenario for a dataset size ranging from 200 KB to 2 MB. As highlighted in the clustering experiments, the execution times of the WSRF specific steps are independent of the dataset size, whereas the execution times of the file transfer and data mining steps are proportional to the dataset size.

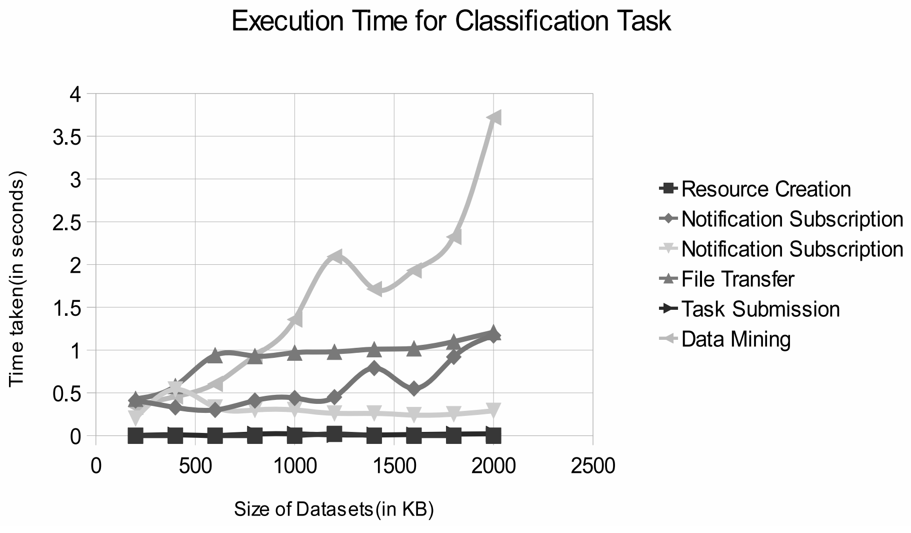


Fig. 5. Execution Time for Classification Task

In particular, the execution time of the file transfer ranges from 0.43 s for 200 KB to 1.21 s for 2 MB, while the data mining execution time ranges from 3.37 s for the dataset of 200 KB, to 37.2 s for the dataset of 2 MB. The total execution time (not shown in the Figure 5) ranges from 4.41 s for the dataset of 200 KB, to 39.89 s for the dataset of 2 MB.

To better highlight the overhead introduced by the WSRF mechanisms and the distributed scenario, Figure 6 shows the percentage of data mining, file transfer, and WSRF overhead (i.e., the sum of resource creation, notification subscription, task submission, results notification, and resource destruction steps), with respect to the total execution time.

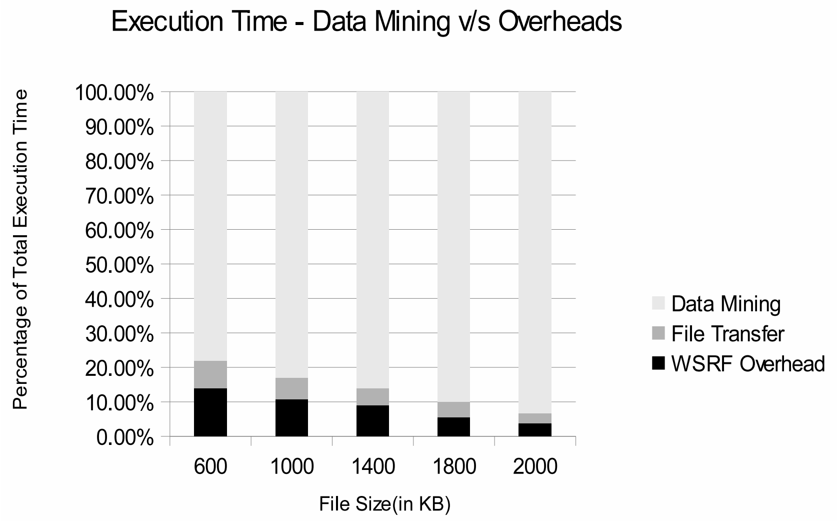


Fig. 6. Execution Time –Data Mining v/s Overheads

The data mining step takes the 76.41% of the total execution time for the dataset of 200 KB, whereas it takes the 93.26% of the total execution time for the dataset of 2 MB. At the same time, the file transfer ranges from 9.75% to 3.03%, and the WSRF overhead range from 13.83% to 3.71%. We can observe that in the scenario neither the file transfer nor the WSRF overhead represent a significant overhead with respect to the total execution time.

As a concluding remark, the performance analysis discussed above demonstrates the efficiency of the WSRF mechanisms as a means to execute data mining tasks on remote machines. By exploiting such mechanisms, Weka4WS can provide an effective way to perform compute intensive distributed data analysis on a large scale Grid environment.

4.3 Speed Up

A class of applications that can efficiently exploit the Weka4WS approach is that of a single dataset analyzed in parallel on multiple Grid nodes using different data mining algorithms. In the following, we describe an example of application in which a real dataset is analyzed with Weka4WS by running multiple instances of the same clustering algorithm, with the goal of obtaining multiple clustering models from the same data source. The covertype dataset [19] from the UCI archive has been used as data source. The dataset has a size of about 72 MB and contains information about forest cover type for 581012 sites in the United States.

Weka4WS has been used to run an application in which 6 independent instances of the KMeans algorithm [20] perform a different clustering task on the covertype dataset. In particular, each KMeans instance has been asked to group data into a given number of clusters, ranging from 2 to 7, based on all the attributes but the last one (the cover type). The same application has been executed using a number of computing nodes ranging from 1 to 5 in order to evaluate the speedup of the system.

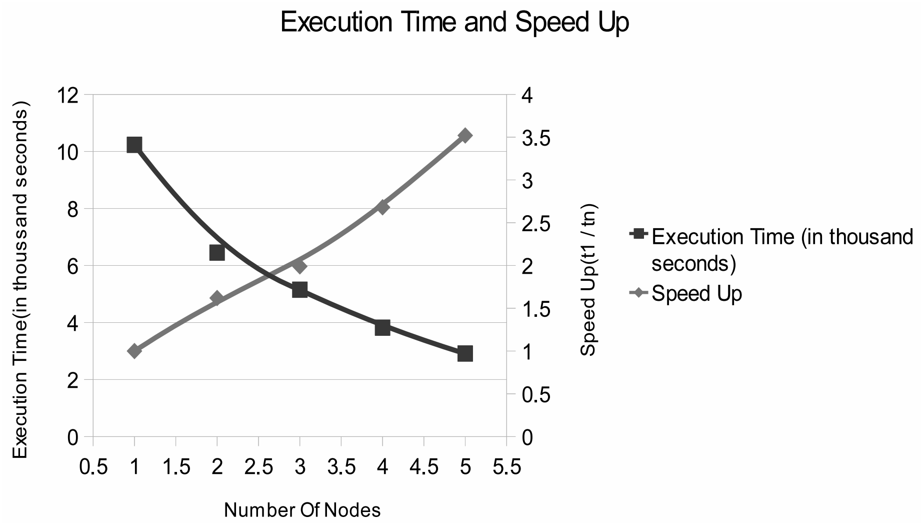
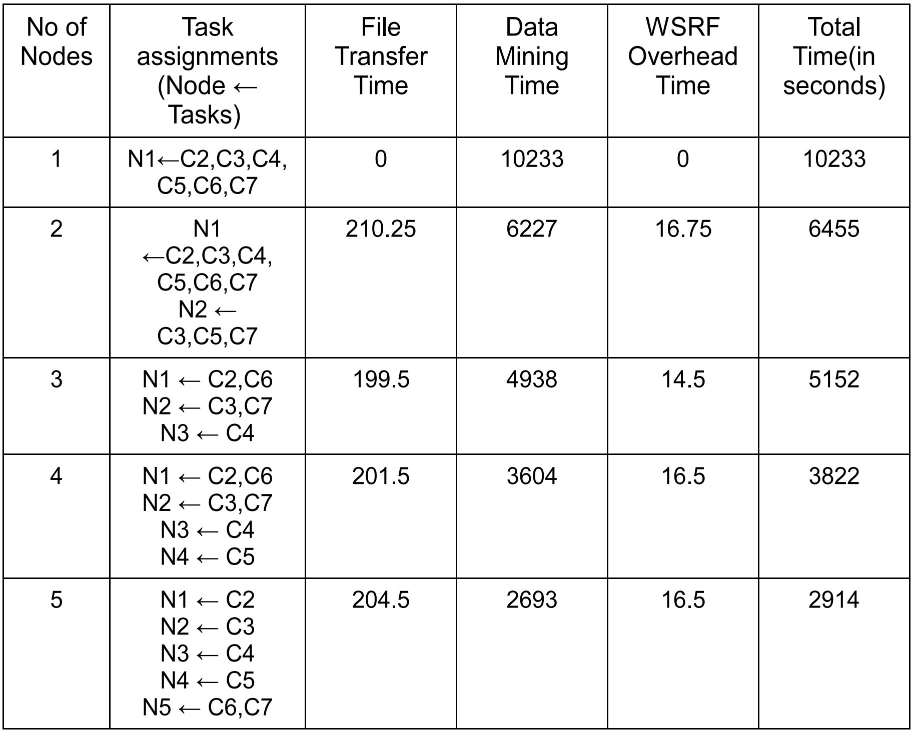


Fig. 7. Execution Time and Speedup values for various number of nodes

The execution time and speedup values for different number of nodes are represented in Figure 7. Figure 8 reports the execution time of the application when 1, 2, 3, 4 and 5 computing nodes are used. The 6 clustering tasks that constitute the overall application are indicated as C2–C7, where the notation Cn refers to the task of grouping data into n clusters. The table shows how the clustering tasks are assigned to the computing nodes (denoted as N1–N5), as well as the partial execution times (file transfer time, data mining time, and WSRF overhead), and the total execution time.

The total execution time decreases from 10233 s obtained using 1 computing node, to 2910 s obtained with 5 nodes. The achieved execution speedup ranges from 1.62 using 2 nodes, to 3.52 using 5 nodes.



No of Nodes	Task assignments (Node ← Tasks)	File Transfer Time	Data Mining Time	WSRF Overhead Time	Total Time(in seconds)
1	N1←C2,C3,C4, C5,C6,C7	0	10233	0	10233
2	N1 ←C2,C3,C4, C5,C6,C7 N2 ← C3,C5,C7	210.25	6227	16.75	6455
3	N1 ← C2,C6 N2 ← C3,C7 N3 ← C4	199.5	4938	14.5	5152
4	N1 ← C2,C6 N2 ← C3,C7 N3 ← C4 N4 ← C5	201.5	3604	16.5	3822
5	N1 ← C2 N2 ← C3 N3 ← C4 N4 ← C5 N5 ← C6,C7	204.5	2693	16.5	2914

Fig. 8. Task assignments and execution times for different number of nodes (time in seconds)

5 Conclusion

We described the design and the implementation of Weka4WS by exploiting the WSRF library provided by Globus Toolkit 4. Firstly, the installation of Globus toolkit is performed successfully with the services–GridFTP, WSGRAM and RFT. Weka4WS is successfully installed to exploit these services of Globus toolkit. To evaluate the efficiency of the implemented system, we also presented a performance analysis of Weka4WS that discusses the execution of two distributed data mining tasks–Clustering and Classification–in a wireless local area network. The experimental results demonstrate the low overhead of the WSRF–Web service invocation mechanisms with respect to the execution time of data mining algorithms and the efficiency of the WSRF framework as a means for executing data mining tasks on remote resources. By exploiting such mechanisms, Weka4WS provides an effective way to perform compute intensive distributed data analysis on large scale Grid environments.

6 Future Work

Weka4WS provides an effective way to perform compute intensive distributed data analysis on large scale Grid environments. Future developments of Weka4WS one can consider are:

1. To make use of the dynamic information on the resources to make an efficient scheduling of the task on the Grid nodes, in order to reduce the execution time of particularly complex applications which operate on big amount of data.
2. A framework in which a distributed data mining application can be composed by several tasks that execute on multiple Grid nodes in parallel and/or in sequence.
3. To support data parallelism, that is the distribution of the data across different parallel computing nodes, besides the currently employed task parallelism which focuses on distributing execution tasks across different computing nodes.

References

1. Ye, N.: The Handbook of Data Mining. Lawrence Erlbaum Associates, Mahwah (2003)
2. Witten, I.H., Frank, E.: Data Mining. Practical Machine Learning Tools and Techniques. Morgan Kaufmann Publisher, San Francisco (2005)
3. Cannataro, M., Talia, D., Trunfio, P.: Distributed data mining on the grid. Future Generation Computer System 18, 1101–1112 (2002)
4. Zeng, L., Xu, L., Shi, Z., Wang, M., Wu, W.: Distribued Computing Environment: Approaches and Applications, pp. 3240–3244. IEEE, Los Alamitos (2004)
5. Chattratichat, J., Darlington, J., Guo, Y., Hedvall, S., Kohler, M., Syed, J.: An Architecture for Distributed Enterprise Data Mining (2002)
6. Du, W., Agrawal, G.: Developing Distribued Data Mining Implementations for a Grid Environment. In: Proceedings of the 2nd IEEE/ACM International Symposium on Cluster Computing and the Grid, pp. 1–2 (2002)
7. Dutta, H.: Empowering Scientific Discovery by Distributed Data Mining on the Grid Infrastructure (2007)
8. OASIS Web Service Resource Framework (WSRF) TC (2004), http://www.oasisopen.org/committees/tc_home.php?wg_abbrev=wsrf
9. The Weka4WS user guide, http://grid.deis.unical.it/weka4ws
10. Cannataro, M., Pugliese, A., Talia, D., Trunfio, P.: Distributed Data Mining on Grids: Services,Tools, and Applications. IEEE Transactions on Systems, Man, and Cybernetics 34(6), 2451–2465 (2004)
11. The Globus Toolkit, http://www.globus.org/toolkit/
12. GT WS MDS WebMDS: System Administrator's Guide, http://www.globus.org/toolkit/docs/4.2/4.2.0/info/webmds/index.html
13. WS GRAM Admin Guide, http://www.globus.org/toolkit/docs/4.2/4.2.0/execution/gram4/index.html
14. Overview of Grid Security Infrastructure, http://www.globus.org/toolkit/docs/4.2/4.2.0/security/security

15. GridFTP Admin Guide, http://www.globus.org/toolkit/docs/4.2/4.2.0/data/gridftp/index.html
16. Allcock, W., Bresnahan, J., Kettimuthu, R., Link, M., Dumitrescu, C., Raicu, I., Foster, I.: The Globus striped GridFTP framework and server. In: Supercomputing Conf. 2005 (2005)
17. Hettich, S., Bay, S.D.: The UCI KDD Archive, University of California, Department of Information and Computer Science. US Census datasets, http://kdd.ics.uci.edu/databases/census1990/USCensus1990.html
18. Hettich, S., Bay, S.D.: The UCI KDD Archive, University of California, Department of Information and Computer Science. In: The Third International Knowledge Discovery and Data Mining Tools Competition datasets, http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html
19. Hettich, S., Bay, S.D.: The UCI KDD Archive, University of California, Department of Information and Computer Science. The forest cover type for 30 x 30 meter cells obtained from US Forest Service (USFS) Region 2 Resource Information System (RIS) data, http://kdd.ics.uci.edu/databases/covertype/covertype.html
20. MacQueen, J.B.: Some Methods for classification and Analysis of Multivariate Observations. In: 5th Berkeley Symposium on Mathematical Statistics and Probability 1967, pp. 281–297. University of California Press, Berkeley (1967)

Text Independent Emotion Recognition Using Spectral Features

Rahul Chauhan, Jainath Yadav, S.G. Koolagudi, and K. Sreenivasa Rao

School of Information Technology
Indian Institute of Technology Kharagpur
Kharagpur - 721302, West Bengal, India
rchouhan2008@gmail.com, jaibhu38@gmail.com, koolagudi@ieee.org,
ksrao@iitkgp.ac.in

Abstract. This paper presents text independent emotion recognition from speech using mel frequency cepstral coefficients (MFCCs) along with their velocity and acceleration coefficients. In this work simulated Hindi emotion speech corpus, IITKGP-SEHSC is used for conducting the emotion recognition studies. The emotions considered are anger, disgust, fear, happy, neutral, sad, sarcastic, and surprise. Gaussian mixture models are used for developing emotion recognition models. Emotion recognition performance for text independent and text dependent cases are compared. Around 72% and 82% of emotion recognition rate is observed for text independent and dependent cases respectively.

Keywords: Gaussian mixture models, emotion recognition, IITKGP-SEHSC, spectral features, text dependent emotion recognition, text independent emotion recognition.

1 Introduction

Emotion recognition from speech has gained increasing attention in the recent years. Human beings extensively use emotions to convey the intended information along with the textual message. Majority of the today's speech systems are developed using neutral speech. Component of emotion is essential to make them more natural for practical applications. Speech emotion recognition has some of the important applications such as: call center conversation analysis that helps to improve quality of service of a call attendant [2], interactive movie [3], story telling and E-tutoring [1] applications would be more practical if they can adapt themselves to listeners or students emotional states. The automatic way to analyze the emotions in speech is useful for indexing and retrieving the audio/video files based on emotions [4]. Medical doctors may use the emotional contents of the patients speech as a diagnosing tool for various ailments. Emotion analysis of telephone conversation between criminals would help crime investigation department. Conversation with robotic pets and humanoid partners would be more realistic and enjoyable, if they are able to express and understand emotions like humans. Emotion recognition based on a speech is one of intensively studied research topics in the domains of human-computer interaction and affective

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 359–370, 2011.

computing. Speech emotion recognition can also be used to augment automated medical or forensic data analysis systems. Emotions are often portrayed differently in different cultures and languages. For example, a specific type of intonation which indicates admiration in Japanese, indicates disbelief in English [6]. A method of translating emotions, in addition to text, between the languages may help improve multi/cross lingual communication. The present study is aimed at recognizing the text-dependent and text-independent emotions from speech utterances using spectral features. Gaussian mixture models are used to develop emotion recognition models. Text-dependent emotion recognition is performed by using the same text prompts for both training and testing of the models. However the utterances are chosen form different recording sessions. In text-independent case, the text prompts used for training and testing the emotion recognition models are different.

From the literature, it is observed that, majority of the studies on emotion recognition have used algorithms based on prosodic features [12]. In the present work, the focus has been given to the spectral features. In particular the effectiveness of mel frequency cepstral coefficients (MFCCs) is explored for speech emotion recognition in the context of text dependent and independent cases. Rest of the paper is organized in 4 sections. Section 2 contains the description of speech database, section 3 describes feature extraction and development of emotion recognition models. In section 4 the result are discussed. Paper is concluded with summary and some important citations.

2 IITKGP:SEHSC (Indian Institute of Technology Kharagpur: Simulated Emotion Hindi Speech Corpus)

The proposed database is recorded using 10 (5 male and 5 female) professional artists from Gyanavani FM radio station, Varanasi, India [13]. The artists have sufficient experience in expressing the desired emotions from the neutral sentences. The male artists are in the age group of 28-48 years with varied experience of 5-20 years. Similarly female artists are from the age group of 20-30 years with 3-10 years of experience. For recording the emotions, 15 Hindi text prompts are used. All the sentences are emotionally neutral in meaning. Each of the artists has to speak the 15 sentences in 8 basic emotions in one session. The number of sessions recorded for preparing the database is 10. The total number of utterances in the database is 12000 ($15\ sentences \times 8\ emotions \times 10\ speakers \times 10\ sessions$). Each emotion has 1500 utterances. The number of words and syllables in the sentences vary from 4-7 and 9-17 respectively. The total duration of the database is around 9 hours. The eight emotions considered for collecting the proposed speech corpus are: anger, disgust, fear, happy, neutral, sad, sarcastic and surprise. The speech samples are recorded using SHURE dynamic cardioids microphone C660N. The distance between the speaker and the microphone is maintained to be around 1 ft. Speech signal was sampled at 16 kHz and each sample is represented as 16 bit number. The sessions are recorded on alternate days to capture the variability in human speech production mechanism. In each session, all the artists have given the recordings of 15 sentences in 8 emotions. The recording is done in such a way that each artist has to speak all the sentences at a stretch in a particular emotion. This provides coherence among the sentences for each emotion category. Since, all the artists are from the same organization, it ensures the coherence in the quality of the collected speech data. The entire speech database is recorded using single microphone and at the same location. The recording was done in a quiet room, without any obstacles in the recording path [8].

2.1 Subjective Evaluation

The quality of the emotions expressed in the database is evaluated using subjective listening tests. Here, the quality represents how well the artists simulated the emotions from the neutral sentences. This evaluation is carried out by 25 post graduate and research students of IIT Kharagpur. This study is useful for comparing the emotion recognition performance in case of human beings and machine. The study is also helpful to determine clearly discriminable and confusing emotions among 8 classes.

In this study, 40 sentences (5 sentences from each emotion) from each artist are considered for evaluation. Before taking the test, the subjects have given the pilot training by playing 24 sentences (3 sentences from each emotion) from each artist's speech data, for understanding (familiarizing) the characteristics of emotions. The forty sentences used in the evaluation are randomly ordered, and played to the listeners. For each sentence, the listener has to mark the emotion category from the set of 8 basic emotions. The overall emotion classification performance for the chosen male and female artists' speech data is given in Table 1. The observation shows that the average recognition rate in case of both male and female speech utterances is about 71% and 74% respectively. Anger, neutral and sad emotions are recognized well compared to other emotions. Disgust and surprise are comparatively less accurately recognized. The expected overlap in the classification is observed between happy, fear and surprise. The emotion recognition performance of subjective listening tests and prosodic features is almost same, which indicates that, human beings mostly use prosodic cues to identify the emotions.

3 Development of Emotion Recognition Models

Emotion recognition using pattern classifiers is basically a two stage process as shown in Fig 1. In the first stage emotion recognition models are developed by training the models using the feature vectors extracted from speech utterances of known emotions. It is known as supervised learning. In second stage testing (evaluation) of the trained models is performed by using the speech utterances of unknown emotions.

Table 1. Emotion classification performance of male and female speech, based on subjective Evaluation; Abbreviations: A-Anger, D-Disgust, F-Fear, H-Happy, N-Neutral, Sad-Sadness, Sar-Sarcastic, Sur-Surprise

	Male Artist(Average: 70.75)								Female Artist(Average: 74.25)							
	A	D	F	H	N	Sad	Sar	Sur	A	D	F	H	N	Sad	Sar	Sur
Anger	91	0	4	3	2	0	0	0	88	0	8	4	0	0	0	0
Disgust	0	51	0	2	10	19	10	8	0	48	0	0	9	30	7	6
Fear	0	0	65	7	0	0	0	28	0	0	71	4	0	0	0	25
Happy	0	0	5	69	0	0	0	26	0	0	8	82	0	0	0	10
Neutral	0	10	0	0	85	5	0	0	0	15	0	0	83	2	0	0
Sad	0	10	0	0	12	72	6	0	0	4	0	0	10	86	0	0
Sarcastic	0	11	0	0	18	0	71	0	0	7	0	0	16	2	75	0
Surprise	0	0	21	16	1	0	0	62	0	0	24	15	0	0	0	61

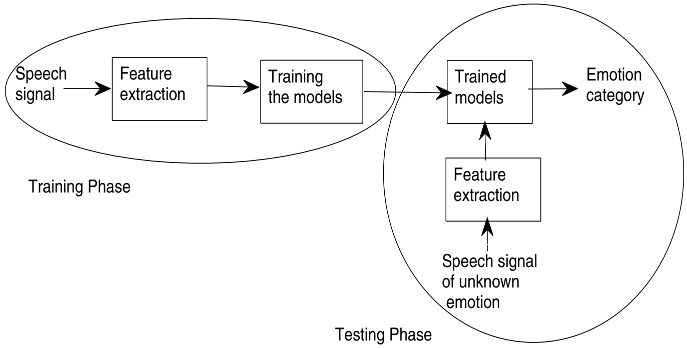


Fig. 1. Block diagram of emotion recognition system

3.1 Feature Extraction

Mel frequency cepstral coefficients (MFCCs) and its derivatives have been used as features for recognizing the speech emotions. Velocity and acceleration coefficients of MFCCs represent finer variations in the spectral properties of the phoneme in the utterances, during expression of emotions. Thirteen mel frequency cepstral coefficients along with 13 velocity (δ) and 13 acceleration ($\delta - \delta$) coefficients are extracted from speech signal to represents gradual spectral variations. The speech frames of size of 20 ms, each time overlapped by 10 ms are used to extract these 39 features. Their concatenation formed the feature vectors. Hamming window is used while framing the speech signal. MFCCs features are extracted from these frames using the MFCC algorithm given in [7]. Cepstral mean subtraction and variance normalization are carried out in order to compensate recording variations. The normalization is done at the sentence level.

3.2 Gaussian Mixture Models (GMM)

In this work, GMMs are used to develop emotion recognition systems using spectral features. GMMs are known to capture distribution of data points from the input feature space. Therefore, GMMs are suitable for developing emotion recognition models using spectral features, as the decision regarding the emotion category of the feature vector is taken based on its probability of coming from the feature vectors of the specific model.

Gaussian Mixture Models (GMMs) are among the most statistically matured methods for clustering and for density estimation. They model the probability density function of observed data points using a multivariate Gaussian mixture density. Given a set of inputs, GMM refines the weights of each distribution through expectation-maximization algorithm. Once a model is generated, conditional probabilities can be computed for test patterns (unknown data points). Number of Gausses in the mixture model is known as number of components. They indicate the number of clusters in which data points are to be classified. In this work, one GMM is developed to capture the information about one emotion. The components within each GMM capture finer level details among the feature vectors of each emotion. Depending on the number of data points, number of components may be varied in each GMM. Presence of few components in GMM and trained using large number of data points may lead to more generalized clusters, failing to capture specific details related to each class. On the other hand over fitting of the data points may happen, if too many components represent few data points. Obviously the complexity of the models increases, if they contain higher number of components. In this work, GMM's are designed with 64 components and iterated for 30 times to attain convergence of weights. The emotion recognition systems developed using spectral features are shown in figure 2.

The emotion recognition models are developed on five speakers' (3 male and 2 female) speech data taken from IITKGP-SEHSC. For text-independent case, out of 15 text prompts, 10 are used for training the models and 5 are used for validation. From the chosen speakers, in each emotion, 100 utterances ($10 \quad sentences \times 10 \quad sessions$) are used for training the models and remaining 50 utterances ($5 \quad sentences \times 10 \quad sessions$) are used for testing. For text-dependent case, for each emotion all 15 text prompts from seven sessions ($15 \quad sentences \times 7 \quad sessions \quad = \quad 105 \quad utterances$) are used for training the models and 45 utterances from remaining three sessions ($15 \quad sentences \times 3 \quad sessions$) are used for validating the trained models.

4 Results and Discussion

4.1 Text-Independent Emotion Recognition

In this paper the speech data of 5 speakers (2 males + 3 females) is analyzed for emotion recognition as the similar results were observed for all the speakers and

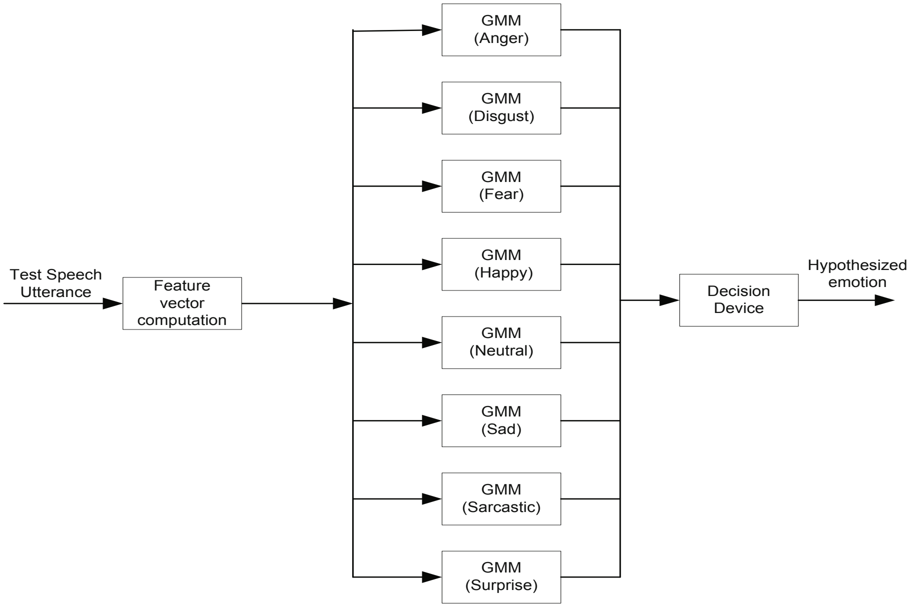


Fig. 2. Block diagram of emotion recognition system using Gaussian mixture models

because of the space constraint it was not possible for us to keep the result of all the speakers and analyze the result. The results obtained from text-independent and text-dependent study are shown in table 7. First column denotes the speaker information (SP1 to SP5), column 2 to column 9 denote the average emotion recognition rate for different emotions (anger, disgust, fear, happy, neutral, sadness, sarcastic, and surprise) corresponding to each speaker. It is observed from the table that the recognition rate for Sadness for all speaker is always more than 76%. Minimum recognition rate was observed for surprise which is around 34% for speaker 2.

4.2 Text-Dependent Emotion Recognition

Text-dependent emotion recognition results are shown in table 7 through columns 10-17. Text-dependent emotion recognition results are better compared to the results of the text-independent case. There is an increase of around 10% in average emotion recognition performance. This observations indicates the influence of text on the performance of emotion recognition. In text-dependent case the recognition rate for sadness for all speakers is always more than 89%. Similar observation is done in text-independent case as well. Minimum recognition rate is observed for sarcastic 52% for speaker 4.

The confusion matrices for both text-independent and text-dependent case are shown in the Tables 2–6.

Table 2. Emotion classification performance for speaker 1; Abbreviations: A-Anger, D-Disgust, F-Fear, H-Happy, N-Neutral, Sa-Sad, S-Sarcastic, Sur-Surprise

	Text-independent(Average: 63.88)								Text-dependent(Average: 77.50)							
	A	D	F	H	N	Sad	Sar	Sur	A	D	F	H	N	Sad	Sar	Sur
Anger	67	4	0	29	0	0	0	0	67	4	9	16	0	4	0	0
Disgust	2	78	0	9	0	0	0	11	4	90	0	4	2	0	0	0
Fear	2	0	62	7	7	20	0	2	0	0	85	0	2	13	0	0
Happy	11	4	0	69	0	0	7	9	9	4	0	83	0	0	4	0
Neutral	0	2	2	4	45	44	2	0	0	2	0	0	56	40	2	0
Sadness	0	2	9	7	7	76	0	0	0	0	0	0	0	100	0	0
Sarcastic	4	7	0	13	11	0	56	9	2	0	0	4	0	0	94	0
Surprise	24	0	0	16	0	0	0	60	33	0	0	16	0	0	2	49

In Table 2, the percentage classification results show that speaker-1 has expressed disgust and sadness more clearly compared to other emotions. Misclassification of anger emotion as happy is generally observed and is obvious as they share similar acoustic properties along arousal dimension.

Table 3. Emotion classification performance for speaker 2; Abbreviations: A-Anger, D-Disgust, F-Fear, H-Happy, N-Neutral, Sa-Sad, S-Sarcastic, Sur-Surprise

	Text-independent(Average: 72.77)								Text-dependent(Average: 83.05)							
	A	D	F	H	N	Sad	Sar	Sur	A	D	F	H	N	Sad	Sar	Sur
Anger	89	0	0	0	0	0	0	11	93	0	0	0	0	0	0	7
Disgust	4	60	2	7	0	0	20	7	7	67	0	0	0	0	22	4
Fear	16	0	76	0	0	9	0	0	2	0	81	0	2	13	0	2
Happy	0	4	0	76	0	0	20	0	2	0	0	87	0	0	11	0
Neutral	0	0	0	2	98	0	0	0	0	0	0	0	100	0	0	0
Sadness	2	0	11	0	9	78	0	0	2	0	4	0	0	94	0	0
Sarcastic	9	22	0	9	4	2	34	20	2	22	0	4	4	2	61	4
Surprise2	4	0	0	0	0	0	2	74	13	0	0	0	0	0	2	85

Speaker-2 is male and he has expressed anger and neutral in more distinguishable manner. The model has confused between disgust, surprise and sarcastic, accordingly mis-classification is also observed in Table 3.

Average emotion recognition of speaker-3 is very high. Fear and disgust are expressed more clearly than other emotions. Out of eight emotions, major misclassifications are observed in case of happy emotion. Table 4 shows the corresponding results.

All emotions of speaker-4 are recognized with similar performance. The extreme passive emotions like neutral and sadness are recognized comparatively better than the other emotions. The results are shown in Table 5

Speaker 5 has efficiently expressed both the extreme emotions such as anger and sadness. The results are shown in table 6.

The results of Table 7 are graphically shown in Fig 3. It can be observed from the figure that recognition rate for text-dependent case observed to be more than the result of text-independent case for all emotions. Emotion recognition rate for disgust and sarcastic are less compared to the other emotions. This

Table 4. Emotion classification performance for speaker 3; Abbreviations: A-Anger, D-Disgust, F-Fear, H-Happy, N-Neutral, Sa-Sad, S-Sarcastic, Sur-Surprise

	Text-independent(Average: 85.00)								Text-dependent(Average: 90.55)							
	A	D	F	H	N	Sad	Sar	Sur	A	D	F	H	N	Sad	Sar	Sur
Anger	84	8	0	0	0	0	0	8	96	2	0	2	0	0	0	0
Disgust	2	92	0	2	0	2	0	4	9	80	0	9	0	0	0	2
Fear	0	0	98	0	0	2	0	0	2	0	92	4	0	2	0	0
Happy	6	8	2	66	10	4	4	0	7	0	0	84	7	0	2	0
Neutral	0	0	0	2	88	10	0	0	0	0	0	2	98	0	0	0
Sadness	0	4	4	4	0	88	0	0	0	2	0	9	0	89	0	0
Sarcastic	0	12	0	4	0	4	78	2	0	2	0	4	0	2	92	0
Surprise	2	12	0	2	0	0	0	86	0	4	0	0	0	0	0	96

Table 5. Emotion classification performance for speaker 4; Abbreviations: A-Anger, D-Disgust, F-Fear, H-Happy, N-Neutral, Sa-Sad, S-Sarcastic, Sur-Surprise

	Text-independent(Average: 71.75)								Text-dependent(Average: 79.16)							
	A	D	F	H	N	Sad	Sar	Sur	A	D	F	H	N	Sad	Sar	Sur
Anger	66	2	2	16	0	2	2	10	87	0	4	7	0	0	0	2
Disgust	0	72	2	0	26	0	0	0	0	76	0	0	24	0	0	0
Fear	0	16	66	0	6	12	0	0	0	20	65	0	13	0	0	2
Happy	4	2	0	64	0	16	10	4	0	0	0	87	0	4	7	2
Neutral	0	6	2	0	92	0	0	0	0	7	2	0	91	0	0	0
Sadness	2	4	2	0	4	82	6	0	0	0	0	0	7	89	4	0
Sarcastic	0	0	0	0	0	4	58	38	0	0	0	4	0	0	52	44
Surprise	2	2	0	4	0	4	14	74	0	2	0	2	0	0	7	89

Table 6. Emotion classification performance for speaker 5; Abbreviations: A-Anger, D-Disgust, F-Fear, H-Happy, N-Neutral, Sa-Sad, S-Sarcastic, Sur-Surprise

	Text-independent(Average: 63.00)								Text-dependent(Average: 80.55)							
	A	D	F	H	N	Sad	Sar	Sur	A	D	F	H	N	Sad	Sar	Sur
Anger	92	0	0	6	0	0	2	0	100	0	0	0	0	0	0	0
Disgust	0	44	0	20	0	2	28	6	0	56	4	7	0	4	22	7
Fear	0	0	44	28	0	16	0	12	0	0	76	0	0	13	0	11
Happy	2	0	0	86	0	4	4	4	0	2	0	81	0	4	2	11
Neutral	4	8	4	18	44	16	2	6	0	9	4	4	72	11	0	0
Sadness	0	0	8	2	6	78	0	6	0	0	2	0	0	96	0	2
sarcastic	0	6	2	24	0	12	48	10	0	2	2	9	0	9	71	7
surprise	2	2	4	16	2	0	6	68	0	0	4	0	0	0	0	96

Table 7. Average Emotion classification performance for all speakers ; Abbreviations: A-Anger, D-Disgust, F-Fear, H-Happy, N-Neutral, Sad-Sadness, Sar-Sarcastic, Sur-Surprise

	Text-independent Average: 72.00								Text-dependent Average: 82.00							
	A	D	F	H	N	Sad	Sar	Sur	A	D	F	H	N	Sad	Sar	Sur
SP1	67	78	62	69	45	76	56	60	67	90	85	83	56	100	94	49
SP2	89	60	76	76	98	78	34	74	93	67	81	87	100	94	61	85
SP3	84	92	98	66	88	88	78	86	96	80	92	84	98	89	92	96
SP4	66	72	66	64	92	82	58	74	87	76	65	87	91	89	52	89
SP5	92	44	44	86	44	78	48	68	100	56	76	81	72	96	71	96

is the general observation, mostly because sarcastic is not the basic emotion and it is difficult to express disgust independently. Therefore, considerable misclassification is observed in case of these two emotions.

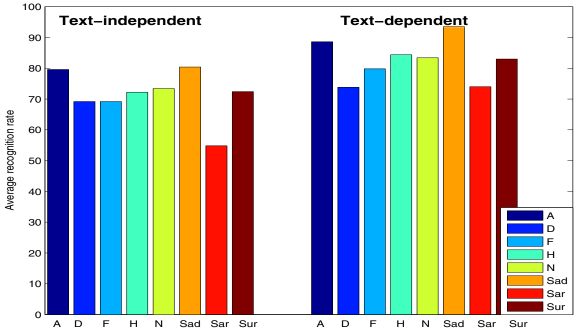


Fig. 3. Emotionwise comparison for text-independent and text-dependent emotion recognition

Fig 4 shows the comparison of average emotion recognition rates with respect to each speaker for both text-independent and dependent cases. These comparisons show that the emotion recognition performance also heavily depends on the speaker information as well as his or her ability to express the emotions. In general, emotion recognition rate for speaker 3 (SP3) is higher compared to other speakers in both text dependent and independent cases. It indicates that the speaker 3 had expressed emotions in a more distinguishable way than the other speakers.

Fig. 5 and Fig. 6 indicate the comparative emotion recognition performance of all speakers for text-dependent and text-independent cases separately. Obviously emotions are recognized better, when supporting textual information is

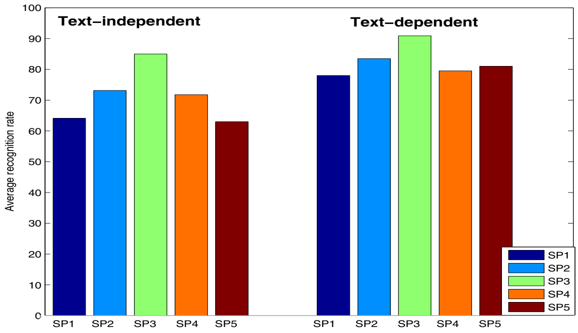


Fig. 4. Speakerwise comparison for text-independent and text-dependent emotion recognition

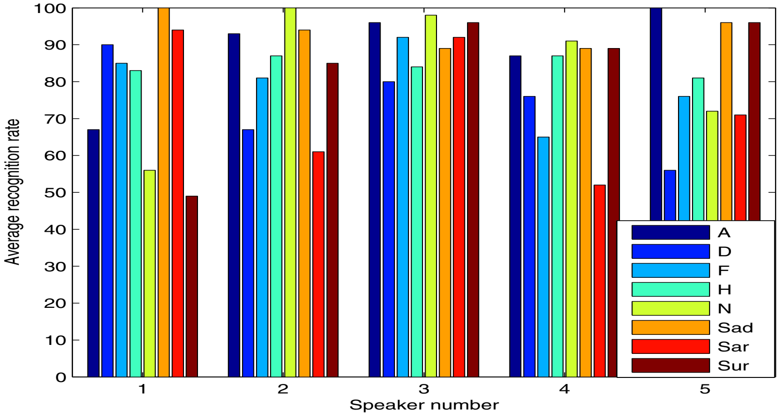


Fig. 5. Text-dependent emotion recognition

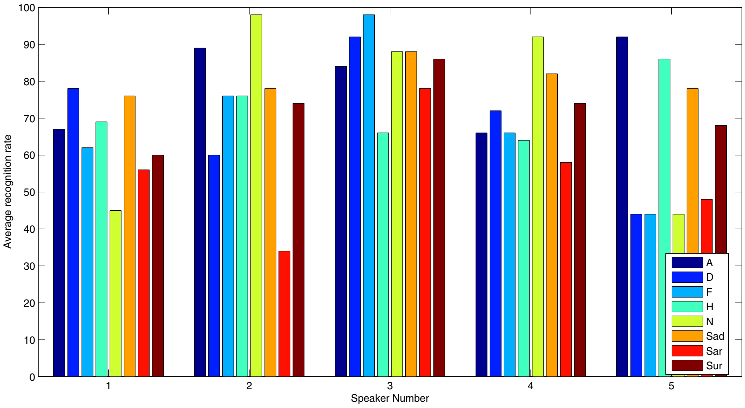


Fig. 6. Text-independent emotion recognition

available. The active extreme emotion like anger, for speaker 2, 3 and 5 (male) is recognized with better performance compared to other emotions. The passive extreme emotions like sadness and neutral for speaker 1 and 4 (female) are recognized with better performance compared to other emotions. These are also the general perceptual observations.

5 Summary and Conclusions

In this paper, text-independent emotion recognition using spectral features has been studied and the results are compared with the results of text-dependent emotion models. Emotional speech corpus, ITKGP-SEHSC, has been used for this study. Emotion recognition rate depends on text, emotion and speaker. Emotion recognition rate in text-dependent case is more compared to the performance of text-independent models. Disgust and sarcastic are not properly recognized, compared to other emotions. The text-independent recognition performance can be improved further by combining the evidence from excitation source and prosodic parameters. Other nonlinear models such as neural networks [11,9] and support vector machines [10] may further enhance the emotion recognition performance.

Acknowledgements

Authors would like to acknowledge IIT Kharagpur for providing funds to collect IITKGP-SEHSC, speech corpus. Our sincere thanks to the students of IIT Kharagpur, who spent their time for subjective evaluation of this database.

References

1. Ververidis, D., Kotropoulos, C.: A state of the art review on emotional speech databases. In: 11th Australasian International Conference on Speech Science and Technology, Auckland, New Zealand (December 2006)
2. Lee, C.M., Narayanan, S.S.: Toward detecting emotions in spoken dialogs. IEEE Trans. Speech and Audio Processing 13, 293–303 (2005)
3. Nakatsu, R., Nicholson, J., Tosa, N.: Emotion recognition and its application to computer agents with spontaneous interactive capabilities. Knowledge Based Systems 13, 497–504 (2000)
4. Sagar, T.V.: Characterisation and synthesis of emotionsin speech using prosodic features. Master's thesis, Dept. of Electronics and communications Engineering, Indian Institute of Technology Guwahati (May 2007)
5. Murray, I.R., Arnott, J.L., Rohwer, E.A.: Emotional stress in synthetic speech. Progress and future directions. Speech Communication 20, 85–91 (1996)
6. Nicholson, T.K., Nakatsu, R.: Emotion recognition in speech using neural networks. Neural Computing and Applications 9, 290–296 (2000)
7. Rabiner, L., Juang, B.H.: Fundamentals of Speech Recognition. Prentice-Hall, Englewood Cliffs (1993)
8. Koolagudi, S.G., Reddy, R.: Yadav, Jainath., Rao, K.Sreenivasa.:IITKGP-SEHSC: Hindi speech corpus for emotion analysis. In: IEEE International Confrence on Device Communication BIT MESRA, India (February 2011)
9. Yegnanarayana, B.: Artificial Neural Networks. Prentice-Hall, New Delhi (1999)

10. Burges, C.J.C.: A tutorial on support vector machines for pat- tern recognition. Data Mining and Knowledge Discovery 2(2), 121–167 (1998)
11. Razak, A.A., Isa, A.H.M., Komiya, R.: A Neural Network Approach for Emotion Recognition in Speech. In: 2nd International Confrence Artificial Intelligence in Engineering and Technology, Kota Kinabalu, Sabah, Malaysia (August 2004)
12. Koolagudi, S.G., Kumar, N., Rao, K.S.: Speech emotion recognition using segmental level prosodic analysis. In: IEEE International Confrence on Device Communication BIT MESRA, India (February 2011)
13. Koolagudi, S.G., Krothapalli, R.S.: Two stage emotion recognition based on speaking rate. International Journal of Speech Technology 14 (March 2011)

Segment Specific Concatenation Cost for Syllable Based Bengali TTS

N.P. Narendra and K. Sreenivasa Rao

School of Information Technology,
Indian Institute of Technology Kharagpur, Kharagpur - 721302, West Bengal, India
{narendrasince1987,ksrao1969}@gmail.com

Abstract. This paper proposes a new method of concatenation cost calculation for enhancing the optimality in unit selection. Instead of defining same set of concatenation costs for all types of speech unit transitions, costs are defined based on the type of unit transitions. Different types of unit transitions that can occur mainly in an utterance are voiced to voiced, voiced to unvoiced and unvoiced to unvoiced transitions. Natural measure of continuity is identified for each of these transitions, and costs are defined accordingly. For voiced to voiced transitions, in addition to spectral continuity, pitch and energy continuity metrics are proposed. In case of voiced to unvoiced and unvoiced to unvoiced transitions, silence duration embedded in the unvoiced region is proposed as the continuity metric. This approach of segment specific concatenation cost calculation improves the quality of syllable based text to speech synthesis. Listening tests provide a proof on the effectiveness of proposed methodology which has clearly shown the decrease in perceptual discontinuity at joins, and improvement in the overall quality of the synthesised speech.

Keywords: Concatenation cost calculation, unit selection, Bengali TTS.

1 Introduction

Text to speech synthesis (TTS) system synthesises the speech for the given input text. Now a days, TTS applications on personal computers and embedded systems which can produce higher quality speech, are in demand. Limited domain TTS systems are being deployed in commercial places such as railway and flight schedule queries, where people can listen to information rather than reading it from screen. Computer screen readers which use TTS can support blind people to access the computer and internet. Concatenative speech synthesis is the predominant approach for achieving good quality natural speech. In this approach, pre-recorded speech waveforms stored in the database are selected and concatenated to produce the desired speech. Choice of basic sound unit (speech segment) is one of the factors which decide the quality of synthesis. Researchers have tried using half phones, phones, diphones, triphones, polyphones, syllables and words as basic units. Selection of each kind of sound unit has their own advantages and disadvantages. Indian languages have a well defined syllable structure. The

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 371–382, 2011.

basic sound units written and pronounced are in the form of syllables. More over syllables can preserve coarticulation effect. So, for building TTS in Bengali, syllables are used as the basic units of synthesis.

Once text is given as input to TTS system, it is split into sequence of sound units. Linguistic and prosody analysis module generate target feature specification for each of the sound unit. Depending on target specification, units are selected from the large database based on unit selection algorithm [1]. The purpose of unit selection algorithm is to select an optimal sequence of units that best matches with the target unit specification and matches with other units in the sequence. Improvements in the unit selection algorithm are done by clustering the units of same type based on the questions related to prosodic and phonetic situations [2]. While selecting the units two costs are defined namely Target cost and concatenation cost. Concatenation cost gives the measure of how two units join without any perceptual distortion. Target cost gives the measure of closeness a unit to the target unit. Viterbi search is used for efficiently searching the best unit sequence which has the least overall cost.

To select an optimal sequence of units which can yield good quality speech close to natural utterance, cost calculations should be done efficiently. Among the two costs, concatenation cost is important as it determines compatibility and appropriateness of speech units to be selected for concatenation during synthesis. Spectral continuity is most widely used continuity metric for concatenation cost computation [3] [4]. For a tone based language like Chinese mandarian, tone is used as one of the concatenation metric [5]. One of the previous approaches have also shown that by designing appropriate concatenation cost alone without taking target cost into account, best sequence of units can be selected [6]. Widely popular, Festival speech synthesis engine by default makes use of optimal coupling technique to find the cost and the position for join [7].

Generally, selection of speech units is done by computing concatenation cost between units at the time of synthesis. Computation of concatenation cost is carried out uniformly for all types of speech units. Following this procedure of uniform cost computation, though may be appropriate for some cases, but need not be best suitable for all cases. Selection of some speech units may be proper as continuity metrics in concatenation costs best define speech unit transitions. Selection of some speech units may not yield natural joins as the continuity metrics may not correlate with the natural continuity metrics at the joins. These facts provide strong motivation for proposing concatenation costs based on different types of speech unit transitions.

In this work, a new method of concatenation cost calculation is proposed for enhancing the optimality in unit selection, there by increasing the quality of synthesised speech. Different types of speech unit transitions that can occur in an utterance are identified. For each of these these transitions, concatenation cost computation is done by defining a separate set of continuity metrics. Continuity metrics are derived by observing the natural instances of speech unit transitions. This way of cost calculation prompts unit selection algorithm to select the units whose joins are close to natural transitions. Listening tests clearly indicated that the proposed concatenation cost metrics have improved the performance of synthesiser.

2 Development of Bengali TTS

Bengali TTS is built using syllable as the basic unit under festival framework. Festival offers general tools for building unit selection synthesiser in any new language. The basic requirement of TTS is text corpus, which consists of 7762 sentences covering 4374 unique syllables and 22382 unique words. Letter to sound rules are developed which indicate how the written text has to be spoken. Syllabification rules are framed. Speech corpus is recorded using the text corpus in neutral emotion. Recording is done in clean noiseless chamber at 16KHz sampling frequency and stored in 16 bit PCM. Recorded sentences are segmented and labeled with syllable identities. Prosody models are built to predict the prosody of target syllables at the time of synthesis. Prosody models include duration and F0 models. Then speech corpus is organised in a systematic manner in the form clusters [2]. At the time of synthesis, the input text will be parsed into the sequence of sound units. For each of the sound units, target specification is provided through linguistic and prosodic features. Depending on target specification, unit selection algorithm selects an optimal sequence of units from the speech corpus by minimising two costs namely concatenation cost and target cost. In this paper optimality in unit selection is improved by proposing a new method in concatenation cost calculation.

3 Proposed Segment Specific Concatenation Cost

Concatenation cost gives an estimate of the quality of join between candidate units u_{i-1} and u_i for two desired targets. Concatenation cost can be split into sub costs where each of the sub costs indicates one continuity metric. Sub costs should be weighted appropriately depending on which continuity measure contributes more to perceptual distortion. Weighted sub costs are added to get the concatenation cost.

$$C^c(u_{i-1}, u_i) = \sum_{j=1}^{q} w_j^c C_j^c(u_{i-1}, u_i) \quad . \tag{1}$$

u_{i-1} and u_i are the candidate units of the (i-1)th and ith targets, q is the total number of sub costs and w_j are the weights given to sub costs. If u_{i-1} and u_i are the consecutive units in speech corpus, then their concatenation is natural and therefore the cost of zero.

Most of the previous approaches have defined a uniform set of concatenation cost for all types of unit to unit transitions. Spectral continuity is a popular measure of continuity between the units. This measure of continuity is suitable for units which have their transition regions voiced. If the units have a transition from voiced to unvoiced regions, then spectral continuity cannot be used to calculate the measure of continuity between the units. So instead of defining uniform set of concatenation cost, different types of possible transitions must be identified and a separate set of concatenation costs should be defined for each type of these unit transitions.

Different types of possible transitions are

1. Voiced to voiced transition
2. Voiced to unvoiced transition
3. Unvoiced to unvoiced transition
4. Unvoiced to voiced transition

To observe the frequency of occurrence of unit transitions, randomly 100 sentences are selected from the corpus. Each of the sentences is split into sequence of syllables. Depending on the type of segments at the syllable joins, unit transitions are grouped into one of the four classes namely voiced to voiced, voiced to unvoiced, unvoiced to unvoiced and unvoiced to voiced. Number of occurrence of each type of unit transitions is counted for all sentences. The frequency of occurrence of each of unit transitions is expressed in percentage and they are shown in Table 1.

Table 1. Frequency of occurance of different types of unit transitions

Type of transition	Frequency of occurance(%)
Voiced to voiced	64.7
Voiced to unvoiced	29.4
Unvoiced to unvoiced	5.5
Unvoiced to voiced	0.3

3.1 Voiced to Voiced Transition

When voiced to voiced transitions are observed, there will be continuity in energy, pitch and spectrum. According to these, a set of concatenation costs are proposed for voiced to voiced transitions.

1. Acoustic distance
2. Energy difference
3. Voiced f0/pitch difference
4. Derivative of energy difference
5. Derivative of pitch difference

Acoustic distance gives the measure of spectral continuity at the point of join. Euclidean distance between the MFCCs of last frame of previous unit and MFCCs of first frame of present unit is considered as the acoustic difference.

Abrupt energy difference at the point of join is one of the factors that contribute to perceptual distortion. Difference between the energy of last frame of previous unit and the energy of first frame of next unit is computed. Energy difference as one of the subcost ensures smooth variation of energy at the syllable joins. Pitch cannot vary drastically between the syllables and even between the words. So pitch difference of frames at the point of join is calculated as a measure of pitch continuity. To ensure that energy and pitch contours are varying smoothly around the point of join, two more subcosts namely derivative of energy difference and pitch difference are proposed.

Weighted sum of all subcosts yields to get concatenation cost. Deriving optimal weights is also an issue. Weights should be determined in such a way that the synthesised waveform is close to the natural waveform. Objectively, determining the weights is difficult as there is no direct relation between sub-costs and human perception. Hence weights are heuristically chosen and synthesised waveforms are perceptually observed. Set of weights which yield more natural sounding speech are chosen as weights of subcosts.

3.2 Voiced to Unvoiced Transition

In general transition from voiced to unvoiced speech segments, continuity of energy, pitch and spectral characteristics are not observed. Voiced speech segments include vowels, semivowels and voiced consonants, whereas unvoiced speech segments consist of unvoiced stop consonants and fricatives. As there is no continuity of conventional features such as energy, pitch or spectrum, these cannot be used as continuity metric for analyzing transition between voiced and unvoiced segments. Natural instance of voiced to unvoiced transition is observed to choose the measure of continuity between units. Voiced to unvoiced transition region is characterised by short duration of silence followed by a sudden burst of noise. This short duration of silence is implicit in the unvoiced sound. This silence duration is not given intentionally by speaker, and it is due to the production constraints of the specific sound units. Unvoiced stops are produced by constriction in the vocal tract for short period followed by sudden release of air pressure [8]. This short duration of silence or pause is not same for all stop consonants. It depends on the speaking rate of speaker, syllable context (previous and present syllables), position in word, number of syllables present in the word and so on. At the time of synthesis, if the unit selection algorithm is not taking care of duration of silence while picking up the unvoiced stop sound units, then unvoiced stop sounds are not perceived properly and it perceives like distortion. In a sentence, on an average 4-5 stops are present. If there is an error in picking the units with right amount of pause or silence duration, then in addition to perceptual distortion the synthesised sentences appears to be spoken at a faster rate. So consideration of proper duration of silence is necessary while picking the unvoiced stop sound units for synthesis.

At the time of synthesis, amount of silence duration required for the desired unvoiced stop consonant has to be determined by appropriate prediction model. In this work, linear regression model is used for predicting the duration of silence for the unvoiced stop consonant sounds. Difference between predicted silence duration and original silence duration for unvoiced stop consonants present in the database is found. This difference is weighted appropriately and used as the concatenation cost between voiced to unvoiced transition. If the predicted silence duration is very close to calculated silence duration, cost is less and subsequently there is more chance of picking up those sound units.

3.2.1 Silence Duration Prediction Module

Silence duration prediction is done by building linear regression models. Before building models, voiced to unvoiced transitions are divided into different classes depending on the position of occurrence of voiced to unvoiced transition in the word, type of segment that contributes to voiced sound and type of syllable (having different types of stop consonants and vowels) that contributes unvoiced sounds. Linear regression models are built for each of the classes based on the factors that are influencing the silence duration embedded in the unvoiced sound. Since only few factors are used, the size of training corpus need not be very large. Thus the prediction model does not encounter any data sparsity problem, which often happens in using other models, such as neural networks and classification and regression tree models.

In voiced to unvoiced transition, voiced region is due vowel or voiced consonants. Unvoiced region is due to stop consonants. Syllables which contain unvoiced regions may be open syllables (syllables of type CV) or closed syllables (syllables of type CVC). Based on the above, there can be four types of voiced to unvoiced transitions.

1. Vowel to open syllable
2. Vowel to closed syllable
3. Voiced consonant to open syllable
4. Voiced consonant to closed syllable

Based on the type of stop consonant and vowel present at the nucleus of syllable, open syllable can be further classified as ka, kaa, ki, kii..etc. Closed syllables can be classified as KaC, KaaC, KiC..etc where C is any consonant. If all stop consonants are considered, the total number of closed and open syllables will be 80. All 80 classes are considered in each of the above four broad classes, making a total of 320 classes. Linear regression models are built for each of the classes.

Unvoiced region can occur at the beginning of the word (at the first syllable) or within the word (other than first syllable). As the properties and duration of silence, vary greatly on the position of unvoiced region within the word. Based on the position of unvoiced region in the word whether it is at the beginning of the word or within the word, different approaches are followed in building regression models. Given a voiced to unvoiced transition, first classification is done into one of the 320 classes. Then, based on the occurrence of transition in the word different approaches are used for building models.

3.2.2 Linear Regression Models for Unvoiced Region Present within the Word

For predicting silence or pause duration embedded in the unvoiced sounds, relative ratios are used instead of absolute values. This is because, the entire speech corpus is not recorded in a single session, hence there may exist some variation in the speaking rate. We define the pause ratio of pause (PR)

$$PR = \frac{DP}{\frac{1}{NSw}\sum_{j=1}^{NSw}(DSj)} \quad . \tag{2}$$

Where PR is the pause ratio, DP is duration of pause or silence embedded in unvoiced region, DSj duration of jth syllable in the word, NSw is the total number of syllables in the word. Here we are concerned with the syllable rate of word, as the prediction for silence duration is done within the word.

At the training stage, average pause ratios are found based on four factors. Firstly, average pause ratio is found by considering all examples of the class. Then average pause ratios are found based on the position of unvoiced sound in the word (i.e., Unvoiced region can occur at second, third, fourth syllable and so on). Average pause ratios are computed based on the number of syllables in the word. Average pause ratios are found based on the word position in the utterance (beginning, middle and end position in the utterance). From the pause ratios obtained from four factors, linear equation is derived by connecting all four pause ratios to approximate the original pause ratio values using linear regression technique. Average syllable duration based on the number of syllables in the word are computed and stored, which is used at the time of testing.

At the testing stage, for a given voiced to unvoiced transition, pause ratios are derived based on the above mentioned four factors and combined using linear regression equation to get the predicted pause ratio. Average syllable duration obtained based on the number of syllables in the word is multiplied with predicted pause ratio to get the predicted silence duration. For example, if we want to predict silence duration occurring within the word "kathaa" which is a two syllable word in the utterance "tini aara kono kathaa". First, average pause ratio is found out for the class having syllable transition from vowel ('a' in "ka") to open syllable "thaa". Average pause ratio with unvoiced sound at the second syllable is found within the class. Average pause ratio of two syllable word having this class of voiced to unvoiced transition is found. Average pause ratio is found considering only the last word in the utterance (based word position in utterance) which contains specified class of voiced to unvoiced transition. These four pause ratios are combined using the derived linear regression equation. Pause ratio obtained from linear regression method is multiplied with average syllable duration of two syllable word to get predict silence duration.

Models are built using a training data of 6000 sentences and tested with the data of 1000 sentences. Average errors are obtained by calculating the difference between original silence duration and predicted silence duration from the test data. As it is difficult to represent average errors for all 320 classes due to lack of space, average errors are represented for four broad classes as shown in Table 2.

Table 2. Average errors of four broad classes for unvoiced region present within the word

Unit transitions	(% error)
Vowel to open syllable	12.4
Vowel to closed syllable	13.5
Voiced consonant to open syllable	12.0
Voiced consonant to closed syllable	11.1

3.2.3 Prediction Model for Unvoiced Region Present at Beginning of the Word

In predicting silence or pause duration embedded in unvoiced region, relative ratios are used instead of absolute values for the same reason as mentioned in the previous section. Pause ratio of pause (PR)

$$PR = \frac{DP}{\frac{1}{NSu}\sum_{j=1}^{NSu}(DSj)} \ . \tag{3}$$

Where PR is the pause ratio, DP is duration of pause or silence embedded in unvoiced region, DSj duration of jth syllable in the utterance, NSu is the total number of syllables in the utterance. Here we are interested in the syllable rate of entire utterance, as prediction is done for silence duration of unvoiced sound present at beginning of the word, in other words between two words.

At the training stage, average pause ratios are found based on five factors. Firstly, average pause ratio is found by considering all examples within the class. Then average pause ratios are computed based on the number of syllables in the word present just before the word containing unvoiced stop and based on the number of syllables in the word which contains unvoiced stop at its first syllable. Average pause ratios are found based on the position of word in the utterance and based on the number of words in the utterance. From the pause ratios obtained from five factors, linear equation is derived connecting all five pause ratios to approximate the original values using linear regression technique. Average syllable duration based on the number of syllables in the utterance are computed and stored which is used at the time of testing.

At the testing stage, pause ratios are derived based on the five factors and combined using linear regression equation to get the predicted pause ratios. Average syllable duration obtained based on the number of syllable in the utterance is multiplied with the predicted pause ratio to get the predicted silence duration. Same as in previous section, models are built using a training data of 6000 sentences and tested with the data of 1000 sentences. Average errors for the four broad classes are given in Table 3.

3.3 Unvoiced to Unvoiced Transition

For predicting the silence duration between unvoiced to unvoiced transitions, the same procedure is used as in voiced to unvoiced transition. Here also, according

Table 3. Average errors of four broad classes for unvoiced region present at beginning the word

Unit transitions	(% error)
Vowel to open syllable	10.3
Vowel to closed syllable	9.2
Voiced consonant to open syllable	11.3
Voiced consonant to closed syllable	8.8

to the occurrence of transition, prediction is done for two cases 1) unvoiced region present within the word and 2) unvoiced region present at beginning of the word. Classification of unvoiced to unvoiced transition is done slightly different. Frequency of occurrence of this transition in an utterance is very less. If classification is done as mentioned for voiced to unvoiced transition, then most of the classes will be having zero or very less number examples. Instead a broader way of classification is done. In unvoiced to unvoiced transition, both unvoiced regions are due to any of the stop consonants. As the silence duration embedded in stop consonant depends more on the present stop than the previous one, classification is done as unvoiced to each of the stop consonant transition (k, kh, ch, chh,...). If all stop consonants are considered, then total numbers of possible classes are 10. Linear regression models are built for each of the classes. Rest of the procedure in prediction module remains same as for voiced to unvoiced transition. Average errors of two broad classes for unvoiced to unvoiced transition is shown in Table 4.

Table 4. Average errors of two broad classes for unvoiced to unvoiced transition

Transition position	(% error)
Unvoiced region at beginning of word	9.6
Unvoiced region within the word	9.8

3.4 Unvoiced to Voiced Transitions

As the frequency occurrence of unvoiced to voiced transitions is very less in the whole database. So we are not defining any specific continuity metrics for unvoiced to voiced transitions.

4 Evaluation of the Proposed Segment Specific Concatenation Cost

Bengali TTS is developed by incorporating the proposed segment specific concatenation cost in unit selection. Developed Bengali TTS is evaluated in two stages. At the first stage of evaluation, perceptual listening tests are conducted to measure the improvements in the perceived continuity at the concatenation points. At the second stage, overall quality of synthesised speech is evaluated through subjective listening tests. In both stages of evaluation, default target cost present festival is used. Listening tests are conducted with 10 research scholars in the age group of 22-35 years.

4.1 Evaluation of Perceived Continuity at the Concatenation Points

In the first stage of evaluation, listening tests are conducted slightly in a different way. Subjects are first shown the written form of each sentence, with an indication of different types of joins. At first, sentences are synthesised using only spectral continuity as concatenation cost and played to subjects. Subjects were asked to count the number of perceived discontinuities in the synthesise sentences. Subjects could listen to each sentence as many times as they want. Then after incorporating proposed concatenation cost calculation, synthesised sentences are played to subjects and were asked to count the number of perceived discontinuities. In this study, voiced to voiced transitions are grouped together as voiced joins. Voiced to unvoiced and unvoiced to unvoiced transitions are grouped as unvoiced joins, since for both transitions; silence duration embedded in the unvoiced region is used for concatenation cost calculation. Subjects are asked to count the number of discontinuities perceived at voiced joins and unvoiced joins for each of the above situations.

As our interest is on whether the proposed method has improved perceptual continuity, in other words decreased perceptual discontinuity, results got from listening tests are tabulated in the form of percentage decrease in the discontinuity. Results are given in the Table 5. Second column gives decrease in discontinuity for voiced joins for each of the sentences. Third column shows decrease in perceptual discontinuity for unvoiced joins. Fourth column gives the overall decrease in discontinuity. Last row provides overall average values from all ten sentences. In the table, 100% means all the perceptual discontinuities present in synthesised sentences are reduced and 0% means none of the perceptual discontinuities are reduced after incorporating proposed costs. From the table, it is observed that there is an overall decrease in discontinuity in each of the sentences. This fact confirms that the proposed method has improved the quality of synthesised sentences. On further keen observation reveals that percentage of decrease in discontinuity for voiced joins is not as much as compared to unvoiced joins. As the first stage uses spectral continuity, most of the units which have voiced joins are picked up properly. There is no continuity metric specific to unvoiced joins in the first stage. So, most of the discontinuities are in unvoiced joins. After applying the proposed method, there is significant decrease in discontinuities at unvoiced joins and slight decrease in discontinuities at voiced joins. The method has tried to pick up the best possible units from the speech corpus. Further accurate prediction of silence duration and deriving some more continuity metrics which can best define unvoiced joins can make the proposed method more efficient.

Table 5. Percentage decrease in discontinuity for ten sentences

Sentence No	Voiced joins(%)	Unvoiced joins(%)	Overall(%)
1	0	50	20
2	66.7	100	75
3	0	100	50
4	50	**	50
5	66.7	100	80
6	**	50	50
7	0	50	33.3
8	100	100	100
9	0	50	33.3
10	0	100	50
Total	31.48	77.77	54.16

** No discontinuity was perceived.

4.2 Evaluation of Overall Quality of Synthesised Speech

In the second stage of evaluation, subjects are asked to judge distortion and quality of speech on a 5-point scale for each of the sentence. The 5-point scale for representing the quality of speech and the distortion level is given in Table 6. At first step (step-1), subjects are asked to rate sentences, synthesized using only spectral continuity as concatenation cost. Then at second step (step-2), after incorporating proposed concatenation cost metrics, subjects were asked to give their scores for synthesised sentences.

Mean opinion scores (MOS) obtained from two steps of evaluation is given in the Table 7. From the table, it can be observed that MOS scores after step-2 is higher than step-1. This confirms that the overall quality of speech has increased. Subjects noticed that the naturalness and intelligibility has been increased with incorporation of proposed concatenation cost calculation into Bengali TTS.

Table 6. Instructions to evaluators

Score	Subjective perception
1	Poor speech, with distortion and very low intelligibility
2	Poor speech with distortion and intelligible
3	Good speech with less distortion and intelligibility
4	Very good speech quality with less naturalness
5	As good as natural speech

Table 7. Mean opinion scores

Evaluation step	MOS
Step-1	3.05
Step-2	3.41

5 Conclusion

In this work, new set of concatenation cost metrics are proposed to enhance the optimality in unit selection. Different types of unit transitions are identified and natural continuity metrics for each of the transitions are found. For voiced to voiced joins, in addition to spectral continuity, pitch and energy continuity metrics are proposed. In case of voiced to unvoiced and unvoiced to unvoiced joins, silence duration embedded in the unvoiced region is proposed as the measure of continuity. The proposed concatenation cost metrics improved the quality of syllable based text to speech synthesis. This is confirmed by listening tests conducted on synthesised sentences where overall decrement in perceptual discontinuity at concatenation points is observed. Further accurate prediction of silence duration prediction and deriving some more continuity metrics which can best define different types of joins can improve efficiency of proposed method.

References

1. Hunt, A.J., Black, A.W.: Unit selection in a concatenative speech synthesis system using a large speech database. In: Proceedings of IEEE Int. Conf. Acoust., Speech, and Signal Processing, vol. 1, pp. 373–376 (1996)
2. Black, A.W., Taylor, P.: Automatically clustering similar units for unit selection in speech synthesis. In: Eurospeech 1997, vol. 2, pp. 601–604 (1997)
3. Karabetsos, S., Tsiakoulis, P., Chalamandaris, A., Raptis, S.: One-class classification for spectral join cost calculation in unit selection speech synthesis. IEEE Signal Processing Letters 17(8), 746–749 (2010)
4. Vepa, J., King, S.: Join cost for unit selection speech synthesis, pp. 35–62. Prentice-Hall, NJ (2004)
5. Dong, M., Lua, K.T., Li, H.: Unit selection-based speech synthesis approach for mandarian chinese. Journal of Chinese Language and Computing, 135–144 (2006)
6. Blouin, C., Rosec, O., Bagshaw, P.C., d'Alessandro, C.: Concatenation Cost Calculation and Optimization for Unit Selection in TTS. In: IEEEWorkshop on Speech Synthesis, SantaMonica CA, USA (2002)
7. Conkie, A., Isard, S.: Progress in speech synthesis. Progress in speech synthesis (1997)
8. Benesty, J., Sondhi, M.M., Huang, Y.: Springer Handbook of Speech Processing. Springer, Heidelberg (2008)

A Service Profile Based Service Quality Model for an Institutional Electronic Library

Ash Mohammad Abbas

Department of Computer Engineering,
Zakir Husain College of Engineering and Technology
Aligarh Muslim University, Aligarh - 202002, India
am.abbas.ce@amu.ac.in

Abstract. Devising a scheme for evaluating the service quality of an institutional electronic library is a difficult and challenging task. The challenge comes from the fact that the services provided by an institutional electronic library depend upon the contents requested by the users and the contents housed by the library. Different types of users might be interested in different types of contents. In this paper, we propose a technique for evaluating the service quality of an institutional electronic library. Our scheme is based on the service profiles of contents requested by the users at the server side which is hosted at the library. Further, we propose models to analyze the service quality of an electronic library. For analyzing the service quality, we present two analytical models. The first one is based on the number of days by which the item to be served by the library is delayed and the penalty points per day for the duration for which the item is delayed. The second model is based on the credits earned by the library if the item is served in a timely fashion, and the penalties, thereof, if the item is delayed. These models may help in evaluating the service quality of an electronic library and taking the corrective measures to improve it.

Keywords: Service quality, service profile, eLibrary, delay based model, credit based model.

1 Introduction

The proliferation of Internet, specifically, the World Wide Web (WWW) has made a tremendous impact on the society in terms of what we need, how we act, and our habits. For example, instead of going to a physical library, we now wish to retrieve the contents at our desktops, laptops, or on mobile devices. We wish to have ubiquitous access to the Internet irrespective of whether we are moving or even traveling from one part to the other part of the world. If we want to read a book, we want that the book should be readily available at the device we are using at the moment. This has led to a concept of electronic library (eLibrary), where the contents can be downloaded to the device by simply clicking the mouse or just pressing some buttons.

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 383–394, 2011.

The physical libraries in todays world cannot survive if they do not provide the contents electronically. Therefore, in addition to housing physical books, a physical library should also possess electronic books (ebooks), videos, lecture slides, thesis, reports, journals in electronic form (ejournals), etc. The library has to be connected to the Internet so that internal users (or users from inside the institute) and external users, including those from different parts of the world, may have an access to the resources and contents of the library. In other words, a modern library has to act as a content provider, rather than traditionally providing only books, journals, reports, and thesis, all in physical form.

Many of the researchers have focused on what types of contents should be provided by an eLibrary. In [1], the impact of query correlation and query semantics on the information retrieved from online digital libraries is described. A set of guidelines and criteria for selecting the electronic resources is available at [2]. The way, the libraries use the print and electronic resources is discussed in [11]. The design of a middleware for building adaptive systems, called DISSelect-based Adaptive System (DISSAS), has been described in [3], which can be used to enable adaptation in web-based applications and legacy information systems. In [6], an e-content selection method using multiple criteria analysis in web-based personalized learning environments is described. The work in [9] presents a timely and keyword-based dynamic content selection for public displays. In [4], a user driven content selection scheme for digitization of Ebooks-on-Demand (EoD) networks is presented.

A conceptual model in the form of a questionnaire called *ServQual* to evaluate the service quality of a system was presented in [8]. The ServQual was, originally, comprising of ten aspects of service quality, namely, *reliability*, *responsiveness*, *competence*, *access*, *courtesy*, *communication*, *credibility*, *security*, *understanding the customer*, and *tangibles*. It was aimed to measure the gap between customer expectations and experience. Later, in [12], the model was refined to contain only five attributes: **r***eliability*, **a***ssurance*, **t***angibles*, **e***mpathy*, and **r***esponsiveness*; and was renamed as RATER, which is an acronym for the set of attributes it contains. Since their introductions, the ServQual and RATER are used for assessing the service quality in different fields such as health care, business, financial, marketing, etc. The questionnaires of ServQual or RATER can also be used for determining the service quality of a library. However, these tools ServQual and RATER are generalized to evaluate the service quality of any system, and not specifically of a library system. A specific tool for evaluating the service quality of a library, called "LibQUAL+®" is described in [13]. Item sampling in service quality assessment surveys to improve the response rate and reduce the burden on the respondent using a tool called "LibQUAL+® Lite" is studied in [14]. A brief comparison of these tools is presented in Table 1. A common feature of these tools is that the assessment of service quality of an underlying system is based on the outcomes of the surveys.

In [7], an assessment of the service quality of Thammasat University Library System is studied using a modified version of the questionnaire of ServQual. The authors, therein, use a concept of *zone of tolerance*. They conducted a survey on different classes of users such as undergraduate students, graduate students, faculty members, and researchers. They consider organizational, access, and personal effects on the service quality of the library by manually counting and categorizing the problems that users face in each of the three categories. The effect of individual differences such as gender and status of a private university library namely, Independent University Bangladesh Library, is investigated in [10]. The authors, therein, carried out a survey using a modified version of the questionnaire of ServQual, and their findings suggest that scores of different classes of users may differ based on their gender and status. An assessment of service quality at central library of Management and Planning Organization (MPO), Iran, is carried out in [5]. Their study focuses on the factors related to the library environment, information dissemination, and library personnel. They suggest that a focus on the training and development of library staff may help in providing better services.

However, most the evaluations of service quality of a library (e.g. [7] , [10], [5]) are carried out for physical libraries. Not much work is available in the literature for electronic libraries. As we mentioned earlier, there is a paradigm shift from physical libraries to electronic libraries. However, we expect that physical libraries shall continue to exist, at least in the near future, due to many reasons. For example, one reason can be that not all physical contents can be converted to their truly electronic form, and the kind of entertainment by visiting physical libraries and looking at physical objects cannot be attained by simply watching their electronic form. The only thing is that many of the users would like to get the electronic contents, however, some other users may prefer to visit the library physically depending upon the availability of time and depending upon tightness their schedules. Libraries should serve both classes of users in the best possible manner, as a result, their comes a concept of the service quality of a library, be it electronic or physical, which is the theme of this paper.

In this paper, we propose analytical models for evaluating the service quality of an institutional electronic library. Our models are based on the service profiles of contents requested by the users. Specifically, we propose two models for the service quality. The first model is based on the duration by which an item to be served by the library is delayed and the penalties thereof. The second model is based on the credits if the items requested by users are served in a timely fashion and the penalties if the items to be served by the library are delayed. Our model is not based on any survey, however, it is based on how the requests are handled by the library. In other words, our model is based on the requests served by the library and the service quality is evaluated on the basis of the requests served.

The rest of this paper is organized as follows. In section 2, we describe the notion of service profiles for the requests received by the library for specific contents. In section 3, we present analytical models for service profile based service quality. In section 4, we present results and discussion. The last section is for conclusion.

Table 1. A Comparison of The Tools for Service Quality Assessment

Tools	*Basis*	*Features*	*Comments*
ServQual [8]	Survey	10 aspects of service quality	Generalized to any system
RATER [12]	Survey	5 aspects of service quality	Generalized to any system
LibQUAL+® [13]	Survey	Item Sampling	Specific to Library

2 Service Profiles

In this section, we propose a service profile based scheme for evaluating the service quality provided by the eLibrary.

A library maintains the service profile about the services provided to users and also to different categories of users so as to improve its services in the future. The service profile contains the information about the services provided by the eLibrary. Specifically, a service profile of an eLibrary contains the following open ended set of attributes.

<*RequestID, RequestTime, UserID, ContentID, ContentType, ContentHits, ContentAvailStatus, ContentDeliveryTime, ArrangementStatus, NotificationStatus, NotificationTime, UserAcceptance, ReasonsNotDelivered, ExcessDelay*>.

The attribute *RequestID* is an identifier for the request generated by an end-user. The attribute *RequestTime* represents the time of the reception of the user request by the eLibrary. The attribute *UserID* is an identifier of the user who generated the request. The attribute *ContentID* is an identifier for the content requested by the user, and the attribute *ContentType* represents the type of the content the user has requested such as physical book, ebook, video, ppt slides, journal, tutorials, reports, thesis, etc. The attribute *ContentHits* represents the number of user requests received for a specific content within a specified observation time. The attribute *ContentAvailabilityStatus* tells whether the content is available or not available. If the content is available, then the attribute *ContentDeliveryTime* tells the time when the content is delivered to the user. Otherwise, the attribute *ArrangementStatus* tells whether the content will be arranged/procured by the eLibrary or not. If the arrangement/procurement of the content will be carried out by the eLibrary, then the expected time the arrangement/procurement is going to incur. The attribute *NotificationStatus* tells whether the notification is sent to the user or not, and the attribute *NotificationTime* represents the time when the notification was sent to the user informing him about the arrangement/procurement. The attribute *UserAcceptance* represents whether the user agrees to the time taken by the eLibrary in arrangement/procurement of the content. If the content is not delivered at all, then the reasons are recorded for not delivering the content to the user in the *ReasonsNotDelivered* field. The attribute *ExcessDelay* represents the delay in excess to what has been agreed between the user and the eLibrary.

In what follows, we present models for analyzing the service profiles based quality of service provided by an eLibrary.

3 Analysis of Service Profiles

For analyzing the service quality of an eLibrary, we present two models: (i) delay based service quality model, and (ii) credit based service quality model. We describe each of them as follows.

3.1 Delay Based Service Quality Model

This model is based on the absolute delays (say, in days) between the day on which the request was made by the user or the item was due to be delivered, and the day on which request was actually serviced.

Let τ be the *ExcessDelay*, in number of days, incurred after the expiry of the expected time of delivery as notified by the eLibrary to the user, and p be the penalty points per day assigned by the library itself, with the viewpoint to evaluate the service quality provided by the eLibrary. Let $\phi(p, \tau)$ be the service quality of the eLibrary with parameters p and Δ and let it be given by the following expression.

$$\phi(p, \tau) = 1 + pe^{-p\tau} \tag{1}$$

where, $p \geq 0$, $\tau \geq 0$. In this model, the maximum value of the service quality is,

$$\phi_{\max} = 1 + p. \tag{2}$$

The maximum value of service quality occurs when the parameter $\tau = 0$. The minimum value of the service quality is $\phi_{\min} = 1$ and occurs at $p = 0$. The service quality is calculated for all requests the eLibrary receives and then the average value of the service quality can be determined by taking the average over all requests considered, which is expressed as follows.

$$\bar{\phi} = \frac{\sum_{i=1}^{n} \phi_i}{n}. \tag{3}$$

To discuss how the service quality varies with the variations in the parameters p and τ, we need to compute the partial derivatives of the service quality with respect to these parameters. We compute the partial derivatives of the service quality with respect to the parameters p and τ in the following lemma.

Lemma 1. *The partial derivative of the service quality with respect to τ is as follows.*

$$\frac{\partial \phi}{\partial p} = (1 - p^2)e^{-p\tau} \tag{4}$$

and,

$$\frac{\partial \phi}{\partial \tau} = -p^2 e^{-p\tau}. \tag{5}$$

Proof. The partial derivative of the service quality with respect to p is as follows.

$$\begin{aligned}\frac{\partial\phi}{\partial p} &= pe^{p\tau}(-p) + e^{-p\tau}\\ &= -p^2e^{-p\tau}\\ &= (1-p^2)e^{-p\tau}.\end{aligned} \tag{6}$$

Similarly, the partial derivative of the service quality with respect to the excess delay τ is as follows.

$$\begin{aligned}\frac{\partial\phi}{\partial \tau} &= pe^{p\tau}(-p)\\ &= -p^2e^{-p\tau}.\end{aligned} \tag{7}$$

Let the variation in the service quality with respect to p be $\varDelta\phi_p$, and the variations in the service quality with respect to the variations in τ be $\varDelta\phi_\tau$, then the overall variations in the service quality is as follows.

$$\varDelta\phi = \frac{\partial\phi}{\partial p}\varDelta\phi_p + \frac{\partial\phi}{\partial\tau}\varDelta\phi_\tau. \tag{8}$$

In the above model, the penalty for an item delivered late has been incorporated and the effective penalty varies with the duration by which the item is delivered late. However, the above model does not take into account any credits for the items delivered in time.

We now present a credit based service quality model.

3.2 Credit Based Service Quality Model

Let H be the number of requests served in order (i.e. on the same day or on or before the day mutually agreed between the eLibrary and the user). Let L be the number of requests served late (i.e. after the mutually agreed day or time between user and the eLibrary). Let there be c number of credits assigned for each request served in time. If the time of service of the request is delayed, then a penalty p is imposed on to the eLibray. Note that the total number of requests received by the eLibrary is the summation of the number of requests served in time and the number of requests served late, i.e. $H + L$. We now define the service quality as follows.

$$\phi = \frac{cH - pL}{(c+p)(H+L)} \tag{9}$$

where, $-1 < \phi < 1$. If $c = p = q$, then the expression of the service quality becomes as follows.

$$\phi = \frac{H-L}{2(H+L)} \tag{10}$$

Note that when $c = p = q$, and $H = L$, we have $\phi = 0$; and if $L = 0$, $\phi = \frac{1}{2}$; similarly, if $H = 0$, $\phi = -\frac{1}{2}$. We can now say that for $c = p$, $-\frac{1}{2} \leq \phi \leq \frac{1}{2}$.

To discuss how the service quality varies with the number of requests served in a timely fashion and the number of requests served late, we need to compute the partial derivatives of the service quality with respect to H and L, respectively. We prove the following lemma about the derivatives of the service quality with respect to the number of requests served late as well as the number of requests served in a timely fashion.

Lemma 2. *The partial derivatives of the service quality with respect to the number of requests served late as well as with respect to the number of requests served in a timely fashion are given by,*

$$\frac{\partial \phi}{\partial H} = \frac{L}{(H+L)^2} \tag{11}$$

and

$$\frac{\partial \phi}{\partial L} = -\frac{H}{(H+L)^2}. \tag{12}$$

Proof. Using the law of division, the partial derivative of the service quality, as defined by (9), with respect to H, is as follows.

$$\begin{aligned}\frac{\partial \phi}{\partial H} &= \frac{(c+p)(H+L).c-(cH-pL).(c+p)}{\{(c+p)(H+L)\}^2} \\ &= \frac{c(c+p)H+c(c+p)L-(c+p)cH+(c+p)pL}{\{(c+p)(H+L)\}^2} \\ &= \frac{(c+p)^2L}{\{(c+p)(H+L)\}^2} \\ &= \frac{L}{(H+L)^2}.\end{aligned} \tag{13}$$

Similarly, the partial derivative of the service quality with respect to L is given by,

$$\begin{aligned}\frac{\partial \phi}{\partial L} &= \frac{(c+p)(H+L).(-p)-(cH-pL).(c+p)}{\{(c+p)(H+L)\}^2} \\ &= \frac{-p(c+p)H-p(c+p)L-c(c+p)H+(c+p)pL}{\{(c+p)(H+L)\}^2} \\ &= \frac{-(c+p)^2H}{\{(c+p)(H+L)\}^2} \\ &= -\frac{H}{(H+L)^2}.\end{aligned} \tag{14}$$

From the expressions (11) and (12), it is clear that partial derivatives do not depend on the number of credits, c, or penalty points, p, and depend only on how many requests were served in a timely fashion and how many requests were served late.

One can utilize the partial derivatives to compute the variation in the service quality. Let $\Delta\phi_H$ be the variation in the service quality due to variations in the number of requests served in time, and $\Delta\phi_L$ be the variation in the service quality due to variations in the number of requests served late. Then, the overall variation in the service quality is as follows.

$$\Delta\phi = \frac{\partial\phi}{\partial H}\Delta\phi_H + \frac{\partial\phi}{\partial L}\Delta\phi_L. \tag{15}$$

In order to find the maximum and/or minimum values of the service quality, the derivatives have to be equal to 0. Using (11), we have,

$$\frac{\partial\phi}{\partial H} = 0.$$

Or,

$$\frac{L}{(H+L)^2} = 0.$$

This gives rise to $L = 0$. The second derivative of service quality for $L = 0$ comes out to be $+ve$, signifying that at $L = 0$, there is a maxima for the service quality. Putting $L = 0$ in (9), we get,

$$\begin{aligned}\phi_{\max} &= \frac{cH}{(c+p)H} \\ &= \frac{c}{c+p}.\end{aligned} \tag{16}$$

For $c = p$, we have, $\phi_{\max} = \frac{1}{2}$. In other words, when the number of credits per request served in time is equal to the number penalty points per request served late, then the maximum value of the service quality is $\frac{1}{2}$.

Similarly, equating the partial derivative of service quality given by (12) to 0, we get $H = 0$. At $H = 0$, the second derivative of the service quality is $-ve$, therefore, there is a minima for the service quality at $H = 0$. Putting $H = 0$ in (12), we get the minimum value of the service quality which is as follows.

$$\begin{aligned}\phi_{\min} &= -\frac{pL}{(c+p)L} \\ &= -\frac{p}{c+p}.\end{aligned} \tag{17}$$

For $c = p$, we have, $\phi_{\min} = -\frac{1}{2}$. The minimum and maximum values of service quality, as given by (16) and (17), confirm our earlier argument that $-\frac{1}{2} \leq \phi \leq \frac{1}{2}$ for credit-based service quality model.

In what follows, we present results and discussion.

4 Results and Discussion

The eLibrary gathers information about the service profiles of different types of contents provided by the eLibrary. Based on the information gathered for a certain period of the observation time, statistical analysis of the service profiles is performed so as to improve the service quality provided by the eLibrary to its users. The information gathered in a manner described above is analyzed. The library keeps track of how many hits were made by users with different profiles and how many requests were timely satisfied and for how many requests arrangements/procurements from else where were made and how many requests were not satisfied at all. What type of requests were the most frequent and what type of requests were less frequent. The analysis of service profiles of the contents requested by the users and provided by the eLibrary is carried out in order to evaluate the service quality provided by the eLibrary to its users. Based on the average value of the service quality, measures can be adopted to improve the service quality of the eLibrary.

Let us examine the how the service quality varies in the delay based model. Figure 1 shows the service quality as a function of the number of days by which the service of requests is delayed, where the number of penalty points for each request is one per day or two per day. We observe that as the delay in the number of days is increased, the service quality decreases exponentially. Also, we observe that service quality decreases more rapidly if the number of penalty points is increased from one penalty point per day to two penalty points per day. Note that when the number of penalty points per day is 1, the maximum value of the service quality is 2, and when the number of penalty points per day is $\phi_{\max} = 2$, the maximum value of the service quality is $\phi_{\max} = 3$.

Figure 2 shows the service quality as a function of the number of penalty points for each request served late, where the requests are delayed by one day or two days. We observe that the service quality is 1 for the number of penalty points equal to 0, and after that it reaches to its normal value at penalty points equal to 1. After that, the service quality starts decreasing with the number of penalty points for a given number of days by which the request is late.

However, the delay based service quality model uses only one parameter, namely, penalty points, and more is the number of penalty points per day, the maximum value of the service quality is set higher. This seems reasonable in the sense that if the decrement for the service quality (which is the number of penalty points per day), the maximum value from which the service quality should start decreasing, is set higher as compared to the situation where the decrement in the service quality is relatively small as there is no way to increase the service quality. In other words, the larger value of the number of penalty points per day also plays the role of implicit credits: higher the penalty, larger is the value of the maximum service quality. There are no explicit credits. This is analogous to a bankers cash: larger the rate of withdrawal from the bank, more cash the banker should have with himself/herself to start with. It is possible that a banker giving away money to his/her customers at a higher rate may finish his/her start money more rapidly as compared to the one from whom rate of withdrawal is smaller and who starts with a smaller money.

We now examine the service quality in the credit based model, where the library is assigned explicit credits when the requests are served in a timely

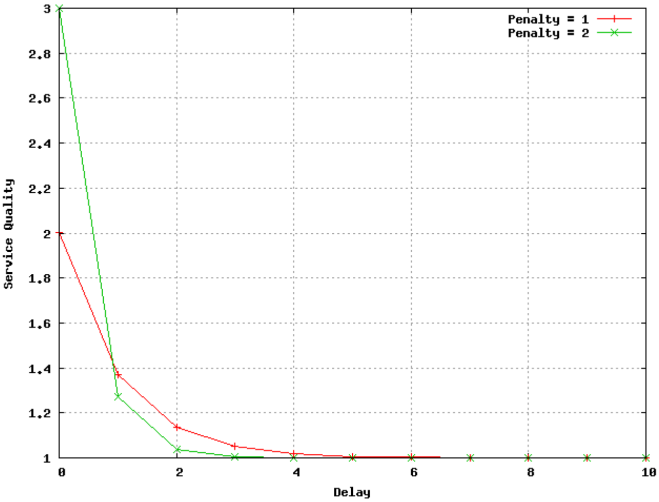


Fig. 1. Service quality as a function of the number of days by which the service of requests is delayed, where the number of penalty points for each request is one per day or two per day (*Delay Based Model*)

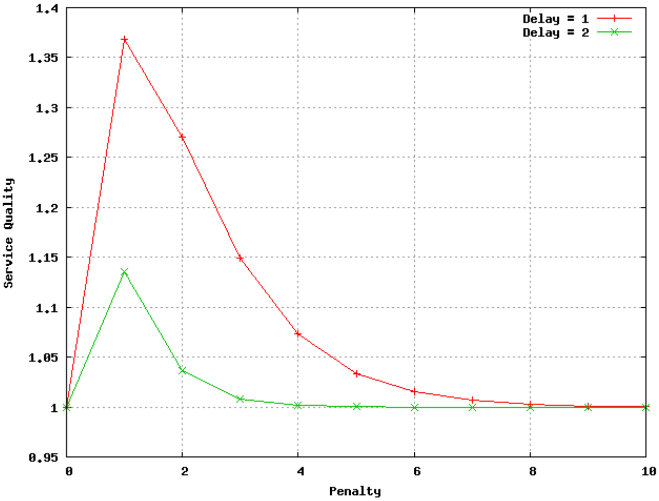


Fig. 2. Service quality as a function of the number of penalty points for each request served late, where the requests are delayed by one day and two days (*Delay Based Model*)

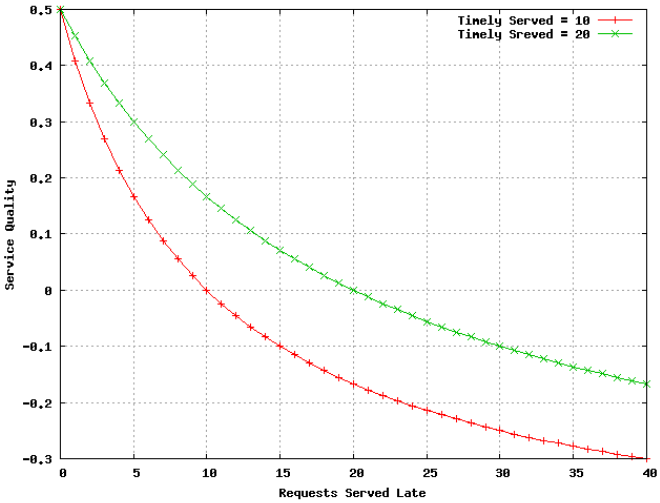


Fig. 3. Service quality as a function of the number of requests served late, where the number of requests served in a timely manner is 10 and 20 (*Credit Based Model*)

fashion, and explicit penalty points when the item to be served by the library is late. Figure 3 shows the service quality of an eLibrary as a function of the number of requests that were served late for the credit based model. The number of requests served in a timely manner is taken to be 10 and 20. We observe that as the number of requests served late is increased, the service quality the eLibrary decreases. At a certain point in time, the service quality becomes negative. It means that the service quality has been deteriorated significantly and corrective measures should be taken to improve the service quality of the eLibrary. In what follows, we conclude the paper.

5 Conclusion

Devising a scheme for evaluating the service quality of an institutional electronic library is a difficult and challenging task. The challenge comes from the fact that the services provided by an institutional electronic library depend upon the contents requested by the users and the contents housed by the library. Different types of users might be interested in different types of contents. In this paper, we propose a technique for evaluating the service quality of an institutional electronic library. Our scheme is based on the service profiles of contents requested by the users at the server side (i.e. service profiles are maintained at the server of the eLibrary and not at the side of the end user). For analyzing the service quality, we presented two analytical models. The first one is based on the number of days by which the item to be served by the library is delayed and the penalty points per day. The second model is based on the credits earned by the library if the item is served in a timely fashion, and the penalties if the item is delayed. These models may help in evaluating the service quality of the eLibrary and taking the corrective measures to improve it.

References

1. Abbas, A.M.: The Impact of Query Correlation and Query Semantics on Online Digital Libraries. In: Proceedings of International Conference on Digital Libraries and Knowledge Organization (ICDK), Gurgaon, pp. 1–10 (2011)
2. Criteria for Selecting Electronic Resources. Library and Technology Services, Brandies University (2001), `http://lts.brandeis.edu/about/policies/collection/selecting.html`
3. Gallucci, L., Cannataro, M., Palopoli, L., Veltri, P.: DISSAS: A DISSelect-based Middleware for Building Adaptive Systems (2006), `http://www.win.tue.nl/~acristea/A3H/../7-4-6-A3H-06-cannataro-revised.pdf`
4. Gstrien, S., Muhlberger, G.: User Driven Content Selection for Digitization the Ebooks on Demand Network. In: Proceedings of International Conference on Cultural Heritage Online, pp. 1–6 (2009), `http://www.rinascimento-digitale.it/eventi/conference2009/proceedings-2009/grstein.pdf`
5. Hassanzadeh, M., Sharifabadi, S.R., Derakhshan, M.: Assessment of Service Quality at Central Library of Management and Planning Organization (MPO). Iran. International Journal of Information Science and Management 8(1), 107–118 (2010)
6. Manouselis, N., Sampson, D.: Dynamic Educational e-Content Selection using Multiple Criteria Analysis in Web-based Personalised Learning Environments. In: Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications, pp. 1213–1218 (2002)
7. Nimsomboon, N., Nagata, H.: Assessment of Library Service Quality at Thammasat University Library System. A Research Report, pp. 1–73 (2003), `http://www.kc.tsukuba.ac.jp/div-comm/pdf/report0403.pdf`
8. Parasuraman, A., Zeithaml, V.A., Berry, L.L.: A Conceptual Model for Service Quality and its Implications. Journal of Marketing 49, 41–50 (1985)
9. Ribeiro, F.R., Jose, R.: Timely and Keyword-Based Dynamic Content Selection for Public Displays. In: Proceedings of IEEE International Conference on Complex, Intelligent and Software Intensive Systems (CISIS), pp. 655–660. IEEE Press, Los Alamitos (2010)
10. Shoeb, M.Z.H., Ahmed, S.M.Z.: Individual Differences in Service Quality Assessment: A Study of a Private University Library System in Bangladesh. Emerald Journal on Performance Measurement and Metrics 10(3), 193–211 (2009)
11. Shorten, J.: What do Libraries Really do with Electronic Resources? The Practice in 2003. In: Fenner, A. (ed.) Integrating and Digital Resources in Library Collections. Haworth Information Press, Binghamton (2006)
12. Zeithaml, V.A., Parasuraman, A., Berry, L.L.: Delivering Quality Service: Balancing Customer Perceptions and Expectations. Free Press, New York (1990)
13. Cook, C., Heath, F.: Users' Perceptions of Library Service Quality: A "LibQUAL+®" Qualitative Study. Library Trends 49, 548–584 (2001)
14. Thompson, B., Kyrillidou, M., Cook, C.: Item Sampling in Service Quality Assessment Surveys to Improve Response Rates and Reduce Respondent Burden: The "LibQUAL + ® Lite" Example. Emerald's Journal on Performance Measurement and Metrics 10(1), 6–16 (2009)

Improving Fading-Aware Routing with Circular Cache Layers in Wireless Sensor Networks

Sudhanshu Pant, Naveen Chauhan, Narottam Chand,
L.K. Awasthi, and Brij Bihari Dubey

Department of Computer Science and Engineering
National Institute of Technology Hamirpur, India
sudhanshupant1986@gmail.com, naveenchauhan.nith@gmail.com,
dubey.brijbihari@gmail.com, nar@nitham.ac.in, lalit@nitham.ac.in

Abstract. In this paper, we propose a cooperative caching scheme that exploits an energy efficient model in Wireless Sensor Network. Cooperative caching is done in the form of concentric circular cache layers around the sink. A Circular cache layer is a group of nodes falls under circumference of the circle formed from the sink as center with a certain radius. At a time only a single cache layer becomes active which caches the data in it. A fading-aware routing is utilized to minimize the total power consumption during packet transfer. The scheme performs well in the multiple sink environments. Proposed scheme is compared with the existing Fading-aware energy efficient routing approach. Simulation results show the performance and better efficiency of the network.

Keywords: Caching, Cooperative caching, Circular Cache layer, WSN, Energy Efficiency.

1 Introduction

The recent development in sensor nodes equipped with flash memory is giving new direction in designing and deploying the energy efficient Wireless Sensor Network. In large scale sensor network there are thousands of sensor nodes distributed over a vast field [1]. In a WSN the node that gathers the data information refers to sink. The sink may be connected to the outside world through internet where the information can be utilized within time constraints.

A lot of research in data routing [2], data compression [3] and in-network aggregation [4] has been proposed in recent years. In this paper a new data caching technique is proposed which caches data nearer to the sink. The optimal way to cache data near to the sink is the circular layered area around the sink. Our scheme caches data in the nodes surrounded by the sink in the circular cache layers. Each concentric circle forms a cache layer and the data gets cached in these circular cache layers starting from the innermost circle. Each concentric circular cache layer gets a token which decides which cache layer will caches the data. The Circular Cache layers (CCL) are formed by the Circular Cache Formation (CCF) algorithm. The algorithm searches for the geographical coordinates of the nodes around the sink, by flooding

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 395–402, 2011.

request messages, to determine there distance form the sink. The nodes which fall under a particular distance from the sink forms a cache layer. The discovery of cached data is operated by a simple cache discovery scheme. Finally, a data replacement policy is given which helps in removing obsolete data from the caches.

Rest of the paper is organized as follows. Section II describes the related work done so far in the cooperative caching. In section III, the concept of proposed scheme is discussed without giving the details of how the scheme works. Section IV gives the explanation of the presented scheme describing query and data processing within the sensor nodes. Section V gives the simulation and analysis including comparison analysis and results. Conclusion out of exploiting the proposed scheme is discussed in Section VI.

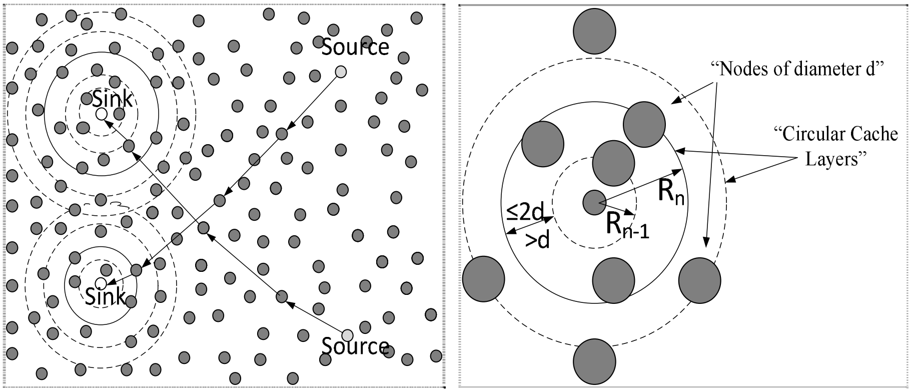
2 Related Work

Several works has been proposed by the authors exploiting caching the data either in some intermediate nodes or at a location nearer to the sink in the Wireless Sensor Networks. Indeed providing solutions to optimally caching the data has been a big area to be focus on, several proposed schemes performs well. Jinbao Li *et al.* [5] proposes a caching scheme for the multi-sink sensor network. The sensor network forms a set of network trees per sink. The common subtree is formed out of these set and the root of the common subtree is selected as the data caching node to reduce the communication cost. GroCoCa [6] suggests a data item to be cached depends on two factors of the access affinity on the data items and the mobility of each node. Depending upon the mobility pattern of the mobile nodes they are tightly coupled to form a group. The Cluster Cooperative protocol [7] given by N. Chand *et al.* attempts to form non-overlapping clusters based in geographical proximity. The idea is to partition whole Mobile Adhoc Network (MANET) into equal size cluster and for a cache miss in the local cache of the node, each client looks for the data item in the cluster. In [8] implementation of TCP support for the WSNs and schemes to avoid end to end transmission between the nodes by caching TCP segments and by local retransmission of TCP segments has been proposed. N. Chauhan [10] *et al.* proposes scheme called Global Cluster Cooperation strategy in Mobile Adhoc Networks, which proposes a methodology of keeping the information regarding each and every nodes contents at cluster level.

3 System Environment

3.1 Network Model

Our scheme assumes Sensor field comprises of large number of sensor nodes capable of communicating each other through wireless medium. Each sensor nodes is aware of there geographical coordinates (x, y), which are also the node identities. A typical setup of the sensor network is shown in Figure 1(a). The Figure 1(a) shows two sinks are requesting data from two different sources. The CCLs of two sinks may overlap.



(a) (b)

Fig. 1. (a) Typical Setup of Wireless Sensor Network , (b) Formation of Circular Cache Layers

In such case the overlapped node is a part of both sink cache. This paper exploits the Fading-aware routing as proposed by J. Levendovszky *et al.* [9]. The communication path setups for the two sinks are shown by the arrows. The active cache layer is shown with the dark circles.

3.2 Cooperative Caching

Circular Cache Layers are utilized to cache the data from the sources. The CCLs are the concentric circles of sensor nodes with the sink as centre and data gets stored starting from the centre and then to the next CCL taking one at a time. The CCF algorithm builds the CCLs around the sink. The CCF algorithm forms the Cache Circular Layers for each and every sinks and the CCL having the token will cache the data in it. As soon as the cache of all the nodes in the active CCL becomes full the token is passed to the next successive CCL. Figure 1(b) depicts the formation of Cache Circular Layers. The radii for two adjacent cache layers are selected as per the size of sensor node. For simplicity we consider circular sensor node and Cache Circular Layers are formed according to the size of the sensor nodes.

4 Query and Data Processing

This section presents the design of the caching scheme. The whole process occurs in following steps:

1. Each sink builds there CCL around its location. The CCF algorithm builds the CCLs for each and every sink.
2. Sink queries the required data through flooding the request around it.
3. The requested data gets provided to the sink either by the cache or directly through the source if there is a cache miss due to cache invalidation or a fresh request.

4.1 Fading-Aware Communication Model

J. Levendovszky *et al.* [9] proposes a Fading-aware reliable and energy efficient routing for optimal path selection and energy balancing of the bottleneck node [9]. The path selected for communication between the source and the sink is described by the set R = (i_1, i_2,i_3,.... ,i_L). The generic fading model says that if d is the distance of packet transfer and g is the transmission power then probability $P^{(r)}$ of correct reception of transmitting the packet can be given as

$$P^{(r)} = \psi\ (\mathrm{d}, \mathrm{g}) \tag{1}$$

where ψ (d, g) is a strictly monotone increasing as g is increased and it should be strictly monotone decreasing function of d. If G_{ij} is the energy consumption in transmitting a packet from node i to node j and d_{ij} is the distance between them then the transmission energy consumption for each packet will be $G_{ij}\Delta T$, where ΔT is the time it takes for each packet to send.

4.2 Circular Cache Layer

To provide data nearer to the sink, the proposed circular cache layer scheme provides the best solution. The CCLs are formed as per the specified radius of the concentric circles in the CCF algorithm. The sink itself is considered as layer 0 cache. Layer 1 circular cache is formed by distributing message identifying the radius R1. All the nodes covering radius R1 and not including the layer 0 node, falls under the layer 1 cache. Similarly all the nodes covering radius R2 and not including the layer 0 and layer 1 cache nodes forms the layer 2 cache and so on. To store the data items, initially layer 0 cache becomes active and called as Active Cache Layer (ACL). To activate an ACL a token is passed to all the nodes of the ACL. As soon as all the cache becomes full, the token is passed to next successive layer and that layer becomes the ACL. By the time the highest CCL will get full, some entries in the lower level CCL may get free due to time-to-live value associated with each data items. After filling up all the nodes in the last layer, the next layer which becomes ACL is layer 0 then layer and so on. In this round the nodes which are free gets filled. If no nodes are found free then a cache invalidation policy is needed to replace the data items. The cache invalidation rules are discussed later in this section in cache management scheme. Each layers of cache consist of increasingly varying number of nodes. Not all nodes in the active cache layer have free cache memory. Rather than waiting for the total number of nodes in cache layer to become free, we give emphasis on the amount of free available cache in the layer in question.

Circular Cache Formation (CCF) Algorithm:

```
for I from 1 to max_cache_layer
    sink floods message with radius = $R_i$ and sink co-
    ordinates (x, y)
    node n ($x_n$, $y_n$) receives the message
    if$\sqrt{\{(Xn-X)^2 + (Yn-Y)^2\}} \le Ri$ && n != lower layer node.
        mark n as layer I node.
```

```
            forward the received message.
        else discard message.
        end if
    end for
```

The max_cache_layer is the measure of the number of queries generated per unit time interval and the size of the cache. If m queries are generated per unit time and it needs an average of n Bytes to store the query reply then the sensor network is producing m*n Bytes of data per unit time. Figure 1(b) gives an idea about how radius of particular CCL is selected. For building two consecutive CCLs we must have

$$D > R_n\text{-}R_{n-1} \leq 2d \tag{2}$$

This value is taken so as to make efficient CCLs. The maximum number of nodes that can be in a layer n can be given as:

$$N = K * 4 * [R_n^2 - R_{n-1}^2]/d \tag{3}$$

where, $0<K\leq1$ and d denotes the approximate physical diameter of a sensor node, considering nodes are spherical in shape. The value of K denotes the density of nodes. We can set max_cache_layer parameter by using this theoretical result. The maximum numbers of cache layers are analyzed more in analysis and simulation part.

4.3 Query Processing

To make a query, sink floods the query request message to its cache layers one by one. If the data item is found in first layer, it is retrieved from that layer node. Otherwise the query request message is passed to next layer. The query may arrive at a cached sensor node in the higher cache layer and the sensor node will reply with the cached results. If none of the cache layer has the desired data then the query is flooded into the whole network, of course in the forward direction, to retrieve the desired result directly from the source node.

The optimal path setup for the query is done by the Fading-aware routing [9]. The overall energy consumption for a packet transfer by the set R = (i_1, i_2,i_3,.... ,i_L) is given by $\sum_{l=1}^{L} Gi_l i_{l+1}$, where $G_{i_L S}$ is the last transfer from node i_L to the sink. The optimization is expressed as

$$R_{opt} : \frac{min}{R} \sum_{l=1}^{L} G_{i_l i_{l+1}} \tag{4}$$

Each query request message contains the identity of the active cache layer in its header field. When the request is replied back to the sink it gets cached in the active cache layer also.

4.4 Cache Management

In this section, the issues related to caching the data are discussed. As there could be multiple sinks in the sensor network, a node could be involved in caching the data for more than one sink. Each node maintains the information about various caches layer to which it belong for the corresponding sinks.

There is a Time to Live (TTL) value associated with each data which denotes its expiry time. This is the responsibility of the source to set this TTL value with each data. The source sets the TTL value with each data item by calculating the frequency of updates in the data. The idea is to set larger value if frequency of updating is low and smaller values for the data where frequency of updating in high. Simulation has shown to increase in performance by using the above mentioned strategy.

If it is required to invalidate the data entries from the cache due to lack of space then following rules are applied.

1. Access Frequency (AF): The access frequency gives the measure of the number of times the data was accessed per unit time. For the data item to be replaced, the data having lowest access frequency should be replaced.
2. Time-to-Live: The item with shorter TTL remains valid for shorter period is the best item to replace. But it may have higher access frequency so it is likely that this data item will further get accessed many times in future. Based on this factor, the product AF*TTL give the parameter to decide which item should be replaced.
3. Distance from the source (D): The distance of this node from the source node gives another constraint to decide which item should be replaced. If the distance is less from the source node then the query request will have to go very short and thus saves time and network traffic as compared to distant node. So the item with minimum distance from the source node is the beat option to replace.

Based on above parameters the importance of data item is computed as follows:

$$\mathrm{Imp}_i = AF * TTL * D \tag{5}$$

The item with the least Imp_i value will get replaced if it is required to replace a data item from the sensor nodes' cache.

5 Simulation and Analysis

We have simulated the scheme in ns-2 (version 2.34). In this section we evaluate the performance of our scheme.

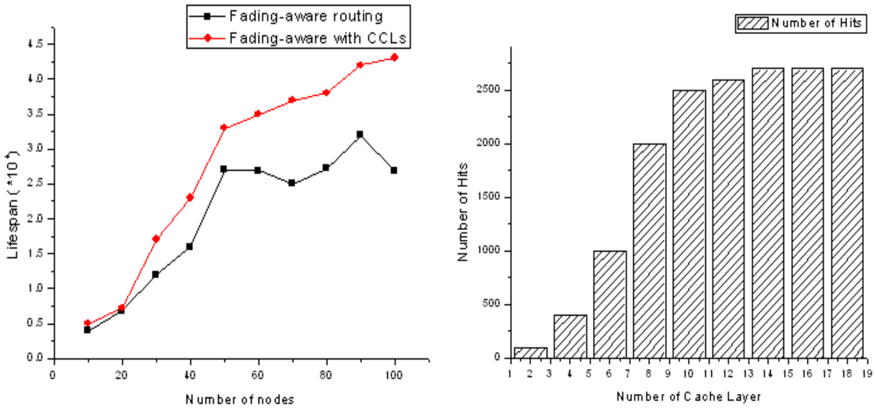
5.1 Simulation Parameters

Sensor field is region of 100 m^2, with 100 sensor nodes deployed in it randomly. The sensor nodes can be in one of these modes: sleeping, sending message, receiving message. For simplicity, sizes of both query and data packets are taken as 64bytes and energy parameters for node are as follows: ET=0.0010 J and ER=0.0008 J, where ET is energy consumed in transmitting a data/query message to one hop and ER is the energy consumed in receiving a data/query message from one hop neighbor. The power consumption during the sleeping mode is set at 0.016 mW. The sensor nodes are assumed to equip with the flash memory of 128 kB.

5.2 Results and Analysis

Effect of Circular Cache Layers on Lifespan of nodes. The Figure 2(a) shows the effect of utilizing the CCLs in the Fading-aware routing [9]. The lifespan is the measure of time interval from the beginning of the operation until the first node dies. The scheme is tested for upto 100 nodes. The Figure 2(a) shows effect on the lifespan of Fading-aware routing with and without utilizing caching. CCLs improve the lifespan of the nodes as the data are directly retrieved from the cache instead of from source which not only avoids obsolete network traffic but also saves energy consumption of the nodes.

Effect of number of cache layers and cache size. The number of cache layers greatly affects the number of hits in the CCLs. Figure 2(b) shows the number of hits when taking different numbers of cache layers. Initially Circular Cache scheme shows less number of cache hit due to less number of nodes involved in caching. As the number of cache layers increases more number of nodes gets involved in caching and hence results in large number of cache hits. The number of cache hits increases as the number of cache layers increases. More number of cache layer i.e., above a certain level, may results in non-increasing cache hits due to the fact that the sink doesn't generate more queries beyond a certain number. Circular Caching scheme shows 62% overall energy saving as compared to the scenario when no caching is used.



(a) (b)

Fig. 2. (a) Network Lifespan, (b) Cache Hits per Cache layers

6 Conclusion

In this paper technique that uses Circular Cache Layers to improve the performance of the Wireless Sensor Networks is discussed. The Circular Cache Layers provides the data items nearer to the sink and thus reduce the response time of the queries. The query results get stored in the active cache layer before reaching to the sink. The Fading-aware routing selects the optimal communication path between source and the sink and thus helps in reducing the energy consumption. We can make an energy efficient WSN only when the load is equally distributed in the network so that all the nodes consumes power equally and network becomes operational as long as possible. The proposed caching scheme can further be enhanced to increase the network performance. Simulation results show the overall network efficiency by means of various graphs. The proposed works is shown to perform well if applied practically in real world scenario under particular situations by the means of proper simulations.

References

1. Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cayirci, E.: A survey on sensor networks. IEEE Communications Magazine 40(8), 102–114 (2002)
2. Abbasi, A., Younis, M.: A survey on clustering algorithms for wireless wireless sensor networks. ACM Journal of Computer Communications 30(14-15), 2826–2841 (2007)
3. Kimura, N., Latifi, S.: A Survey on Data Compression in Wireless Sensor Networks. In: International Conference on Information Technology: Coding and Computing, vol. 2, pp. 8–13 (April 2005)
4. Fasolo, E., Rossi, M., Widmer, J., Zorzi, M.: In-network aggregation techniques for wireless sensor networks: a survey. IEEE Wireless Communications 14(2), 70–87 (2007)
5. Li, J., Li, S., Zhu, J.: Data Caching Based Queries in Multi_sink Sensor Networks. In: IEEE 5th International Conference on Mobile Ad-hoc and Sensor Networks (2009)
6. Chow, C.Y., Leong, H.V., Chan, A.T.S.: GroCoca: Group-based peer-to-peer cooperative caching in mobile environment. IEEE Journal on Selected Areas in Communications 25(1) (January 2007)
7. Chand, N., Joshi., R.C., Misra, M.: Cooperative caching strategy in mobile ad hoc networks based on clusters. Springer Wireless Personal Communications (1), 41–63 (2006)
8. Dunkels, A., Alonso, J., Voigt, T.: Distributed TCP Caching for Wireless Sensor Networks. In: Proceedings of the 3 rd Annual Mediterranean Ad-Hoc Networks Workshop (2004)
9. Levendovszky, J., Tran_thanh, L., Treplan, G., Kiss, G.: Fading-aware reliable and energy efficient routing in wireless sensor networks. Elsevier Computer Communications (2010)
10. Chauhan, N., Awasthi, L.K., Chand, N., Joshi, R.C., Misra, M.: Global Cluster cooperation strategy in Mobile Adhoc Networks. International Journal of Computer Science and Engineering 2(7) (2010)

Nonlinear Channel Equalization for Digital Communications Using DE-Trained Functional Link Artificial Neural Networks

Gyana Ranjan Patra[1], Sayan Maity[2], Soumen Sardar[2], and Swagatam Das[2]

[1] Dept of ECE, ITER, S'o'A University, Bhubaneswar
[2] Dept of ETCE, Jadavpur University, Kolkata
gyanapatra@iter.ac.in, sayanmaity.10@gmail.com,
soumenit@gmail.com, swagatamdas19@yahoo.co.in

Abstract. A major hindrance in the way of reliable and lossless communication is the inter symbol interference (ISI). To counter the effects of ISI and to have proper & reliable communication an adaptive equalizer can be employed at the receiver end. This paper considers the applications of artificial neural network structures (ANN) to the channel equalization problem. The problems related with channel nonlinearities and can be effectively subdued by application of ANNs. This paper contains a new approach to channel equalization using functional link artificial neural network (FLANN). In this paper we have incorporated the novel idea of utilizing an evolutionary technique called Differential Evolution (DE) for the training of FLANN we have compared the results with back propagation (BP) and Genetic Algorithm (GA) trained FLANNs. The comparison has been drawn based upon the minimum Mean Square Error (MSE) and Bit Error Rate (BER) performances. From this study it is evident that the DE trained FLANN performs better than the other types of equalizers.

Keywords: Channel equalization, Functional Link Artificial Neural Network, Differential Evolution (DE), Genetic Algorithm (GA).

1 Introduction

The increase in the demand of multimedia and internet technology related applications have given rise to the necessity of develop efficient and reliable data transmission methods over digital communication channels. The transmission of digital signals is usually carried out over band limited channels for efficient use of spectrum. However these channels experience ISI owing to their dispersive nature and multipath propagation [1],[2],[3].

The ISI can be curtailed down with the application of a proper channel equalizer at the receiver. Equalizers employing finite impulse response (FIR) filters trained with least mean square (LMS) or recursive least squares (RLS) algorithm do not perform satisfactorily for highly nonlinear and dispersive channels. On the contrary due to the

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 403–414, 2011.

presence of nonlinear signal processing in the neural networks (NN) these networks can perform nonlinear mapping between high dimensional input and output spaces [4],[5],[6].

FLANNs are mostly trained by back propagation (BP) algorithm and thus they get easily trapped in the local optima for nonlinearly separable classification problems. The convergence speed using BP learning is also very slow for a given termination criterion. Also the convergence greatly depends upon the initial choice of weights, the learning rate and momentum.

The evolutionary algorithms like genetic algorithm (GA) and Differential evolution (DE) can be utilized for training the weights of ANNs and FLANNs. These algorithms have been suitably used for global optimization problems and proved to be robust with respect to the noisy evaluation functions [7].

In this paper we consider a trigonometric FLANN for the purpose of equalization of nonlinear channels. The training of these FLANNs is done with BP, GA and DE algorithms. The corresponding mean square error (MSE) and bit error rate (BER) performances are compared.

The rest of the paper is organized as follows. In section 2 we have presented an overview of the channel equalization problem. In Section 3 we have briefly described FLANNs which have been used in the study. In Section 4 we have outlined a short description about LMS, GA and DE algorithms. Section 5 deals with simulation studies. The performances of all the equalizer structures are compared in section 6. Section 7 describes the conclusion of this study.

2 Communication Channel Equalization

The schematic of a digital communication system with an equalizer at the front-end of the receiver is shown in Fig. 1. The symbol $\{t_k\}$ denotes a sequence of T-spaced symbols of a BPSK constellation.

$$t_k = \pm 1\,, \tag{1}$$

where the symbols +1 and -1 are assumed to be statistically independent and equiprobable. In Fig. 1, the combined effect of transmitter-side filter and transmission medium are included in the channel.

An FIR filter whose output at the k^{th} instant is given by (2) is widely used to model linear dispersive channel.

$$a_k = \sum_{i=0}^{N-1} h_i t_{k-i}\ . \tag{2}$$

where N is the order of the channel. Since the channel is a nonlinear one, the ‘‘NL’’ block introduces channel nonlinearity to the filter output. The discrete output of the nonlinear channel is given by

$$b_k = \psi\{a_k, a_{k-1}, a_{k-2}.....a_{k-N-1}; h_0, h_1, h_2, ...h_{N-1}\}\,, \tag{3}$$

where, $\psi(\cdot)$ is a nonlinear function created by “NL”.

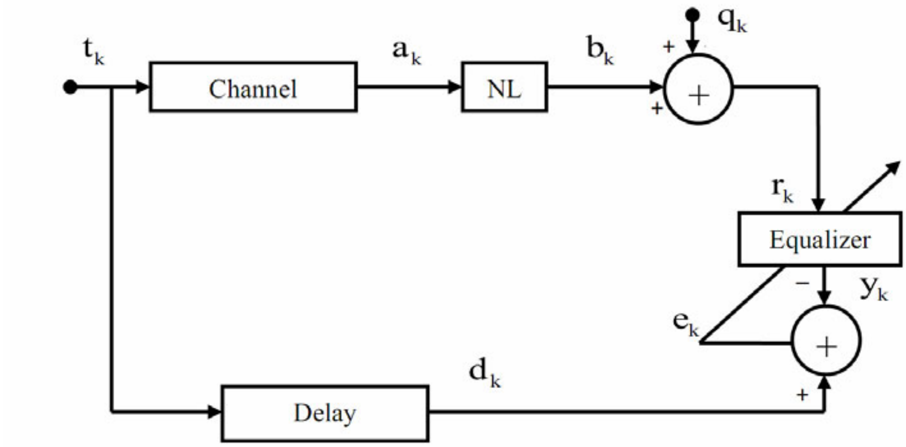


Fig. 1. Schematic diagram of a channel equalizer employed in a communication channel with an adaptive equalizer

The channel output is corrupted with an additive Gaussian noise q_k with a variance of σ^2. The transmitted signal t_k is received as r_k at the receiver.

The equalizer present at the receiver front-end tries to recover the transmitted sequence t_k from d_k $(= t_{k-\tau}$ where τ is the propagation delay associated with the physical channel). During training, the adaptive equalizer (e.g., a FIR filter) takes r_k and its delayed versions (corrupted sequence) and produce y_k. It then tries to reduce the error $e_k = d_k - y_k$ by suitably updating the filter weights using an adaptive algorithm (e.g., LMS algorithm). After the training equalizer weights are fixed and then these weights are used to estimate the transmitted sequence.

3 FLANNs for Channel Equalization

The FLANN [8] shown in Fig. 2 is a single layer network in which the hidden layers are removed. In contrast to the linear weighting of the input pattern by the linear links of an MLP, the functional link enhances the input pattern by using some nonlinear functions. The enhanced patterns are then applied to a single layer Perceptron. Since the hidden layer is absent the computational complexity of FLANNs are much lesser than that of multi layer ANNs. The BP algorithm used to train the FLANN becomes simpler and converges faster owing to its single layer structure.

The FLANN structure considered for the channel equalization problem is depicted in Fig. 2[9, 10, 11]. For a trigonometric FLANN the functional expansion block makes used of a functional model comprising of a subset of orthogonal ‘sin’ and ‘cos’

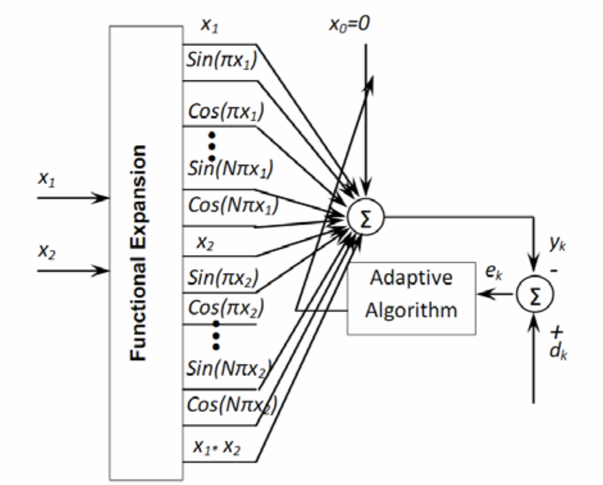


Fig. 2. Structure of a Trigonometric FLANN

basis functions and the original pattern along with its their products. For example if the two dimensional input pattern $X = [x_1, x_2]^T$ is applied then the functional expansion enhances the pattern as

$$[x_1, \cos(\pi x_1)\ \sin(\pi x_1)\ \ \cos(2\pi x_1)\ \sin(2\pi x_1)....$$
$$x_2, \cos(\pi x_2)\ \sin(\pi x_2)\ \ \cos(2\pi x_2)\ \sin(2\pi x_2)....]$$

The weighted sum of the enhanced input is passed through a hyperbolic tangent nonlinear function to produce y_k.

4 Evolutionary Approaches for Training of FLANN

4.1 Differential Evolution

Differential Evolution (DE) [12],[13],[14] is incontrovertibly one of the most powerful stochastic real-parameter optimization algorithms among those are in practice. DE operates through the similar computational steps as employed by a standard Evolutionary Algorithm (EA) though unlike the traditional EAs, the DE-variants perturb the current-generation population members with the scaled differences of randomly selected and distinct population individuals. DE is a simple real-coded evolutionary algorithm. DE works through a simple cycle of stages, first of all randomly population of *NP, D* dimensional real-valued parameter vectors are initiated. Each vector, also known as *genome/chromosome*, forms a candidate solution to the multi-dimensional optimization problem the 'i'-th vector of the population at the current generation can be represented by:

$$\vec{X}_{i,G} = [x_{1,i,G}, x_{2,i,G}, x_{3,i,G}, \ldots.., x_{D,i,G}], \tag{4}$$

where index 'G' represent the current generation. The initial population (at $G = 0$) should uniformly distributed over the entire search space within the prescribed minimum and maximum bounds. Then for each individual $x_{n,i,G}$ in the population a new offspring individual is generated by the following formula:

$$\vec{V}_{i,G} = \vec{X}_{r_1^i,G} + F \cdot (\vec{X}_{r_2^i,G} - \vec{X}_{r_3^i,G}), \tag{5}$$

where the indices $r_1^{\ i}$, $r_2^{\ i}$ and $r_3^{\ i} \in$ [1, NP] are three mutually exclusive integers randomly chosen. The difference of any two of these chosen three vectors is scaled by a scalar number F and the scaled difference is added to the third one whence we obtain the donor vector $\vec{V}_{i,G}$ .The scale factor $F > 0$ is a real constant and is often set to 0.5. Then for enhancing the potential diversity of the population, a crossover operation comes into play after generating the donor vector through mutation. *Trial* vector $\vec{U}_{i,G} = [u_{1,i,G}, u_{2,i,G}, u_{3,i,G}, \ldots, u_{D,i,G}]$ is created by exchanging the individuals of the donor vector with the trial vector.

$$u_{j,i,G} = \begin{cases} v_{j,i,G}, \ for\ j = \langle n \rangle_D, \langle n+1 \rangle_D, \ldots, \langle n+L-1 \rangle_D \\ x_{j,i,G}, \ for\ all\ other\ j \in [1, D], \end{cases} \tag{6}$$

where the angular brackets $\langle\ \rangle_D$ denote a modulo function with modulus D. The integer L is chosen from $[1, D]$. To keep the population size constant over subsequent generations, the next step of the algorithm calls for *selection* to determine whether the target or the trial vector survives to the next generation i.e. at $G = G + 1$. The selection operation is described as:

$$\vec{X}_{i,G+1} = \begin{cases} \vec{U}_{i,G}, \ if\ f(\vec{U}_{i,G}) \le f(\vec{X}_{i,G}) \\ \vec{X}_{i,G}, \ if\ f(\vec{U}_{i,G}) > f(\vec{X}_{i,G}), \end{cases} \tag{7}$$

where $f(\vec{X})$ is the objective function to be minimized. Here the fitness function used is mean-square error.

4.2 Genetic Algorithm

Genetic Algorithm (GA) is an optimization algorithm based on the mechanics of evolution and natural selection. Developed by Holland [15], GAs has been shown to outperform conventional non-linear optimization and local search techniques on 'difficult' search spaces (i.e. when the function is high-dimensional, multi-modal, discontinuous or noisy). GAs have been successfully implemented in channel equalization in [16], [17] for comparison purposes. The working principle of GA has been shown in Fig 3. The GA maintains a constant size population of candidate solutions. Each solution is represented by a fixed length string called a chromosome or genotype, which not only encodes its value (phenotype) but provides ‘genetic material' for the mutation and recombination operators. The individual components of the string are known as genes and each may take one of a small range of values.

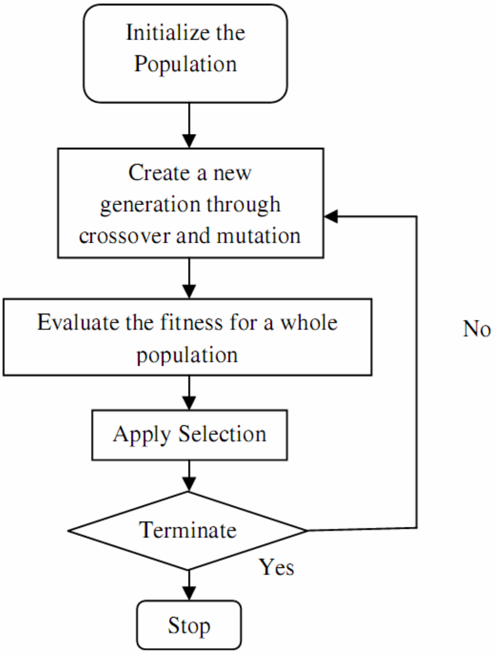

Initialize the Population
Create a new generation through crossover and mutation
Evaluate the fitness for a whole population
Apply Selection
Terminate
No
Yes
Stop

Fig. 3. Flowgraph showing working of Genetic Algorithms

During each iteration (generation) of the GA, the current population of solutions is evaluated and 'selected' to form the basis of the next population. The selection procedure operates to ensure that above-average solutions tend to be propagated to future generations whilst weaker solutions are replaced. Genetic algorithms are able to concentrate their efforts on globally better areas of the search space as a result of their ability to combine partial solutions, largely through the auspices of the 'crossover' operator. Crossover mimics the recombination of DNA that occurs when biological chromosomes line up and swap portions of their genetic information. In its simplest form, a single crossover point is chosen for two randomly selected parent structures (chromosomes) and the portions of genetic material on one side of this point exchanged. In this way, beneficial genes on two chromosomes may be combined in a single, fitter structure.

Like crossover, mutation operator is a very important operator in GA in which a chosen chromosome undergoes a small random change in its genetic material. But unlike crossover which is performed continually, mutation is rarely done. The mutation operator helps in overcoming the GA when trapped in local minima.

Since the weights that are updated in the current channel equalization problem are real numbered thus Real coded Genetic Algorithm (RCGA) is used in lieu of binary coded Genetic Algorithm (BCGA).

5 Simulation Studies

Here we discuss about the Mean Square Error (MSE) and BER performance of the equalizers used in the study. We have considered the following channels. The main reason for using the channel models (8) and nonlinear models (9) is that these models have been widely used by other researchers.

$$\begin{aligned} &CH=1 \quad H_1(z)=1 \\ &CH=2 \quad H_2(z)=0.3410+0.8760z^{-1}+0.3410z^{-2} \\ &CH=3 \quad H_3(z)=0.2014+0.9586z^{-1}+0.2014z^{-2} \\ &CH=4 \quad H_4(z)=0.2600+0.9300z^{-1}+0.2600z^{-2} \end{aligned} \tag{8}$$

The following nonlinearities are also considered.

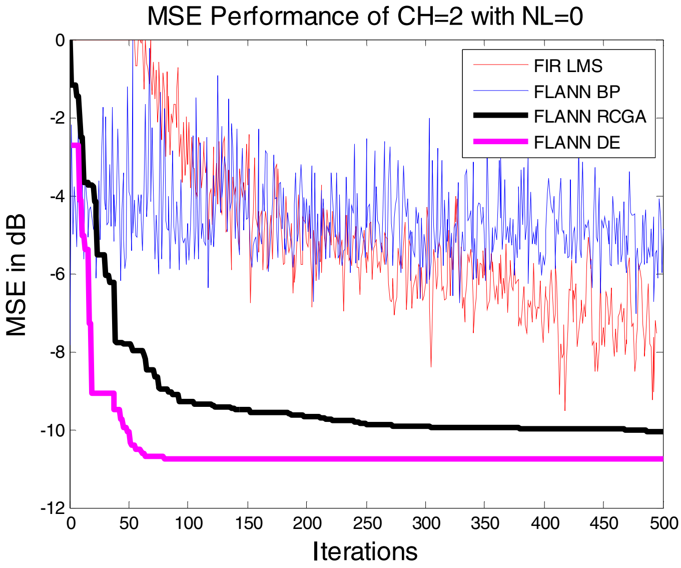
$$\begin{aligned} &NL=0 \quad b_k=a_k \\ &NL=1 \quad b_k=a_k+0.2a_k^2-0.1a_k^3 \\ &NL=2 \quad b_k=\tanh(a_k) \end{aligned} \tag{9}$$

The FLANN was chosen to be a 3 function sinusoidal expansion. The performance of DE based FLANN (FLANN -DE) was compared with an LMS trained FIR filter order N = 8 (FIR-LMS), an BP trained FLANN (FLANN-BP), a Real Coded GA trained FLANN (FLANN-RCGA). The number of agents was chosen to be 50 for both DE and GA. The probability of mutation and crossover for the GA was kept at 0.1 and 0.9 respectively. The values of f and CR for the FLANN –DE was kept at 0.6 and 0.9 respectively. The learning parameters of the FIR-LMS and FLANN-BP were kept at 0.1.

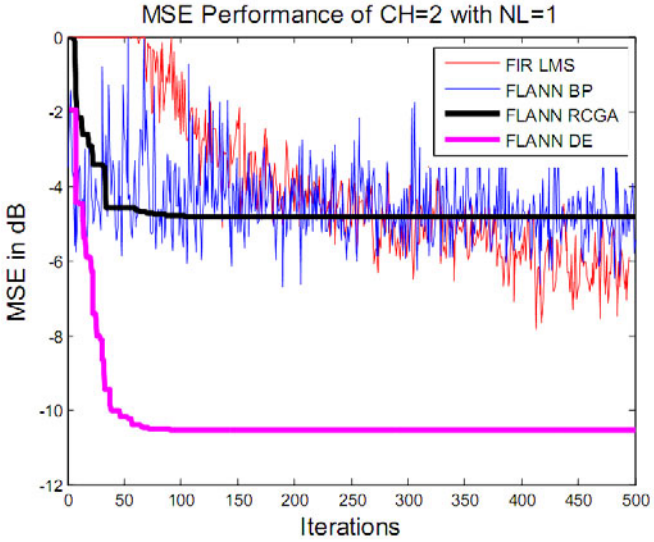
5.1 MSE Performance

To study the convergence characteristics and MSE performance of the equalizers, each equalizer was trained with 500 iterations for all channels at 20 and 30 dB additive noise conditions. To smooth out the randomness of the NN simulation, the MSE was averaged over 50 independent runs.

The MSE characteristics for CH = 2 with 20 dB additive noise for all types of nonlinearities is shown in Fig. 4.

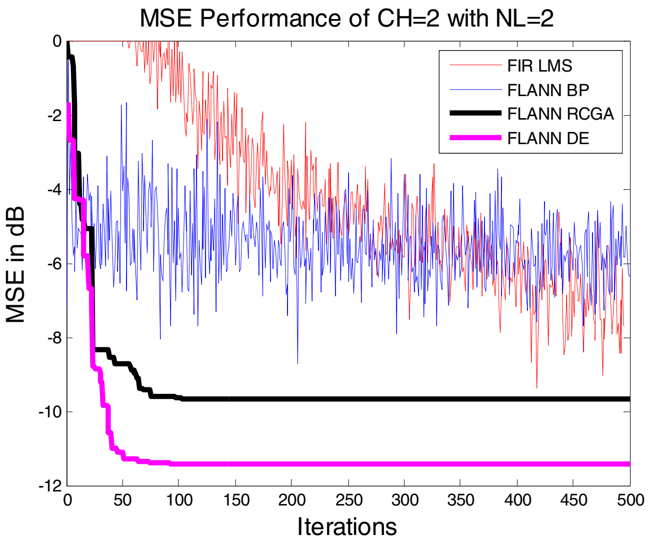


(a)



(b)

Fig. 4. MSE performance the FLANN-based equalizers for CH = 2: (a) NL = 0, (b) NL = 1, (c) NL = 2



(c)

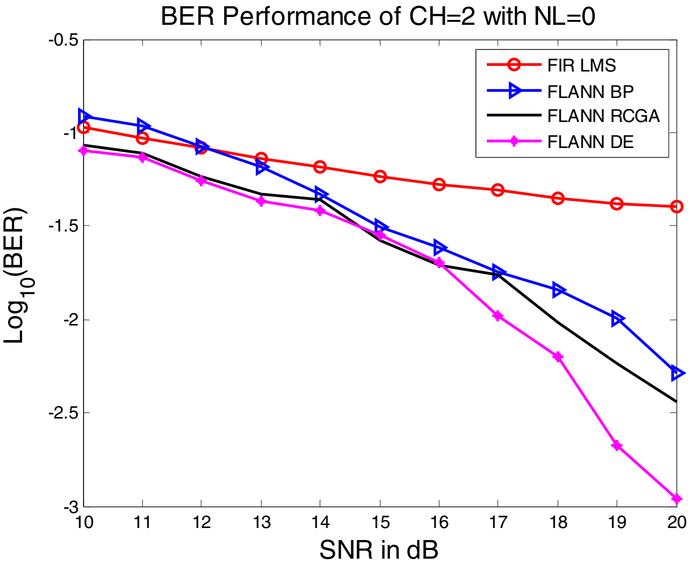
Fig. 4. (*continued*)

From the above study we observe that the MSE obtained by the FLANN trained with DE was the lowest among all the equalizers considered so far in the state of art for the linear and nonlinear channel application. The DE trained FLANN equalizer also showed fastest convergence than the rest of the equalizers.

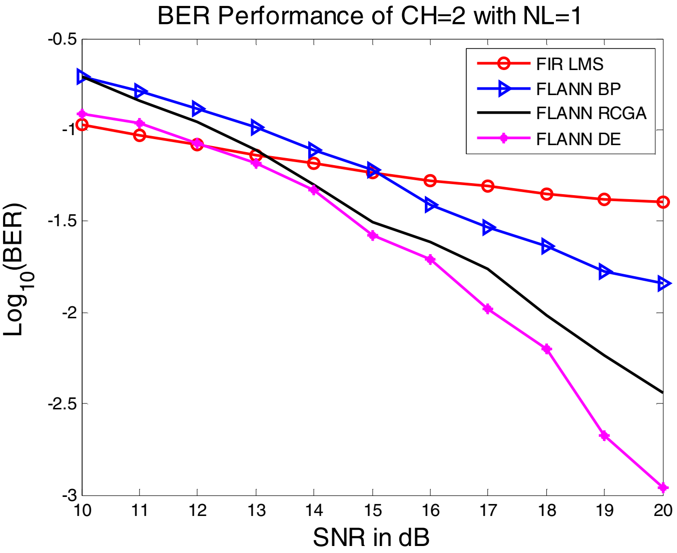
5.2 BER Performance

To study the BER performance of the equalizers, each equalizer was trained with 500 iterations for all the channels at 20 and 30 dB additive noise conditions. To smooth out the randomness of the neural network simulation, the BER was averaged over 50 independent runs.

The BER characteristics for CH = 2 with 20 dB additive noise for all types of nonlinearities is shown in Fig. 5.

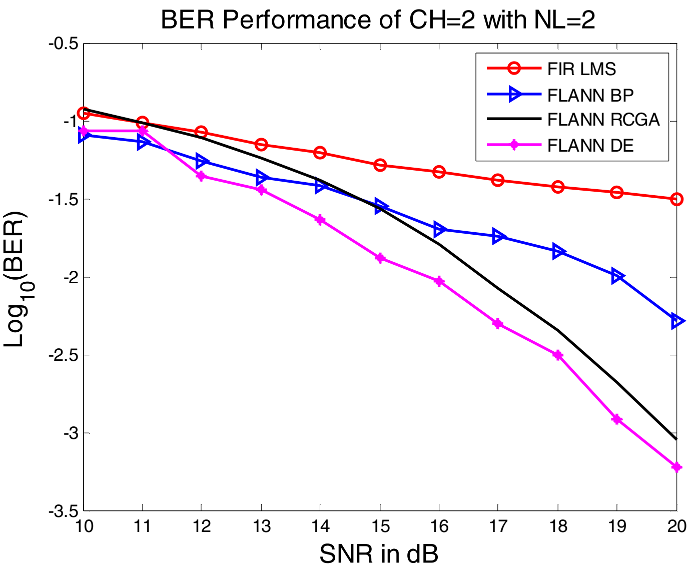


(a)



(b)

Fig. 5. BER performance the FLANN-based equalizers for CH = 2: (a) NL = 0, (b) NL = 1, (c) NL = 2



(c)

Fig. 5. (*continued*)

From the BER performance of CH=2 we observe that in all the three cases the performance of the FLANN trained with DE is better than the rest cases.

6 Conclusion

In this paper a novel method of training of the functional link artificial neural network by differential evolution is proposed. Comparisons with respect to mean square error and bit error rate have been performed with three other equalizers viz.; a FIR filter trained with LMS, a FLANN trained with BP as well as genetic algorithm. From the simulation study it is quite obvious that DE trained FLANN over performed the other equalizers. Thus it has been concluded that the DE algorithm converges faster than the BP and GA algorithm trained FLANN. Also it has been observed that the DE trained FLANN shows better MSE and BER performance than the other equalizers considered so far in the state of art in this domain of application.

The further extension of this paper can be implemented using DE to train the Chebyshev and Legendre expansion FLANNs.

References

1. Haykin, S.: Communication Systems, 4th edn. Wiley, New York (2001)
2. Haykin, S.: Adaptive Filter Theory, 3rd edn. Prentice-Hall, Upper Saddle River (1996)
3. Patra, J.C., Meher, P.K., Chakraborty, G.: Nonlinear Channel Equalization For Wireless Communication Systems Using Legendre Neural Networks. Journal, Signal Processing 89(11) (November 2009)

4. Chen, S., Gibson, G.J., Cowan, C.F.N.: Adaptive Channel Equalization Using Polynomial Perceptron Structure. Proc. IEE Part I 137, 257–264 (1990)
5. Gibson, G.J., Siu, S., Cowan, C.F.N.: The Application Of Nonlinear Structures To The Reconstruction Of Binary Signals. IEEE Trans. Signal Process. 39, 877–1884 (1991)
6. Meyer, M., Pfeiffer, G.: Multilayer Perceptron Based Decision Feedback Equalizers for Channels with Intersymbol Interference. Proc. IEE Part I 140, 420–424 (1993)
7. Dehuri, S., Cho, S.B.: A Comprehensive Survey on Functional Link Neural Networks and an Adaptive PSO–BP Learning for CFLNN. Neural Computing & Applications (June 14, 2009)
8. Xiang, Z., Bi, G., Ngoc, T.L.: Polynomial Perceptron and their Applications to Fading Channel Equalization and Co-Channel Interference Suppression. IEEE Trans. Signal Processing 42, 2470–2479 (1994)
9. Gan, W.S., Saraghan, J.J., Durrani, T.S.: New Functional-Link based Equalizer. Electronics Letters 58(17), 1643–1645 (1992)
10. Patra, J.C., Pal, R.N.: A Functional Link Artificial Neural Network for Adaptive Channel Equalization. Eurasip Signal Processing Journal 43(2) (1995)
11. Patra, J.C., Pal, R.N., Chatterjee, B.N., Panda, G.: Identification of Nonlinear Dynamic Systems Using Functional Link Artificial Neural Networks. IEEE Trans. Syst. Man Cybern. B, Cybern. 29(2), 254–262 (1999)
12. Storn, R., Price, K.V.: Differential evolution – A Simple and Efficient Heuristic for Global Optimization Over Continuous Spaces. Journal of Global Optimization 11(4), 341–359 (1997)
13. Price, K.V., Storn, R., Lampinen, J.: Differential Evolution - A Practical Approach to Global Optimization. Springer, Berlin (2005)
14. Das, S., Suganthan, P.N.: Differential Evolution – A Survey of the State-of-the-Art. IEEE Transactions on Evolutionary Computation (2010)
15. Holland, J.H.: Adaptation in Natural and Artificial Systems. University of Michigan Press, Ann Arbor (1975)
16. Yogi, S., Subhashini, K.R., Satapathy, J.K., Kumar, S.: Equalization of Digital Communication Channels Based on PSO Algorithm. In: International Conference on Communication Control and Computing Technologies (2010)
17. Jatoth, R.K., Vaddadi, M.S.B.S., Anoop, S.S.V.K.K.: An Intelligent Functional Link Artificial Neural Network for Channel Equalization. In: Proceedings of 8th WSEAS International Conference on Signal Processing, Robotics and Automation, ISPRA 2009, pp. 240–249 (2009)

Efficient VANET-Based Traffic Information Dissemination Using Centralized Fixed Infrastructure

Brij Bihari Dubey, Naveen Chauhan, Lalit Kumar Awasthi,
Narottam Chand, and Sudhanshu Pant

Department of Computer Science and Engineering
National Institute of Technology, Hamirpur – 177005 (H.P.)
{dubey.brijbihari,aveenchauhan.nith,
sudhanshupant1986}@gmail.com,{lalit,nar}@nitham.ac.in

Abstract. In the vehicular ad hoc networks several types of data are disseminated and transmission protocol changes with change in the type of data. Some information is useful for only those vehicles which are in certain specific range or location. Dissemination of traffic update message to all the vehicles is wastage of channel bandwidth. In our proposed method, vehicles which are already trapped in the traffic jam are transmitting traffic update message that is disseminated to the vehicles which might be trapped to the traffic jam. Simulation results demonstrate that the proposed protocol reduces congestion in the dense traffic region and efficiently utilizes the bandwidth in stressful road scenarios.

Keywords: Road Side Units, Packet Category, Intersection RSU, Central RSU, Exterior RSU.

1 Introduction

Vehicular ad hoc networks (VANETs) provide an exciting area of research at the intersection of a number of disciplines and technologies. There is a good scope for applications of VANETs, ranging from diagnostic, safety tools, information services, and traffic monitoring and business services. In recent years traffic jam has become the huge problem in the urban cities in many of the country. Due to traffic jam problem economic loss is more than 120 billion US dollars per year in Japan. Today the biggest challenge on the road is to reduce the traffic and accidents occurring on the road.

In vehicular networks basically there are two types of messages generated by the vehicles. The first type of message is the beacon message which stores status, speed, location and direction related information about any vehicle. This message is periodically updated and exchanged between vehicles in single hop only. Second type of message is generated when any event occurs and these messages may be related to traffic condition information, emergency warning information etc. In VANETs traffic status information is necessary to disseminate to those vehicles which are going to suffer congestion problem in few minutes. This type of information may tolerate

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 415–425, 2011.

slight delay like the parking place information. Chung *et al.* [1] have proposed a distributed mechanism which is known as parking space discovery (PSD) mechanism. Lochert *et al.* [2] proposed an algorithm for the hierarchical aggregation of observations in dissemination-based, distributed traffic information systems. Those vehicles which are already trapped in a traffic jams elect their leaders according to their locations from upstream and downstream respectively [3].

This paper proposes Nearest Intersection Location Dependent Dissemination of Traffic Information (NILDD) protocol which is significantly reducing channel congestion by forwarding traffic update messages to only those vehicles which are just unintentionally willing to trap in the traffic jam. Results show that redundant dissemination of packet containing traffic status information is minimized using NILDD protocols.

The rest of this paper is organized as follows: Section-2 describes the related works. Section-3 describes system architecture and design requirements in the vehicular ad hoc networks. Section-4 describes the proposed Nearest Intersection Location Dependent Dissemination (NILDD) protocol. Section-5 explains simulation results and finally, section-6 concludes this paper and explains the future works.

2 Related Works

VANETs, fixed communication range is not feasible to maintain network connectivity because vehicles are not homogeneously distributed over the network and since networks topology is changing very fast so due to high speed vehicular mobility, the link topology changes rapidly [4, 5, 6] several well defined approach for efficient data dissemination, such as trees, grids and clustering are not easy set up and maintain. The Dynamic Transmission-Range Assignment (DTRA) algorithm [7] is efficient to provide better connectivity in the dense, highly mobile, heterogeneous vehicular network where topology is rapidly changing. Vehicle Information and Communication System (VICS) [8] service is used to forward traffic information to other vehicles using Frequency Modulation (FM) broadcast messages and optical beacon messages. But the drawback of VICS system is that information provided by VICS is lagging behind the real time situation because traffic information is firstly collected at the centralized server and then forwarded. A protocol with high information arrival rate is proposed in [9] for inter-vehicle communication. This protocol manages allotted communication timing among vehicles depending on traffic flows. The dynamics of multihop emergency message dissemination in VANETs is discussed in [10].

3 System Architecture and Design Requirements

In this section, we are introducing few definitions and system architecture for our problem and identify design requirements.

3.1 Definitions

Packet Category (PC) - Packet category is the classification of the packet by which vehicles get to know that whether this packet should be transmitted to all the vehicles or to some group of vehicles or to any particular vehicles. This PC value is associated with every data packet in addition to existing header information. There are typically three major category of information that RSU or any vehicle may have to transmit.

Forwarding to all Vehicles (e.g.- commercial application, any broadcast information).

Forwarding to specific group of Vehicles (e.g. - emergency message, traffic jam update message).

Forwarding to any particular Vehicles only

Traffic Update Message (TUM) – Traffic update message contains the data packet which has to be transmitted to the specific group of vehicles. Message can be signified as TUM message depending PC value associated with it. These messages are associated with the packet category (PC) value and time to live (TTL) value. PC value will denote that this packet is TUM message and TTL value will signify the time period for which TUM message supposed to be valid. If TTL value of any packet is expired then TUM message is no more valid.

Intersection Road Side Units (I-RSU) – Road side units existing at the intersection point of more than one road is called the intersection road side units (I-RSU).

Central Road Side Units (C-RSU) – I-RSU which is nearest to the location of the traffic jam and is present on the same road is the central road side unit (C-RSU). Any I-RSU may act as the C-RSU depending upon the position of the traffic jam location.

Exterior Road Side Units (E-RSU) – Those I-RSU's which are nearest to the C-RSU on all the roads passing from C-RSU, are treated as E-RSU. So E-RSU is actually such the intersection point which receives traffic update information from vehicles and forward to the vehicles which needs this information in next few minutes so that can select alternate paths and avoid the congestion. One E-RSU may act as C-RSU depending upon the congested region.

3.2 System Architecture

Here general double lane road layout has been chosen in which highly mobile vehicles are moving on roads and these roads are either straight or intersecting each other at intersection point (as shown in figure-1). When vehicles are receiving any traffic update message then they may choose any alternate path which suites to its destination. Vehicles trapped in jams will forward messages to E-RSU and E-RSU will store the entire packet details until TTL of packet expires and forward them to the vehicles which will pass through the communication range of this E-RSU in a single hop or limited multihop.

4 Nearest Intersection Location Dependent Dissemination Protocol

Whenever vehicles are ready with the packets to transmit it to the destination, the packet must be associated with PC value. PC value exhibits that the type of the data packet and it realizes that the information stored in this packet is pertaining to emergency warning message, traffic information, commercial broadcast or any other type of message. Data packet will select the destination depending upon the PC value associated with the data packet. If the packet has the PC value equals to TUM then this is supposed to be a traffic information packet and this packet has to be transmitted to only those vehicles which are unintentionally willing to increase the traffic in such region which is already a dense with vehicles. Whenever traffic position is dense in any region, this information must be dealt with the C-RSU and this C-RSU will select all other corresponding E-RSU and forward this information to all the E-RSU. So that E-RSU can deliver this TUM message to those vehicles which are unintentionally just going to suffer the traffic jam problem.

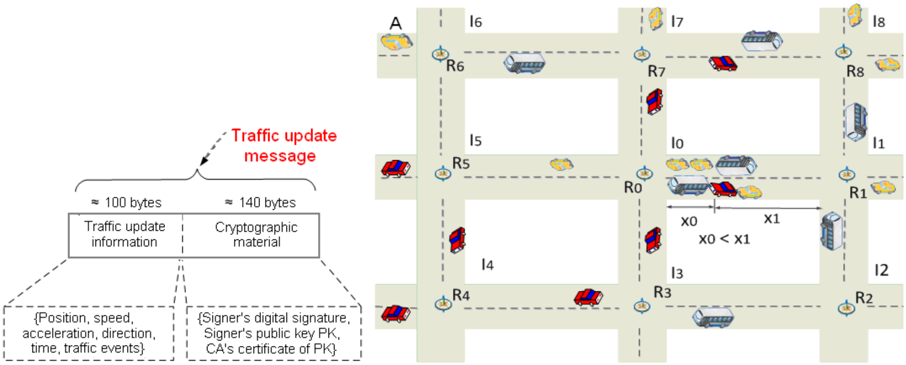

Traffic update message
≈ 100 bytes
≈ 140 bytes
Traffic update information
Cryptographic material
{Position, speed, acceleration, direction, time, traffic events}
{Signer's digital signature, Signer's public key PK, CA's certificate of PK}
A
I6
I7
I8
R6
R7
R8
I5
I0
I1
R5
R0
R1
x0
x1
x0 < x1
I4
I3
I2
R4
R3
R2

Fig. 1. Selection of E-RSU

When packet containing traffic status information (TUM message) will be delivered at the E-RSU then E-RSU will forward this packet to the new vehicles arriving from directions other than direction in which C-RSU is situated near this E-RSU then E-RSU will transmit this TUM message to all those vehicles in a single hop or in limited multihop and E-RSU will suggest some alternate paths to those vehicles to reach their destination. By this way only those vehicles will receive the traffic information data packets which are really going to suffer with traffic jam and deteriorate the traffic condition. If the nearest RSU is not situated at the intersection of the roads and it is at the straight way road then this TUM message will be further forwarded on the same track to find the nearest intersection road side unit (I-RSU).

For forwarding TUM message from congested region to the vehicles which are just unintentionally willing enter in congested region two mechanisms are used. Firstly TUM message will be transmitted by vehicles which are suffering traffic jam using carry and forward mechanism [11] to the C-RSU and C-RSU will check whether it is capable of transmitting this TUM message to the E- RSUs in a single hop. If yes it will transfer and if not then TUM message will be transferred to E-RSUs using carry and forward mechanism again (as in figure 2).

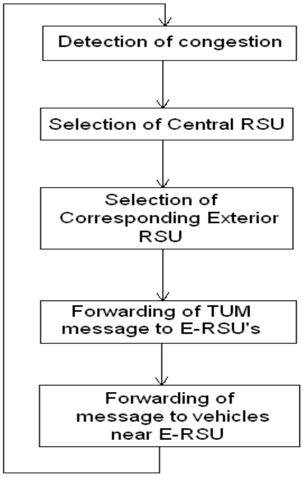

Detection of congestion
Selection of Central RSU
Selection of Corresponding Exterior RSU
Forwarding of TUM message to E-RSU's
Forwarding of message to vehicles near E-RSU

Fig. 2. Transmission Flow Chart of Message

Algorithm 1. NILDD Protocol

```
Initialize PC_value, TTL_value
for any Message do
  if Message is not in the Queue then
     Add PC_value to packet_header
     Add TTL_value to packet_header
     if PC_value  = TUM then
        do TUM_message ← packet_header+Message
        do search_nearest_RSU()
        nearest_RSU ← search_nearest_RSU
        if nearest_RSU = I-RSU then
           C-RSU  ←  I-RSU
           for any E-RSU  do //E-RSU's of this C-RSU
              Forward TUM_message to E-RSU
              if TTL_value = valid  then
                 do Forward TUM_message to vehicles
            end if
            end for
         else
```

```
19:             do search_nearest_RSU()
20:             go to line-10
21:           end if
22:         end if
23:       end if
24:     end for
```

Let any vehicle is moving in any direction then it may receive any traffic update message which is occurring at some other location. So it is not necessary that all the information is useful for every vehicle moving on the road. So it finally depends on the current location of the vehicle and the place where the traffic status information is updated. If vehicle A is moving near the intersection point I_6 and if vehicle is moving towards the north direction then so there is sufficient probability that vehicle A will move on the roads away from the location where intersection point I_0 is situated (As shown in figure - 1). So it is useless to receive traffic update message of any other place near the intersection point I_0 until this information is specially requested by the vehicle A. Because the probability that vehicle A will come near the intersection point I_0 is quite low and forwarding of this traffic information message to vehicle A is just the wastage of bandwidth and poor utilization of services and resources. When this vehicle is near the intersection point I_1, I_3, I_5, I_7 then this may be useful because the probability that this vehicle will move towards the C-RSU R_0 (intersection point I_0) is very high and may be the one of the vehicles which is creating congestion. So when vehicle A is near to any of the intersection point I_1, I_3, I_5, I_7 (having RSU R_1, R_3, R_5, R_7 respectively which are actually E-RSUs corresponding the C-RSU R_0) and vehicle A is receiving traffic status information (occurring near to R_0), then vehicle A may choose alternate paths suggested by RSU and can select other path to their destination. In this way packet forwarding is limited to those vehicles which may enter in congested area.

4.1 Role of Fixed Infrastructure's Position

The packet transferred from the RSU should be delivered to the vehicle moving on the road. But this data packet should be given to the vehicles between the times t_a and t_b (where t_a is the time when vehicle enters in the range of the RSU at the intersection and t_b is the time when vehicle is just crossing the intersection). This time period will be different for different placement of the RSU's. If the RSU is placed at the corner of the intersection then scenario is different as the RSU is placed at the center of the intersection point. At the intersection the radio transmission coverage area of the RSU is different for the RSU's present at the different locations. The coverage area at the road A1 and A2 is more the A3 and A4. If the RSU is at the corner then coverage area is reduced by nearly 15% of as the RSU's presence at the center of the intersection. So, by this way of transmission time of data packet from RSU to the vehicles is different for the different location of the RSU placement. Since the packet can be forwarded only in the period in which the vehicle is in the coverage range so the maximum transmission time allotted for dissemination is the time period in which vehicle enters in the coverage area and evacuates the coverage area and this time interval message should be transmitted to the vehicles. Vehicle will get maximum to transmit or receive data, when they are moving through the nearest to the center of the coverage area and traveling distance closer to 2*a* (approximately).

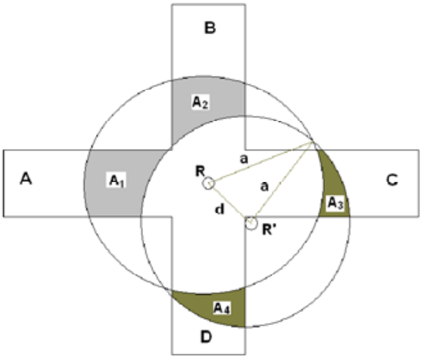


Fig. 3. Coverage Area of the Road Side Units

A1 and A2 is the increase in the coverage area of the RSU at road A and B respectively, when RSU is shifted from corner of the intersection point to the center of the intersection point. So, A_1 can be defined as

$$A_1 = A_2 = \int_{-d/2}^{d/2} \int_{\sqrt{a^2-(x-d/\sqrt{2})^2}-d/\sqrt{2}}^{\sqrt{a^2-x^2}} dxdy \tag{1}$$

Similarly, A3 and A4 is the reduction in the coverage area of the RSU at the road B and C respectively due to placement of RSU from the corner of intersection point to the center of the intersection point. So, A_3 can be defined as

$$A_3 = A_4 = \int_{d/2}^{-d/2} \int_{\sqrt{a^2-y^2}}^{\sqrt{a^2-(y+d/\sqrt{2})^2}+d/\sqrt{2}} dydx \tag{2}$$

Where, *a* is the radius of the circular range of the RSU, *d* is the width of the road AC and BD and RSU is shifted from corner position R' to center of intersection point R.

So, increase in coverage area at road A =
Increase in coverage area at road B =

$$\Delta A = \left[\int_{-d/2}^{d/2} \int_{\sqrt{a^2-(x-d/\sqrt{2}\,)^2}-d/\sqrt{2}}^{\sqrt{a^2-x^2}} dxdy \right] - \left[\int_{d/2}^{-d/2} \int_{\sqrt{a^2-y^2}}^{\sqrt{a^2-(y+d/\sqrt{2}\,)^2}+d/\sqrt{2}} dydx \right]$$

So, total increase in the coverage area is (approximately) 15%. This ΔA is the increment in the coverage area at the intersection of two or more roads.

4.2 Selection Strategy Based on Relevance

The selection of location of placement of fixed units is dependent on the 1) Coverage area of the fixed unit, 2) Vehicular traffic density across the particular road, 3) velocity of the vehicles moving across the specific road. In the given figure-3 if the traffic density on road C and/or D is more than traffic density on road A and/or B. Similarly, if the vehicles speed at road C and/or D is more than the vehicle speed at the road A and/or B. So the fixed unit should be shifted in the direction of vehicles having high speed mobility and high density. So there is a need of finding the optimal location for the fixed units.

The movement of the vehicles on the road is basically of the two types. 1) In one scenario vehicles moving on the road are in the form of the streams. And the in the network there are sufficient number of vehicles to maintain the communication in network i.e. every vehicle on the road is in the radio transmission range of some other vehicles. 2) But typical urban road scenario is quite different and has distributed traffic which has a group of vehicles moving in the radio range of another. The data dissemination is faster in first case due to establishment of connectivity but in the second case the data dissemination is done by the carry and forward mechanism.

5 Simulation Results

In this we evaluate the performance of the NILDD protocol. We use the simulation environment provided by ns2, which is typically used to simulate VANETs. The following are the parameters used during the simulation:

If R_a is the number of roads (path) available as alternate paths at the intersection and R_p is the total number of roads (path) available as all possible paths at the intersection

$$R_a = R_p - 1 \tag{3}$$

Table 1. Parameters used in simulation

Simulation Area	5000m X 3000m
Simulation Time	1200 seconds
No of vehicles	1000
Vehicle Velocity	40km/hr - 60 km/hr
Number of Intersections	5 – 50
Constant Bit Rate (CBR rate)	0.1 – 1 packet per second
Communication Range	100m
Bandwidth	20Mbps
Data packet size	1 KB
I-RSU cache replacement policy	FIFO

Since number of available paths available at any intersection is equal to or more than 3. So total alternate paths available at any intersection is given by

$$R_a \geq 2 \tag{4}$$

n_k = Total number of vehicles coming from k^{th} alternate path and
N_a = Total number of vehicles coming from all the alternate paths

Then
$$N_a = \sum_{k=1}^{\mathrm{R_a}} n_k \tag{5}$$

When TUM message forwarding is started by the E-RSU then all the vehicles reaching towards the intersection $\mathrm{I_1}$ will receive this message and then probability that these vehicles will go towards intersection $\mathrm{I_0}$ is P (as shown in figure 2). Then

$$P = 1 - \left[R_a / R_p\right]^{N_a} \tag{6}$$

Let total number of vehicles entering from intersection I_1 towards intersection I_0 in unit time is N_i and total number of vehicles exiting from intersection I_0 in unit time which has entered from intersection I_1 is N_o.

Then total number of vehicles moving form intersection I_1 towards I_0 in unit time is given by

$$N_i = P * N_a \tag{7}$$

Then congestion factor is given by

$$Congestion\ \ Factor\ (CF) = \frac{N_i - N_o}{N_i} \tag{8}$$

Congestion factor will always less than or equal to one.

$$Congestion\, Factor \leq CF_{\max} = 1$$

If Congestion factor is zero then there is no congestion. When the Congestion Factor will cross the threshold value (CF_{th}), the E-RSU will start transmitting TUM message to those vehicles which are going to enter in the congested region in a single hop or limited multi hop. Congestion Factor may increase maximum up to CF_{max}. After reaching up to this value this congestion factor will again drop and message forwarding will be stopped when congestion factor will reduced to CF_{th}. Transmission of TUM message is started by E-RSU to the vehicles at time T_{start} (when congestion factor crosses the threshold value) and transmission is stopped at time T_{stop} (when congestion factor falls below the threshold value). Transmission of TUM messages is again started when congestion factor again reaches to threshold value. Congestion Factor can measure the traffic load on the particular road. At T_{avg} the value of Congestion factor much less than CF_{max} (as shown in figure-4). Theoretically, congestion factor should not cross CF_{th} value between T_{start} and T_{stop}. Congestion factor is crossing the CF_{th} value but it is almost near the CF_{th} value. This is the success of the NILDD protocol. Congestion has been significantly controlled by transmitting TUM message to only those vehicles which are moving in limited area. Bandwidth is efficiently utilized by using NILDD protocol.

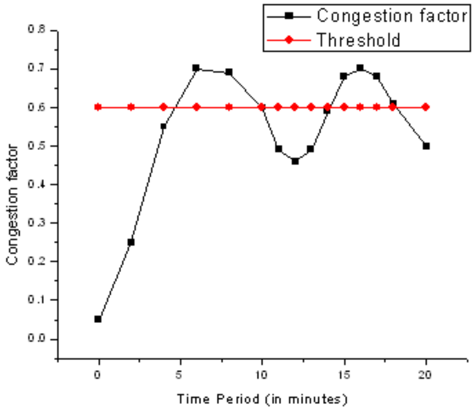


Fig. 4. Congestion Factor Vs Time

6 Conclusions

In this paper, we proposed a traffic jam information dissemination scheme. Our proposed scheme elects central road side units and corresponding exterior road side units to forward information to the vehicles. This scheme forwards traffic update messages to limited vehicles and reduces unnecessary utilization of bandwidth and other resources. In future we can device some mechanism to reduce overhead of the RSUs without reducing the performance of the system.

References

1. Chung, C.Y.: Parking Space Discovery System in Vehicular Ad hoc Networks. In: Master Thesis of National Central University, Taiwan (October 2008)
2. Lochert, C., Scheuermann, B., Mauve, M.: Probabilistic Aggregation for Data Dissemination in VANETs. In: 4th ACM International Workshop on Vehicular Ad Hoc Networks (VANET 2007), Montreal, Quebec, Canada, pp. 1–8 (September 2007)

3. Chang, H., Kwak, H.J., Park, G.T.: 2009.: Efficient Dissemination Method for Traffic Jams Information Sharing Based on Inter-Vehicle Communication. In: IEEE Student Conference on Research and Development, UPM Serdang, Malaysia, pp. 16–18 (November 2009)
4. Namboodiri, V., Agarwal, M., Gao, L.: A study on the feasibility of mobile gateways for vehicular ad-hoc networks. In: CM VANET, pp. 66–75 (October 2004)
5. Naumov, V., Baumann, R., Gross, T.: An Evaluation of Inter-Vehicle Ad Hoc Networks Based on Realistic Vehicular Traces. In: ACM MobiHoc, pp. 108–119 (2006)
6. Wu, H., Fujimoto, R., Guensler, R., Hunter, M.: MDDV: A Mobility Centric Data Dissemination Algorithm for Vehicular Networks. In: 1st ACM VANET, pp. 47–56 (2004)
7. Artimy, M.: Local Density Estimation and Dynamic Transmission-Range Assignment in Vehicular Ad Hoc Networks. IEEE Transactions on Intelligent Transportation Systems (September 2007)
8. Ministry of Land Infrastructure and Transport: The system outline of VICS, `http://www.its.gojp/etcvics/vics/`
9. Saito, M., Tsukamoto, J., Umedu, T., Higashino, T.: Evaluation of Inter-Vehicle Ad-hoc Communication Protocol. In: 19th International Conference on Advanced Information Networking and Applications, pp. 78–83 (2005)
10. Resta, G., Santi, P., Simon, J.: Analysis of Multi-Hop Emergency Message Propagation in Vehicular Ad Hoc Networks. In: 8th ACM International Symposium on mobile Ad Hoc Networking and Computing, pp. 140–149 (2007)
11. Davis, D., Fagg, A., Levine, B.: Wearable Computers as Packet Transport Mechanisms in Highly-Partitioned Ad-Hoc Networks. In: International Symposium on Wearable Computing, p. 141 (2001)

An Analytical Model for QoS Routing in TDMA-Based Ad Hoc Networks

Khaled Abdullah Mohammed Al Soufy and Ash Mohammad Abbas

Department of Computer Engineering,
Zakir Husain College of Engineering and Technology
Aligarh Muslim University
Aligarh - 202002, India
kalsoufi@gmail.com, am.abbas.ce@amu.ac.in

Abstract. Providing quality of service (QoS) in a mobile ad hoc network is a challenging task due to its peculiar characteristics. This paper aims at presenting a routing protocol which identifies a path between a given source and a destination in TDMA-based ad hoc networks. This path is examined for satisfying QoS in terms of end-to-end delay. For that purpose, we introduce an analytical model for end-to-end delays incurred by a packet from a given source to a destination. We have evaluated the performance of our protocol through simulation. The proposed protocol performs better in term of QoS satisfaction ratio as compared to existing protocols.

Keywords: Quality of service; end-to-end delay; queuing delay; TDMA; contention area.

1 Introduction

Recently, an interest has been created in investigating the use of TDMA as an alternative to CSMA/CA at MAC layer of an ad hoc network. A major advantage of TDMA over CSMA/CA is energy efficiency [1]. In case of TDMA the transmissions of neighboring nodes can be scheduled at different time slots [2] so that their transmissions do not interfere with one another. This is in contrast to CSMA/CA where the transmissions of neighboring nodes will collide if both of them try to transmit at the same time. In a TDMA-based ad hoc network, nodes may reserve time slots before transmitting their packets. In case of CSMA/CA, the nodes may reserve the slots for transmissions using very small control packets called RTS/CTS, however, the overheads can be fairly high.

On the other hand, in case of CSMA/CA-based MAC protocols, the packets transmitted by a node may collide with the packets transmitted by the neighboring nodes. This is in contrast with TDMA where there are pre-allocated time slots that are reserved for transmitting the packets [3]. Therefore, TDMA has a potential to provide QoS guarantees.

Some authors have investigated how one can provide QoS in an ad hoc network which is based on TDMA. To that end, a QoS routing protocol based on TDMA

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 426–437, 2011.

is proposed in [4]. We call it Zhu and Corson's Quality of Service (ZCQ) routing protocol. For providing a QoS path and to reserve the required number of time slots, the protocol, therein, utilizes one-hop neighborhood information.

In this paper, we focus on the end-to-end delay guarantee because it is one the significant requirements for the real-time applications. We propose a protocol that identifies a path between a given source and a destination that is able to satisfy the QoS requirements in terms of the end-to-end delays. We evaluate the performance of the protocol using simulations. We observe that the proposed protocol performs better in terms of the QoS satisfaction ratio as compared to the existing protocols.

The rest of this paper is organized as follows. In Section 2, we describe the proposed protocol. In Section 3, we analyze the end-to-end delays using multiple vacation model. Section 4 contains results and discussion. Finally, the last section is the conclusion. In what follows, we describe the proposed protocol.

2 Proposed Protocol

Our protocol consists of three phases: (i) slot allocation, (ii) route discovery, and (iii) route maintenance. We describe each of these phases as follows.

2.1 Slot Allocation

A TDMA frame consists of a number of slots and a slot is divided into data and control subframes. Nodes of the network compete to reserve the time slots for their transmissions. In case of TDMA, nodes are allowed to transmit/receive in the slots that are drawn from a pool of slots with the condition that the slot selected should not be used by any of their neighbors that are located within their contention area (CA). Of course, contention area can be selected in such a fashion so as to avoid hidden and exposed terminal problems. We assume that the area covered by a circle around a node with the radius that coverage twice of the transmission range of a node as the contention area. The basis of this assumptions is that signal power beyond twice of the transmission range of a node is weak to be decoded as the signal. Note in case of CSMA/CA, and therein, it is called the carrier sense range (see Figure 1). Every node in the network tries to determine a set of time slots that it may use for the transmission of its packets by using a mechanism presented in our previous work [5].

2.2 Route Discovery

In the route discovery phase, a source node that wants to communicate to a destination node generates a *Route Request (RREQ)* provided that it does not have a valid route to the destination. An *RREQ* contains the following information. <*SourceAddress, DestinationAddress, SequenceNumber, TraversedHopList, QoSSlots, AvailableSlots, DelayLimit, PathDelay, ContendingNeighbours*>. Note that the tuple <*SourceAddress, DestinationAddress, SequenceNumber*> uniquely

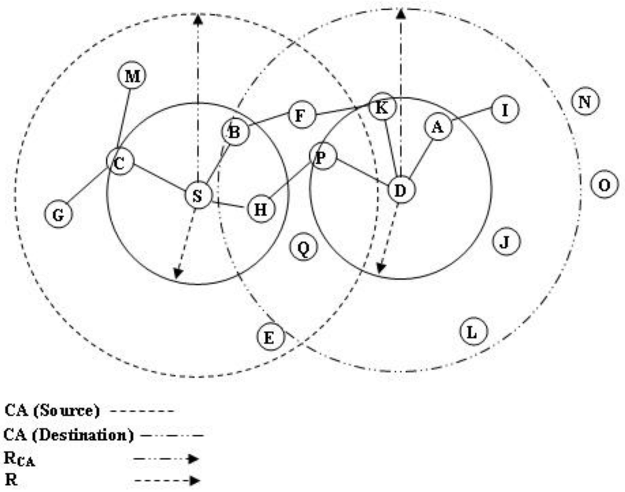


Fig. 1. The contention area of a given source node, S, and the destination node, D

identifies an *RREQ*. Two *RREQs* with the same values for the fields of the tuple are known as copies of one another. Let us call a node, which is neither the source of the *RREQ* nor is the destination of the *RREQ*, as an intermediate node. *ContendingNeighbours* and *DelayLimit* are initialized with threshold values and *QoSSlots* is setup by requested slots to send the date packets from a given source to its destination. When an intermediate node receives a *RREQ*, it examines whether its own address is already present in the *TraversedHopList* field. If so, it discards *RREQ*. Otherwise, it decides to forward the copy of the *RREQ*. Before forwarding the copy of the *RREQ* to the next neighbor, an intermediate node follows the following procedure.

- It estimates the delay from its upstream neighbors to itself including the queuing delays. Let the estimated delay be *HopDealy.* It then adds the value contained in *RREQ.PathDelay* to the *HopDelay.* In other words,

$$RREQ.PathDelay = RREQ : PathDelay + HopDelay.$$

- It compares *RREQ.PathDelay* with *DelayLimit.* If *RREQ.PathDelay* > *DelayLimit*, then it discards the *RREQ*.
- It determines the number of its neighbors (say *NumNeighbor*), which are within its CA, using a beacon mechanism. If NumNeighbor > ContendingNeighbours, then it discards the *RREQ*.
- It checks whether the number of available slots at the current node (say *AvailableSlots*[i]) is less than the *QoSSlots.* If yes, it discards *RREQ*. Otherwise, it forwards *RREQ* to its neighboring nodes.

It is clear that when the *RREQ* reaches the destination, the field *PathDelay* contains the delay required for forwarding a data packet from the source to the destination along the path represented by the *TraversedHopList* of the *RREQ*. Thus the destination checks whether the *RREQ.PathDelay* $\leq$ *DelayLimit*. If yes, it generates a *Route Reply (RREP)*. Otherwise the destination discards the *RREQ*. Note that the traversed path satisfies the bandwidth requirements in terms of the number of required slots. The *RREP* is sent towards the source along the reverse path brought by the copy as *TraversedHopList*. When the first *RREP* reaches at the source, the source starts sending the data packets.

2.3 Route Maintenance

If a node senses a link failure, it informs upstream nodes along all those paths whose part was the failed link. To do so, it unicasts *Route Error (RERR)* messages, one for each failed path. Every node that receives an *RERR* message marks the path invalid and unicasts the *RERR* upstream. Eventually, the *RERR* arrives at the source. When the source receives an *RERR* message it marks the failed path invalid. The source then looks for an alternate path in the route cache. If it finds a path to the destination that is not yet failed and that satisfies the QoS requirements of the flow of packets, it starts sending data packets along the path. Otherwise, it initiates a new route discovery.

In what follows, we present a model for end-to-end delays for a TDMA-based ad hoc networks.

3 End-to-End Delay

In this section, we analyze the average end-to-end delays incurred by a packet from a given source to a destination. For that purpose, we first analyze the delay incurred along a single hop and then we extend our analysis for a path that may consist multiple hops. Note that single hop delays consists of queuing delays, channel access delays, transmissions delays, and propagation delays. Packet transmissions delay is $\frac{L}{C}$, where L is the packet length including the header and C is the capacity of the link. The propagation delay depends upon the distance between transmitting node and the receiving node and on the propagation speed of the medium. In case of a wireless networks, the medium is air, and, the propagation speed is the speed of light, therefore, propagation delays are negligible. The remaining delays are the queuing delays and the channel access delays. We first consider queuing delays and then we consider the channel access delays.

3.1 Queuing Delay

In this subsection, we analyze model the queuing delays in a TDMA based ad hoc network using M/G/1/K queuing system with vacation. Since TDMA system, a node transmits in a set of time slots which is selected for it and then waits for its turn. It is analogous to a server which serves for a specific period called its service time and then goes for a vacation. In the duration of vacation, the server is waiting for its turn to transmit the next packet. Therefore, a queuing model

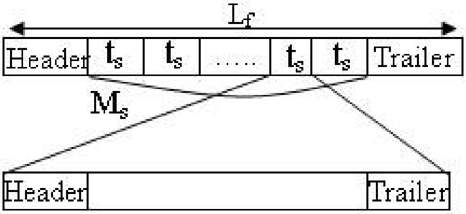


Fig. 2. Format of TDMA frame

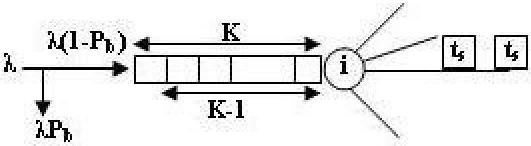


Fig. 3. M/G/1/K Queuing model

such as M/G/1/K with multiple vacations sames to be an appropriate queuing model to obtain the waiting time in a queue of a TDMA based ad hoc network.

Let the arrival rate of the frame be λ, and service rate be μ. Let the service times be independent and identically distributed. The packets that arrive at a node i, wait for the transmission of all packets in the queue including the packet that is being served. We assume that each node has a limited number of buffers in its queue, which is K. We assume that the service discipline is non-preemptive and the scheduling policy is FIFO. Let $\Pi_{v,k}$ be the probability that the number of packets in the queue of a node during a vacation is k, where $k = 0...K$, and $\Pi_{s,k}$ be the probability that the number of packets in the queue of a node during the service time is k, where $k = 0...K-1$. Let $\overline{S}$ be the mean service time and $\overline{V}$ be the mean vacation time. Let $\gamma_{v,k}$ be the probability that there are k packets in the queue just at vacation completion, and $\gamma_{s,k}$ be the probability that there are k packets in the queue just at service completion. Let ρ_c be the probability that the server is busy (i.e., it is not in the vacation period.). To calculate the queuing delay at the node i, we need to compute the average number of packets, N_q, in the queue during the service time and the vacation time. For that purpose, we consider the duration of service completion and the duration of vacation completion.

The mean time duration, T, between two consecutive imbedded points at the equilibrium state is as follows.

$$T = (1 - (\gamma_{v,0} + \gamma_{s,0}))\overline{S} + (\gamma_{v,0} + \gamma_{s,0})\overline{V} \tag{1}$$

where $(\gamma_{v,0} + \gamma_{s,0})$ is the probability of the vacation completion and that the queue is non-empty; and $(1-(\gamma_{v,0}+\gamma_{s,0}))$ is the probability of the service period completion and the queue becomes empty at this moment.

Let us now consider the probability that the server is busy at an arbitrary period of time. We assume that there are two imbedded points, the first point denotes the beginning of the service time and the second point denotes the the completion of the vacation time. Let α_{bs} be the embedded point that denotes the beginning of the service time, which is as follows.

$$\alpha_{bs} = \frac{(1-(\gamma_{v,0}+\gamma_{s,0}))(1-\rho_c)}{\overline{V}} \tag{2}$$

Let α_{bc} be the embedded point that denotes the completion of the service time, which is as follows,

$$\alpha_{bc} = \frac{\rho_c(\gamma_{v,0}+\gamma_{s,0})}{\overline{S}} \tag{3}$$

Note that at the time instance when the vacations time completes, the service time begins, therefore, $\alpha_{bs} = \alpha_{bc}$. Using (2) and (3), we have,

$$\frac{(1-(\gamma_{v,0}+\gamma_{s,0}))(1-\rho_c)}{\overline{V}} = \frac{\rho_c(\gamma_{v,0}+\gamma_{s,0})}{\overline{S}} \tag{4}$$

Simplifying (4), we have,

$$\begin{aligned} &\rho_c\left[(1-(\gamma_{v,0}+\gamma_{s,0}))\overline{S} + (\gamma_{v,0}+\gamma_{s,0})\overline{V})\right] \\ &= (1-(\gamma_{v,0}+\gamma_{s,0}))\overline{S}. \end{aligned} \tag{5}$$

Hence, the probability that the server is busy at an arbitrary period of time (i.e. the server is not on a vacation) is as follows,

$$\rho_c = \frac{(1-(\gamma_{v,0}+\gamma_{s,0}))\overline{S}}{(1-(\gamma_{v,0}+\gamma_{s,0}))\overline{S} + (\gamma_{v,0}+\gamma_{s,0})\overline{V}}. \tag{6}$$

The probability that there are k packets in the queue during the vacation period is as follows.

$$\varPi_{v,k} = \begin{cases} \dfrac{1}{\lambda T}\displaystyle\sum_{j=k+1}^{K} \gamma_{v,j} & k = 0, ...(K-1) \\ 1-\rho_c - \dfrac{1}{\lambda T}\displaystyle\sum_{j=1}^{K} j\gamma_{v,j} & k = K. \end{cases} \tag{7}$$

Similarly, the probability that there are k packets in the queue during the service period, is as follows.

$$\varPi_{s,k} = \begin{cases} \dfrac{1}{\lambda T}\left(\gamma_{s,k} - \displaystyle\sum_{j=k+1}^{K} \gamma_{v,j}\right) & k = 0, ...(K-1) \\ \dfrac{\rho_c(\rho-1)}{\rho} + \dfrac{1}{\lambda T}\displaystyle\sum_{j=1}^{K} j\gamma_j & k = K \end{cases} \tag{8}$$

In the equilibrium, one can find the probability distribution $\{\Omega_k, 0 \leq k \leq K\}$, where Ω_k is the steady state probability that there are k packets in the queue at an arbitrary period of time. Note that the events that the packet arrives at the queue during the service period or it arrives during the vacation period are mutually exclusive. Therefore, the transition probability Ω_j, for $j = 0...(k-1)$ is as follows.

$$\begin{aligned}\Omega_j &= \Pi_{v,j} + \Pi_{s,j} \\ &= \frac{1}{\lambda T} \sum_{j=k+1}^{K} \gamma_{v,j} + \frac{1}{\lambda T}\left(\gamma_{s,k} - \sum_{j=k+1}^{K} \gamma_{v,j}\right) \\ &= \frac{\gamma_{s,k}}{\lambda T}. \end{aligned} \tag{9}$$

The transition probability, Ω_j for $j = k$ is as follows.

$$\begin{aligned}\Omega_K &= \Pi_{v,K} + \Pi_{s,K} \\ &= 1 - \rho_c - \frac{1}{\lambda T}\sum_{j=1}^{K} j\gamma_{v,j} + \frac{\rho_c(\rho-1)}{\rho} + \frac{1}{\lambda T}\sum_{j=1}^{K} j\gamma_j \\ &= \frac{\rho - \rho_c}{\rho}. \end{aligned} \tag{10}$$

The expected number of packets in the queue N_q, is as follows.

$$N_q = \sum_{k=0}^{K} k\Omega_k \tag{11}$$

Or,

$$N_q = \frac{1}{\lambda T}\sum_{j=1}^{K-1} j\gamma_{s,j} + K\left(\frac{\rho-\rho_c}{\rho}\right) \tag{12}$$

The queueing delay of a packet at a node i can be obtained using Little's law, which is as follows.

$$\delta_q = \frac{E(N_q)}{\lambda_c} \tag{13}$$

where λ_c is the effective arrival rate, $\lambda_c = \lambda(1-P_b)$, and P_b is the loss probability (see Figure 3). Since $\rho_c = \rho(1-P_b)$, therefore,

$$P_b = \frac{\rho - \rho_c}{\rho} \tag{14}$$

Substituting (12) into (13), we have the following expression

$$\delta_q = \frac{\frac{1}{\lambda T}\sum_{j=1}^{K-1} j\gamma_{s,j} + K\left(\frac{\rho-\rho_c}{\rho}\right)}{\lambda(1-P_b)} \tag{15}$$

Using (14) and (15), we have

$$\delta_q = \frac{\overline{S}}{\lambda T \rho_c} \sum_{j=1}^{K-1} j\gamma_{s,j} + \frac{K}{\lambda}\left(\frac{\rho}{\rho_c} - 1\right) \tag{16}$$

3.2 End-to-End Delay

As mentioned earlier, the delay incurred at a node consists of queuing delay, packet transmission delay, and MAC access delay. We assume that each packet is transmitted in exactly one time slot and this assumption is satisfied when $\frac{L}{C} \leq t_s$, where t_s is the slot time. Let S_{max} be the maximum number of time slots per TDMA frame. Then, average number of time slots a node needs to wait before transmitting the packet is $\frac{S_{max}}{2}$. Therefore, the average access delay is $\frac{S_{max}}{2} t_s$. Then, the expression for a single hop delays incurred at a node can be written as,

$$\begin{aligned} \delta &= \frac{L}{C} + \delta_q + \frac{S_{max}}{2} t_s \\ &\approx t_s + \frac{\overline{S}}{\lambda T \rho_c} \sum_{j=1}^{K-1} j\gamma_{s,j} + \frac{K}{\lambda}\left(\frac{\rho}{\rho_c} - 1\right) + \frac{S_{max}}{2} t_s \\ &\approx t_s\left(\frac{S_{max}}{2} + 1\right) + \frac{\overline{S}}{\lambda T \rho_c} \sum_{j=1}^{K-1} j\gamma_{s,j} + \frac{K}{\lambda}\left(\frac{\rho}{\rho_c} - 1\right). \end{aligned} \tag{17}$$

Let there be h hops along a path from a given source to a destination, then the end-to-end delays can be given by the following expression,

$$\overline{\Delta} = h\left[t_s\left(\frac{S_{max}}{2} + 1\right) + \frac{\overline{S}}{\lambda T \rho_c} \sum_{j=1}^{K-1} j\gamma_{s,j} + \frac{K}{\lambda}\left(\frac{\rho}{\rho_c} - 1\right)\right] \tag{18}$$

In what follows, we present results and discussion.

4 Result and Discussion

To evaluate the performance of our protocol, we carried out simulations in C++. In our results, we have assumed that there are 100 nodes and the transmission range of each node is assumed to be $250m$ (unless and otherwise stated explicitly) that are distributed uniformly randomly in a region of area $1000m \times 1000m$. The default value of the maximum number of time slots that may be available (if not occupied by neighboring nodes) is 20.

We consider the following parameters: (i) *average end-to-end delay*, and (ii) *QoS success ratio*. The average end-to-end delay denotes average latencies experienced by packets from the source to the destination. The *QoS success ratio* denotes the number of packets that arrive at the destination before the expiry

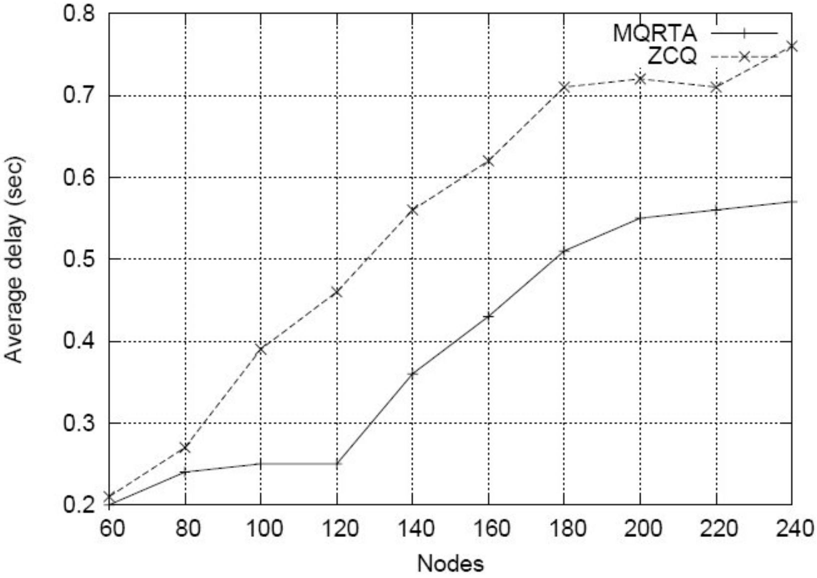


Fig. 4. Average delay as a function of the number of nodes in the network

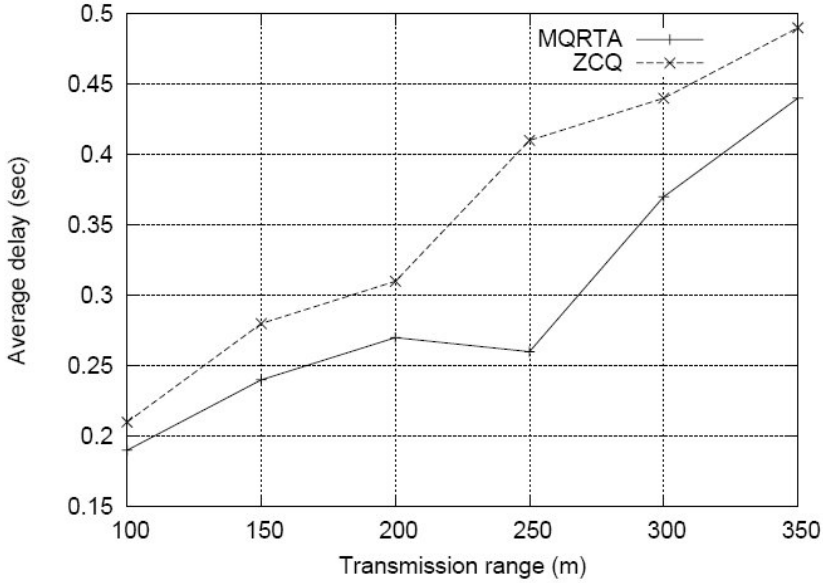


Fig. 5. Average delay as a function of transmission range

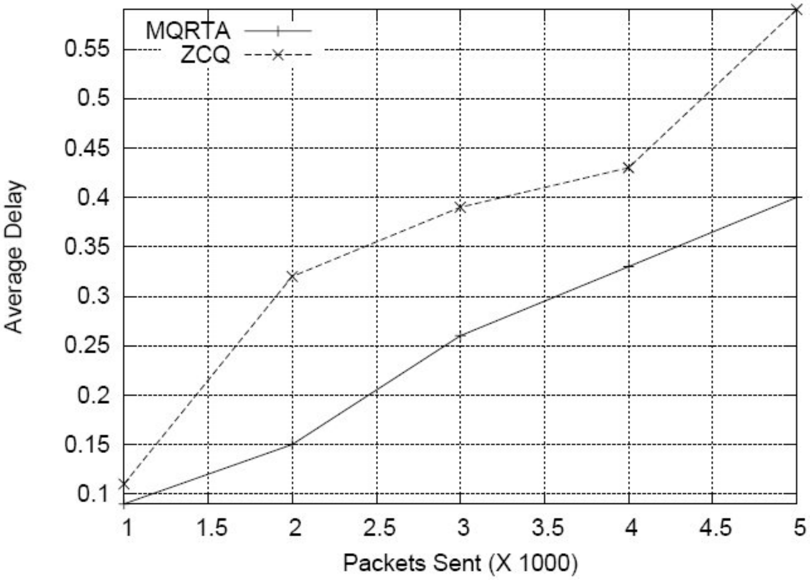


Fig. 6. Average end-to-end delay as a function of the number of packets sent by the source

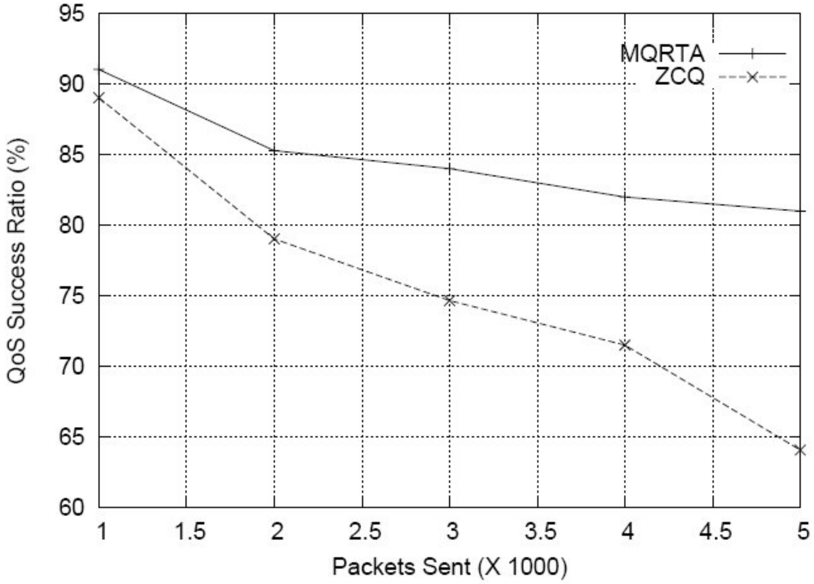


Fig. 7. QoS Success ratio as a function of the number of packets sent by the source

of their respective deadlines divided by the total number of packets sent by the source. We compare the performance of our protocol with ZCQ [4]. We have selected ZCQ protocol because it also provides QoS in a TDMA-based ad hoc network.

Figure 4 shows the end-to-end delay is as a function of the number of nodes in the network. We observe that the delay increases when the number of nodes is increased. The reason is that with the increase in the number of nodes. As a result, the access delay and the end-to-end delay increase. Another observation is that the average end-to-end delay incurred by packets in case of the proposed, MQRTA, is less than those of ZCQ. A possible reason for this observation might be that in the proposed protocol, the source tries to identify a path with minimum neighboring nodes of the nodes lying along the path, however, no such mechanism is employed in case of ZCQ.

Figure 5 shows the end-to-end delay is as a function of transmission range. We observe that as the transmission range of nodes is increased, the end-to-end delay also increases. The reason is that with the increase in transmission range, the number of neighboring nodes of each node along the path increases, and consequently, the access delay is increased. However, MQRTA performs better than ZCQ. This is due to the same reason as that in Figure 4.

Figure 6 shows the average end-to-end delay as a function of the number of packets sent by the source. We observe that as the number of packets sent by the source is increased, the average end-to-end delay increases. With the increase in the number of packets, the queuing delays are increased. In the situation where the number of packets sent by the source is fairly low, there is no significant difference between the average end-to-end delays incurred by packets in case of MQRTA and ZCQ. However, as the number of packets sent by the source increases, the average end-to-end delays incurred by packets in case of MQRTA are significantly smaller as compared to that of ZCQ. The reason is that MQRTA tries to identify a path satisfying average end-to-end delay constraints. Moreover, the access delays are mitigated in MQRTA resulting in the reduction of the total delay, and the packets sent to the destination reach before their respective deadlines. On the other hand, as the path identified by ZCQ becomes heavily loaded, the queuing delays and access delays are increased, thereby increasing the end-to-end delays.

Figure 7 shows *QoS success ratio* as a function of packets sent by the source. We observe that the *QoS success ratio* of ZCQ is significantly low as compared to MQRTA. The reasons are that MQRTA identifies a path between a given source and a destination which is capable of satisfying the bandwidth and delay requirements of a flow of packets. Moreover, the QoS path between the given source and the destination is identified in such a manner so that access delays are mitigated at individual nodes lying along the path. This enables the flow of packets to reach the destination before their respective deadlines or playout times, which accounts for an improvement in the QoS success ratio as compared to ZCQ. In what follows, we conclude the paper.

5 Conclusion

Designing a protocol to provide quality of service in an ad hoc network is a challenging task. In this paper, we proposed a routing protocol which identifies a path between a given source and a destination that satisfies QoS constraints in terms of end-to-end delays the packets may incur from a given source to a destination in a TDMA-based ad hoc networks. We evaluate the performance of the proposed protocol through simulations and compared it with ZCQ [4]. We observed that the proposed protocol performs better in terms of the QoS satisfaction ratio.

References

1. Cionca, V., Newe, T., Dadarlat, V.: TDMA Protocol Requirements for Wireless Sensor Networks. In: 2nd IEEE International Conference on Sensor Technologies and Applications, pp. 30–35. IEEE Computer Society Press, Los Alamitos (2008)
2. Djukic, V., Mohapatra, P.: Soft-TDMAC: A Software TDMA-based MAC over Commodity 802.11 Hardware. In: 28th IEEE International Conference on Computer Communications, pp. 1836–1844. IEEE Press, Los Alamitos (2009)
3. Sahoo, A., Shanti, C.: DGRAM: A Delay Guaranteed Routing and MAC Protocol for Wireless Sensor Networks. IEEE Transactions on Mobile Computing 9(10), 1407–1423 (2010)
4. Zhu, C., Corson, M.S.: QoS Routing for Mobile Ad hoc Networks. In: 21st Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM), pp. 958–967. IEEE Computer Society Press, Los Alamitos (2002)
5. Al Soufy, K.A.M., Abbas, A.M.: Enhancing Bandwidth Reservation Guarantees for QoS Routing in Mobile Ad Hoc Networks. In: 16th Annual IEEE International Conference on High Performance Computing (HiPC), pp. 195–204. IEEE Computer Society Press, Los Alamitos (2009)
6. Smith, J.M.: Properties and Performance Modelling of Finite Buffer M/G/1/K Networks. Computers and Operations Research 38, 740–754 (2011)
7. Bose, S.K.: An Introduction to Queueing Systems. Kluwer Academic and Plenum Publishers (2001)
8. Ajiboye, J.A., Adediran, Y.A.: Performance Analysis of Statistical Time Division Multiplexing Systems. Leonardo Electronic Journal of Practices and Technologies 9(16), 151–166 (2010)
9. Tian, N., Zhang, Z.G.: Vacation Queuing Models Theory and Applications. Springer, Heidelberg (2006)
10. Doshi, B.: Single Server Queues with Vacations. In: Takagi, H. (ed.) Stochastic Analysis of Computer Communication Systems. Elsevier Science Publishers B.V, Amsterdam (1990)
11. Teghem, J.L.: On Finite Capacity Queuing Systems with a General Vacation Policy. Journal of Applied Mathematics and Stochastic Analysis 13(4), 411–414 (2000)
12. Lee, T.T.: M/G/1/N Queue with Vacation Time and Exhaustive Service Discipline. Operations Research Society of America 32(4), 774–784 (1984)

A Model Approach for Using GIS Data in E-Governance Systems

Aakash Trivedi

Dept. of Computer Sciences
SRM University, Ghaziabad India
aakash.trvd@gmail.com

Abstract. In past few years, GIS systems have marked a significant development in India. Governments are now planning to utilize the GIS data collected in E-Governance Systems. But when it comes to actual implementation it is often realized that the geographical data still lacks significant comprehension. Example when it was decided to utilize the global imagery of India in Land Record Management, authorities felt that no matter how efficient the data is but it is still inferior to the imagery done by Google and Yahoo. The matter of fact that Google Maps indexes precise data about roads, colonies and even buildings and its all due to people's participation from apps like Wikimapia etc. In this paper we are presenting a Referenced Approach to merge the data from different GIS systems and implement them in e-governance systems. It also describes the setup of Land record Management system, Survey Management and Personal Identity System based on this approach.

Keywords: IEGS, E-Governance, GIS.

1 Introduction

This paper proposes a new approach for record management in E-Governance Systems. The new approach aims to enhance the applicability and robustness of E-Governance systems. To give a brief introduction to the approach, let’s visualize the scenario of any E-Governance system for a particular department. Let’s say the Survey Department. The current functionality allows survey authorities to append, alter and re-consider survey records online and analyze them.

The current implementation is good but the records maintained by the system are hard to analyze until some manual or additional module of computation marks the records on graphical presentations, like Charts, Maps etc. These records are usually marked on a map and then reports are generated based on the readings. Any reports issued for analysis are generally a *Distribution Map* plotted based on any candidate filter. Our approach is “**Why not append and manipulate records on the map at its first place**”. We are proposing E-Governance Systems to use a referenced architecture to maintain their records centralized on a robust Land Records.

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 438–448, 2011.

2 Improving Comprehension

In this section, we will demonstrate a scenario of optimizing GIS data by adding comprehensive key's to it. i.e we will append local data like roads, parks, railway tracks etc to make the GIS data more comprehensive. The best thing about this task is that government already keeps geographic information about land surveys, plot locations, and ownerships for taxation and legal purposes; converting it into maps and other GIS-related information is only a matter of time, resources, and incentive. So for this we will utilize local data maintained by government firms and append it in our systems. We will use Google Map's Free API's for demo purposes

This is an example of local data maintained by any government firm. Such data includes a rough description about roads, colonies, railway tracks, Popular Organizations, Bridges, Buildings, Airports, Railway Stations and all other stuff that is required to make a map speak. Since the data shown in the figure below is not accurate so the major challenge is to make the transition between this data and actual collected GIS data.

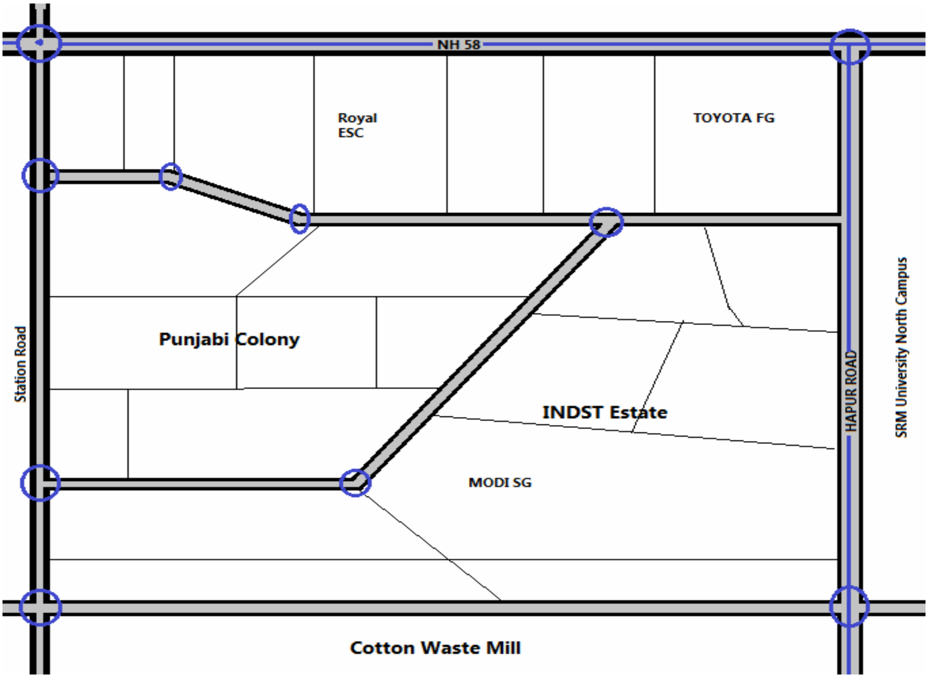


Fig. 1. Example Local Data Maintained by Government

The first task is to break entities like roads, tracks etc. into segments defined by intersections. These intersections are defined by each interior bends found. The intersections identified in above figure are marked by blue circles. Now these intersections are used to describe segments of roads. The following table gives a brief idea about the data structure utilized for this.

Table 1. Database Structure to be Utilized

ID	Street	Start Lat	Start Long	End Lat	End Long
100	NH-58	80.1020	46.9087	80.1206	46.9087
101	SRD	80.1020	46.9087	80.1020	46.8762
102	PCIL	80.9875	46.8099	80.9875	46.9087
103	NH-58	80.1206	46.9087	80.1680	46.9087

Now to represent segments of roads, the identifiers of intersections can be used to address a segment by addressing its start and end intersections and store in different dataset. In this way we can conveniently relate segments having common intersection points. The segments can be separately combined to represent a full structured unit like road, Railway Track. This technique is much beneficial over traditional Full Layout scheme as the segments can be filtered according to the end coordinates of UI's Window. This saves a lot of buffer space and computation required in fetching the whole record layouts.

Now, the above discussion is one approach for appending comprehensive data but many other approaches can be utilized. One may also like to assure a different database or data package for this comprehension data and other data utilized in GIS Applications. Indexing both datasets with same keys creates a way lot of problems.

Now onwards, we will discuss the applications of GIS data in E-Governance Systems. We have developed a web application named IEGS (Integrated E-Governance System) which implements the whole approach discussed in this paper. The application will soon be available with source codes on project web site. The application is now available in two flavors. One is the PHP-MYSQL implementation which has all the basic functionality of the approach and is available as a Ready to deploy version 1.4 which can be easily deployed without much configuration alterations on any Apache, PHP, MySQL web server. The second flavor is the J2EE implementation in which the applicability has been extended to a wide range by integrating several E-Governance systems. We have utilized several features provided by open source java libraries to enhance the robustness of the whole application. The application will be globally deployed as soon as sufficient resources are available.

3 Utilizing GIS Data in E-Governance Systems

E-Governance in India has evolved to a wide scope of functionality within recent years. Now almost every state has succeeded in computerizing its official records and is utilizing it at peak level. The records maintained by government are openly accessible to citizens and the systems provide adequate functionality but even though the implementation seems to be incomplete, i.e. when it comes to applicability, many considers the computerization as being less beneficial as it is supposed to be. One major drawback pointed in the systems after surveys that the E-Gov systems utilized by different departments are not inter-related. One Department hardly utilizes the data maintained by the other system on the same processing object. This paper presents a new approach for E-Governance Systems to use a **Referenced Architecture** for related data records by introducing a sample E-Governance System IEGS (Integrated E-Governance System). We realized that most of the Govt. record keeping is centralized on a "**processing site**". Here processing site is the address (or say postal address) the actual location from where the data has to be or been collected. That's why all Govt. records always consists an "**address**" Attribute. Through this paper, we are proposing to use a new, more accurate attribute i.e. "**map coordinates**". Here map coordinates represents the actual Latitude/Longitude matrix of the concerned location. IEGS first maintains the map coordinate information by its powerful LRMS (Land Record Management System) which provides a unique Land ID to every land officially registered. This unique ID acts as a "**digital address**" for all other record keeping systems.

To inter-relate data from one e-governance system to another system, it is mandatory to have a key attribute that serves as a common pot in all e-governance systems. Initially, we discovered the following two candidates that seem to be passing the raw conditions.

- Individual Personal Identity Record
- Land records

But after further analysis, we realized that we can-not use Individual person's Identity as a key element in our system, because there are many systems that don't impose on each person e.g. you can-not issue an Electricity/house/Phone Bill on each person. Neither we can easily maintain a huge database of all person and design architecture around it. Keys are supposed to be relatively smaller data-sets which can be easily maintained so the above discussion concludes in using land records as key element in our system.

3.1 LRMS (Land Record Management System)

The LRMS is an integral module of IEGS and represents any Land Record Systems run by Revenue departments of states. LRMS keeps record of all known lands legally registered within the scope. It not only provides a unique identity to each land, but also produces a satellite imagery of the land record as done by Google Imagery Data. LRMS is the central node which is responsible for database to map and vice versa translations. It dynamically generates Java Scripts to drive Map resource API's to produce efficient Imagery. In the following paragraphs, we will discuss the GIS strategy implemented to provide a robust Imagery of land records.

3.2 The GIS Strategy

It is clear from the previous paragraphs that our implementation of IEGS will stay focused around a robust Mapping and Imagery System. Revenue departments of several states has worked hard for years to use Remote Sensing Satellites to produce an actual imagery corresponding 9FT from ground. Research on this is still under progress and is been quite a success in recent years but at certain phases many realized that even if the implementation completes, the data would not be as accurate and comprehensive as done by commercial organizations (*like Google, Yahoo etc).*

The matter of fact that Google and Yahoo indexes precise information about roads, streets and even buildings and all this because of people's participation using applications like WikiMapia, and PlotMash. The best part is that these organizations provide services to use their data in your customized applications for Free through various API's. So, after similar evaluation, we selected three candidates that fit into the criterion

- Google Map's API
- Yahoo Map's API
- NASA World Wind Open Source

Although the third one is open source and the implementation will not require any outer resources but it has some limitations. NASA World Wind comes with a wide variety of functions for manipulating and presenting imagery data but still the data lacks comprehension. It barely has any village or small towns plotted on it so it requires a way lot of effort to first initialize the system with preliminary data. No one can assume plotting his home when he has to first plot his town in an unlabeled globe. So as a conclusion we selected Google Map's API to make Map-Data translations.

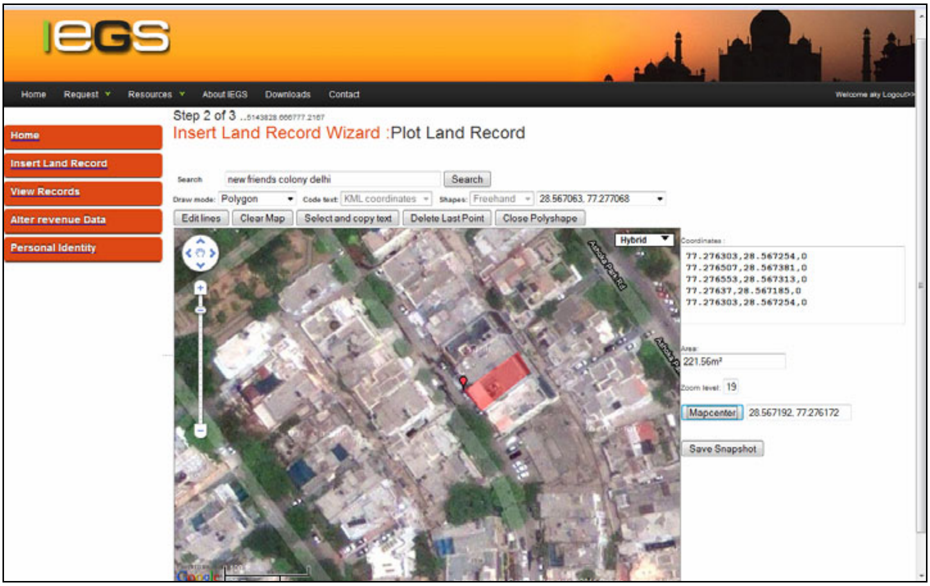
3.3 Google Map's API

Google Maps has a wide array of APIs that let anyone embed the robust functionality and everyday usefulness of Google Maps into your own website and applications, and overlay your own data on top of them So as per the GIS strategy for this system, IEGS utilizes the powerful Google Maps API to make location -> coordinate and vice versa translations. IEGS dynamically generates the Java-Scripts which drives the Google Map object with extended functionalities. The image below shows a Land Record maintained by the PHP Implementation of IEGS. (blue line over the map is stretched by IEGS to mark the boundary of concerned Land Records)

3.4 The Record Entry Interface

This section briefly describes the UI provided by IEGS to locate a land record an the map. A snapshot from J2EE implementation of IEGS is shown below.

IEGS provides an easy to use Clickable Map interface to ease the user in making land record entries. User can use the search box to search the place by city/town name or the by any popular location nearby probably indexed by Google. The GMap object in GMAP API 3 allows tracing the mouse pointer's location as per Latitude-Longitude plot on map. IEGS just adds a Java Script listener that listens to the click event on GMAP abject and then saves the GLatLong respose. The response from user is appended to a dataset to form a Latitude/Longitude Matrix. JavaScript Functions allows user to draw a polygon shaped by simply clicking boundary points on map. The interface dynamically calculates the area as soon as the polygon finishes. This area calculation gives a rough picture of the actual area and can be used in filtering and surveying. The area calculation has been described in the next section. One can conclude that the record entry procedure for actual digitized mapping is quite simple and convenient. User has to just plot the land boundary on map and it automatically produces a Latitude-Longitude Schema for the record. Then the user has to append necessary useful information to the record and after successful record insertion, LRMS assigns a unique Identifier to the record which can be referenced by other systems. For enhanced applicability, IEGS generates a bar code which can be processed manually.



Step 2 of 3 ...5143828.666777.2167

Insert Land Record Wizard :Plot Land Record

Search: new friends colony delhi

Draw mode: Polygon

Code text: KML coordinates

Shapes: Freehand

28.567063, 77.277068

Coordinates :

```
77.276303,28.567254,0
77.276507,28.567381,0
77.276553,28.567313,0
77.27637,28.567185,0
77.276303,28.567254,0
```

Area: 221.56m²

Zoom level: 19

Mapcenter: 28.567192, 77.276172

Fig. 2. The Record Entry Interface

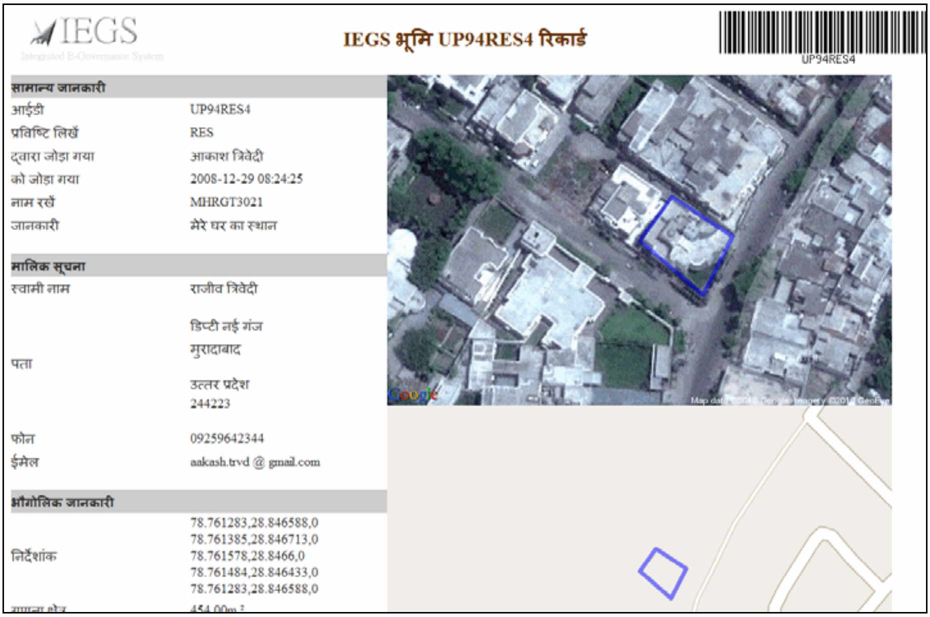
3.5 Area Calculation

In computing the area of an arbitrary region, the human method would be to break it down into simple components, such as triangles, and then sum up the individual areas of these smaller pieces. A triangle's area is just half the base times the height, and solving for the height is possible given enough of the angles and side lengths. Breaking down a complex shape can be a tricky task, however, even for a human. In order for a computer to be able to solve for the area of an arbitrary region, a systematic approach must be developed—one that a simpleminded JavaScript function can reliably apply in all situations. To derive such a method, we represented each point around a figure's perimeter as a coordinate pair labeled x1 and y1, x2 and y2, and so on. After this area can be calculated using the formula:

$$A = \frac{1}{2}\sum_{i=0}^{N-1}(x_{i+1}y_i - x_i y_{i+1})$$

3.6 Land Records by IEGS

The reports generated by IEGS are in both HTML and PDF formats which makes it more printer-friendly.



IEGS
Integrated E-Governance System

IEGS भूमि UP94RES4 रिकार्ड

UP94RES4

सामान्य जानकारी	
आईडी	UP94RES4
प्रविष्टि लिखें	RES
द्वारा जोड़ा गया	आकाश त्रिवेदी
को जोड़ा गया	2008-12-29 08:24:25
नाम रखें	MHRGT3021
जानकारी	मेरे घर का स्थान
मालिक सूचना	
स्वामी नाम	राजीव त्रिवेदी
पता	डिप्टी नई गंज मुरादाबाद उत्तर प्रदेश 244223
फोन	09259642344
ईमेल	aakash.trvd @ gmail.com
भौगोलिक जानकारी	
निर्देशांक	78.761283,28.846588,0 78.761385,28.846713,0 78.761578,28.8466,0 78.761484,28.846433,0 78.761283,28.846588,0
गणना क्षेत्र	454.00m ²

Fig. 3. IEGS Land Report Demonstrating Localization Support

IEGS includes localization support for Hindi Language. Other popular foreign languages are also included in repository. IEGS appends a unique barcode to every generated report to ease further official and legal processing. The purpose of IEGS is to serve as a powerful platform where Land Records can be maintained in a robust, convenient and efficient manner with accurate description and presentation using new technologies, to include features of traditional paper record keeping by maintaining a copy of legal documents, to provide end user an interface though which complicated record keeping becomes convenient, to provide sufficient tools and logic through which system can be further extended to support dependent applications, To consider the modern record processing aspects in implementation. Example: Barcode generation in reports generated for fast manual processing.

4 Utilizing Data in Other Systems

In IEGS, we have utilized an internal reference strategy to inter-relate data components collected from same processing site. For instance, Personal Identification System can be implemented by appending identification information of each person concerned with a land record like the existing system of Rashan Cards. And it is quite convenient as any such processing is usually done by making authorities/executives visit homes within the area and collect data.

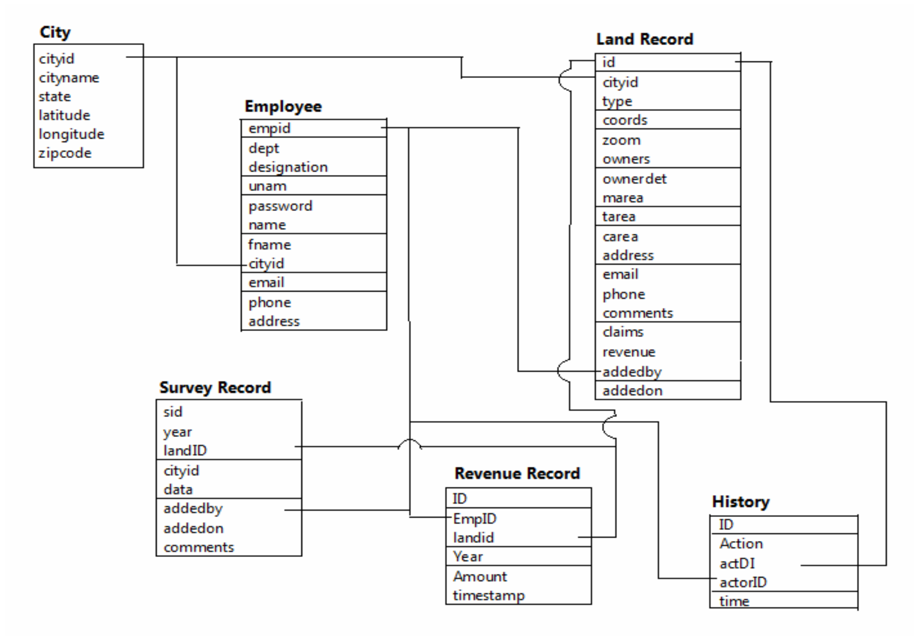


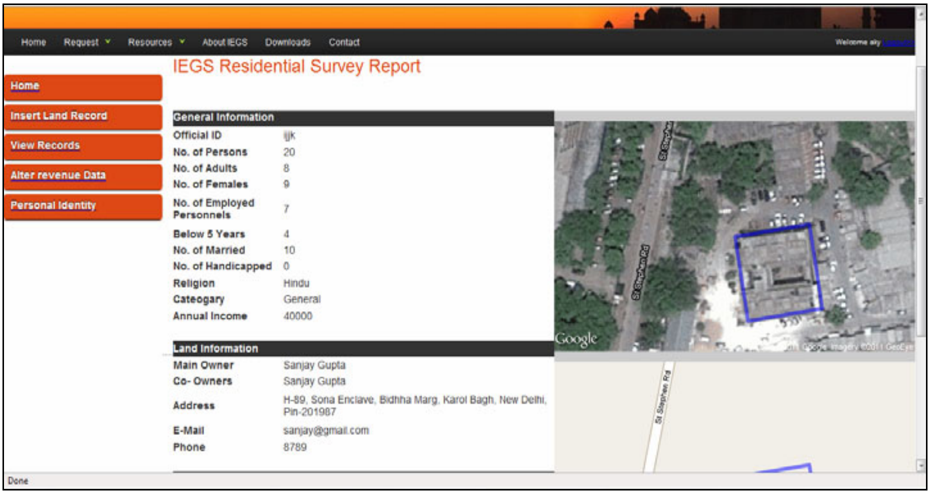
Fig. 4. IEGS Internal Database Schema

Now since the data is actually referred to another system, authorities can easily browse a whole bunch of related record collected about the object by using any of Identifiers used i.e. a Land Record ID will fetch the PIN (Personal Identification Numbers), Voter ID's, PAN's (Personal Account Numbers) and other sought of related data maintained by other systems. This can be assumed as some sought of high profile CIA module that we see in movies like one enters a key element (name, phone, add etc.) and all the related information including satellite imagery appears on the screen.

As an implementation of this model, we provided data outlet modules with every implemented system. These data outlets are just server side data extractors which matches the parameters and returns the data related to concerned object. Data is returned as a XML file which can be processed by any system. At last we have a central Master System which acts as a Master Detail Page which processes Identifier queries on different systems. Once a match is found, it searches in the Reference directories of other systems for the queried object and the details of those entities are also included. This system deals with authentication protocols which identifies the requesting party to be an internal module of IEGS.

The Data Base Schema of IEGS demonstrates how the Land ID can be referenced in other records and can actually be used as a digital address. The diagram is self explanatory. Referencing data adds to the functionality of each system as access to other system's data can help a system make intelligent decisions and at the time of presentation, the whole layers can be summed up as a master detail page.

The database scheme shown above clearly defines the references made internally between different records. Below is a snapshot of Residential Survey data maintained by an internal *Survey Dept.* implementation in IEGS. Figure shows the data indexed by survey department on Land Record RES-51438-09. Figure5 demonstrates fetching of related data automatically without any external linkages.



IEGS Residential Survey Report

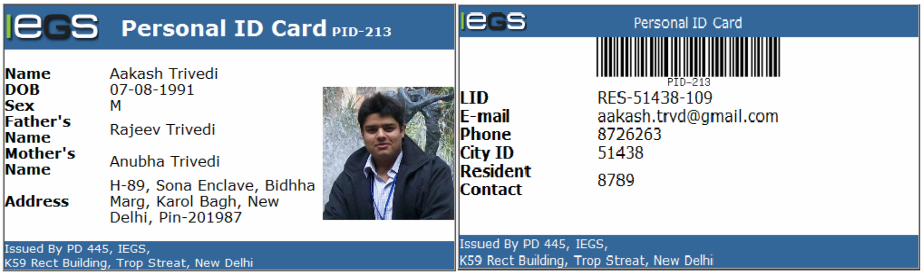
General Information	
Official ID	ijk
No. of Persons	20
No. of Adults	8
No. of Females	9
No. of Employed Personnels	7
Below 5 Years	4
No. of Married	10
No. of Handicapped	0
Religion	Hindu
Cateogary	General
Annual Income	40000

Land Information	
Main Owner	Sanjay Gupta
Co- Owners	Sanjay Gupta
Address	H-89, Sona Enclave, Bidhha Marg, Karol Bagh, New Delhi, Pin-201987
E-Mail	sanjay@gmail.com
Phone	8789

Fig. 5. Survey Record Appended on a Land Record

4.1 Implementing External Systems

In this section, we will discuss the extension services provided by IEGS to implement other independent systems centralized over Land Records. As an example, we implemented the Personal Identity System as an extension system for IEGS. The client system utilizes standard service outlets provided by IEGS to access and manipulate data in both systems. Although the database hierarchy of client system only includes a reference identity for any data to be retrieved during further manipulation. Thus any changes performed internally to records in IEGS will reflect in all external applications.



IEGS Personal ID Card PID-213

Name	Aakash Trivedi
DOB	07-08-1991
Sex	M
Father's Name	Rajeev Trivedi
Mother's Name	Anubha Trivedi
Address	H-89, Sona Enclave, Bidhha Marg, Karol Bagh, New Delhi, Pin-201987

Issued By PD 445, IEGS,
K59 Rect Building, Trop Streat, New Delhi

IEGS Personal ID Card

PID-213

LID	RES-51438-109
E-mail	aakash.trvd@gmail.com
Phone	8726263
City ID	51438
Resident Contact	8789

Issued By PD 445, IEGS,
K59 Rect Building, Trop Streat, New Delhi

Fig. 6. Personal Identification Card Issued by IEGS

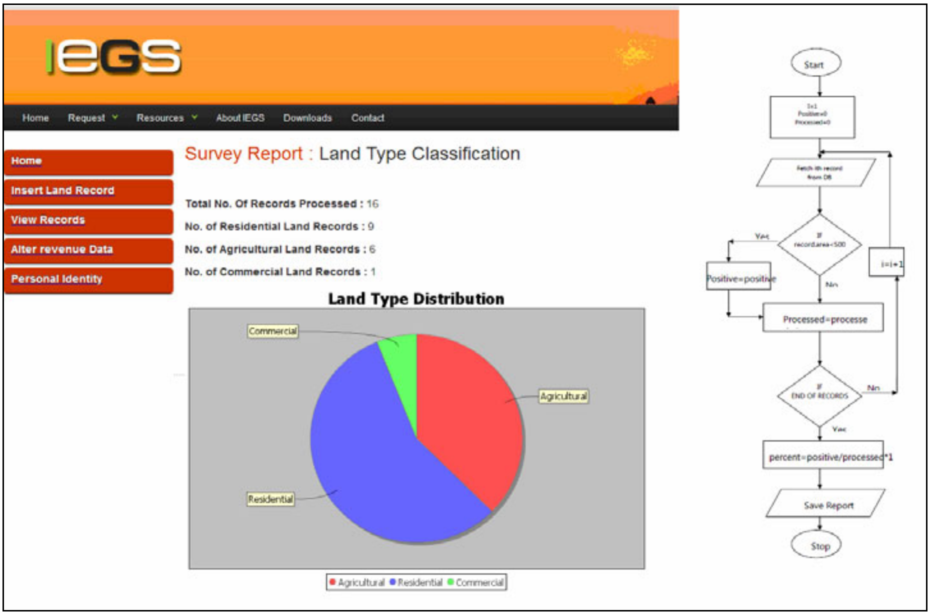
As per the Personal Identification System, we all are aware that each and every person has a permanent address as a candidate for data linkage. Thus the link is provided by appending an IEGS LID reference to the Personal Identification Records. If anyone wants to access the details related to Personal Identification Records, The system retrieves the data already maintained in IEGS and show it along with previous data.

5 Report Generation Strategy

We have seen in previous paragraphs how to reference foreign data to LRMS, here we will see how IEGS extends its applicability by mapping the referenced data on the map. The report generation includes:

- Filtering stored data on basis of attributes.
- Presenting the filtered data using graphs charts and maps.

The scenario demonstrated here is the Report Generation for Land records having area less than 500 m2. The algorithm for %age calculation has been shown below. This is only one scenario but we can implement far more, for example we can just filter the records and log their LRMS ID which can be easily appended on a map to present a Data Distribution Map. Similar strategies can be implemented to generate graphs and charts. We can easily color code the data from key datasets to enhance presentation. These reports are conveniently saved and backups are maintained by IEGS to ease research and survey studies.



Survey Report : Land Type Classification

Total No. Of Records Processed : 16

No. of Residential Land Records : 9

No. of Agricultural Land Records : 6

No. of Commercial Land Records : 1

Fig. 7. Pie Chart Generated by IEGS (*Left*) Percentage Calculation Flow *(Right)*

After the percentage calculation, the presentation layer presents the processed data in forms of pie charts, maps etc. A dynamically generated pie chart from this approach has been shown in figure 8, a snapshot from J2EE implementation of IEGS.

6 Advantages of Using Proposed Model

The major advantage of using this referenced model is that data maintained by IEGS can be easily filtered on various attributes without much effort. The system dynamically generates bar charts, pie charts based on various attributes of project. The centralized module is also able to generate a dynamic map pointing the filtered land records of data referenced in the system. These graphical presentation tools will ease the authorities in analyzing the data to much extent. Moreover since the data maintained by internal systems are governed by a central authority, this provides the functionality of one click access to records to the authorities, which diversifies the scope of the whole application.

The Systems which can utilize this model includes a wide range of record keeping systems that can be merged with LRMS. Some of them are mentioned below:

- Telephone Directory *(Associating phone records that belongs to a particular land record)*
- Election Commission People's Directory *(Voters from a particular Land Record)*
- Electricity (or other) Billing System *(Bill per Land ID utilizing Services)*
- Geological Info System *(Land Records having geo-importance like Mountains, plateaus, national parks etc.)*
- Billing Systems like telephone bills, electricity bills on land basis.
- Personal Identification System
- Survey Commission of India
- Education to all system (*retrieves under 15 student lists and their locations*)

Acknowledgements. There are far too many people to thank; many people have contributed in the development of this paper. I owe my deep regards and honor to express our gratitude to my parents and all the faculty members of SRM University for providing me invaluable support, guidance, help and inspiration all through this paper.

References

1. IEGS Project Home Page, `http://www.iegs.vud.in`
2. LRMS Project Home Page, `http://www.lrms.vud.in`
3. Michael, P., Sambells, J., Turner, C.: Beginning Google Maps Applications with PHP and Ajax. Apress (2006)

GSM Based Power Management

Kaukab Naz, Shahida Khatoon, and Ibraheem

Department of Electrical Engineering
Faculty of Engineering and Technology
Jamia Millia Islamia
New Delhi-110025, India

Abstract. At present, secondary distribution network is controlled manually in most of the places. The limitation of the system is that there is no feedback on the operation. Maintenance staff has to physically check the correct operation. The system has an isolator / changeover switch at the incomer end and a contactor for switching the power. There are no protections against overloads and short circuit faults. There is no metering facility in the system and the revenue calculations are done manually. In this paper a control system is designed for remote operation of LV feeders for continuous monitoring and load shedding with GSM interface with central control station SCADA with two way communication using SMS messages. In this paper a prototype model is developed by integrating SCADA control with GSM technology using the logic Twidosoft v.3 for continuous monitoring and obtaining parameters from the field and remote control of LV feeders. The proposed scheme is intended to give better and more control options, performance, safety to equipment & people and help in reducing downtime by way of suggesting preventive maintenance automatically by the system. Offered system includes the metering functions, so that the meter readings can be downloaded from remote Central Control Station using the same GSM Modem and GSM Network. Automatic re-switching by remote control to enable supplies to be restored quickly to all consumers is a current feature of this scheme. In this project various hardware equipments are configured using IFIX software. Many screens are created for sending and receiving reports. A program is written in MATLAB Editor for determining the channel capacity of GSM network and then the optimum message transfer rate is calculated for this type of application. The scheme has been successfully tested and the analytical discussions and results are presented for the same.

Keywords: GSM, LV feeders, PLC, SCADA, SMS, Optimum message transfer.

1 Introduction

Present research has been carried out by the author as her M. Tech project work which she did at Schneider electrics, India. The objective of the present project was to design a control system for remote operation of LV feeders for continuous monitoring and load shedding with GSM interface with Central control station SCADA with two way communication using SMS messages. At present, secondary distribution network

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 449–462, 2011.

is controlled manually in most of the places. The limitation of the system is that there is no feedback on the operation. Maintenance staff has to physically check the correct operation. The system has an isolator / changeover switch at the incomer end and a contactor for switching the power. There are no protections against overloads and short circuit faults. There is no Metering facility in the system and the revenue calculations are done manually. The proposed scheme is intended to give better and more control options, performance, safety to equipment & people and help in reducing downtime by way of suggesting preventive maintenance automatically by the system. Offered system includes the Metering functions, so that the meter readings can be downloaded from remote Central Control Station using the same GSM Modem and GSM Network. Automatic re-switching by remote control to enable supplies to be restored quickly to all consumers is a current feature of this scheme. Integrating two different network architectures involves many details and critical issues, and it is not an easy task to extract out all of the crucial concepts and design specifics. Instead, we attempt to present a network platform and highlight some of the functional requirements for enabling relaying of calls in GSM and integration of SCADA for remote monitoring and control of secondary distribution network. The proposed solution, has a TWIDO Controller which is very flexible in communication with different devices and capable of handling multiple networks. This makes TWIDO controller to communicate with Central Station over GSM and also to the Field simultaneously. The TWIDO Controller does the additional function of collecting data and transmitting it to the Central Control Station, since controller as it has two communication ports (1xRS 485 Modbus, 1xRS 232). The controller shall communicate with the Central Control Station and carry out the local controls. The controller needs to be switched to cater to the Energy Meter inputs.
The system architecture is as follows:

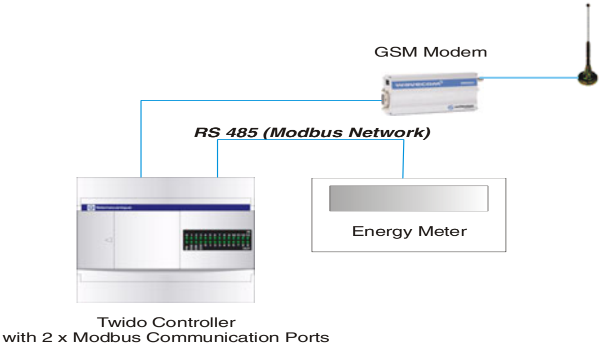


Fig. 1. System architecture

The Central Control Station has a PC with SCADA Software capable of handling Incoming and Outgoing SMS messages and archiving them properly with Message description, Date and time of even etc. The Archives can be created on the basis of Zone & District for easy analysis. The Controller communicates to the Central location on a GSM network and sends SMS to the Control room and/or the maintenance staff on mobile phone about the operation/status and alarms (or as programmed). The unit can also accept commands via SMS from the Central Control Room Computer or from the maintenance staff from mobile phone.

1. Events & Alarms Logging: There are various alarm messages that are coming from the field in form of SMS messages over the GSM Network e. g., Contactor / MCCB failed, Over Current (If that option is selected) and are logged in the database. These alarms can be classified depending on their criticality and areas (e. g., VIP areas etc.). These alarms can be further analysed and reported in predefined format e. g., Total down time due to one particular type of failure for any given month/year etc., Zone of minimum down time etc.
2. Reports: The status of the various field parameters will be collected from the field controllers by SMS and shall be logged into the database of the SCADA Software. Data can be displayed in chronological order or according to any user defined classification. Four Reports are enclosed

    1. Alarm Report
    2. Event Report
    3. Power Report
    4. Power Failure Report

1.1 Benefits

- Automatic control of secondary distribution network based on the programmed time schedule for each circuit in a given zone separately. So that every circuit can be programmed differently and modified from Control Station
- Intimation of the operation (On or Off) to the central location as programmed, by way of SMS
- Parallel SMS to the Maintenance Staff
- Intimation & Logging of alarms/faults by SMS (Contactor / MCCB tripped etc.) from field to the Central Control Station and also intimation to the maintenance staff on mobile
- Security of operation by Passwords
- Remote Force On/Off operations by SMS from Central Control Station or Mobile phone with return confirmation
- Possibility of monitoring load in terms of current consumed to determine actual operation, Central Supervision, Control & logging of the operations and faults. The logging can be archived District and Zone wise.
- Customizable Screens and reports to suit operational and analytical requirements
- The system can be de-fragmented in future to have individual District Control Stations as well in addition to Central Control Station. The Central Control Station can then be reconfigured to do specific tasks.
- As the meter shall provide all electrical parameters along with the Energy consumption viz., all phases Voltages, Currents, Power Factor etc. This additional data can also be available for analytical purpose (if required)

1.2 Transmission of Data

The transmission of this data between Central Location (SCADA) and the GSM Switch can be done in one of the following three ways.

a) By way of SMS – In this case the controller will send the meter reading at a predefined time periodically and same ill be logged by the SCADA Software.
b) By Direct Polling by Central System – The Central SCADA ill call up each unit periodically and fetch details of the metering from each location and update it's records with date and time. In this case all Controllers will have Data Enabled SIM from the GSM Service Provider .
c) By Continuous connection with the Field Control Units – This option enables the Central Station to be always in touch with all the Field controllers and can keep on logging at very frequent intervals. In this case all controllers will have GSM Modems with GPRS / EDGE communication capability provided by the GSM Provider.

In this project the way of transmission of the data between Central Location (SCADA) and the GSM Switch will be done by exchange of SMS Messages.

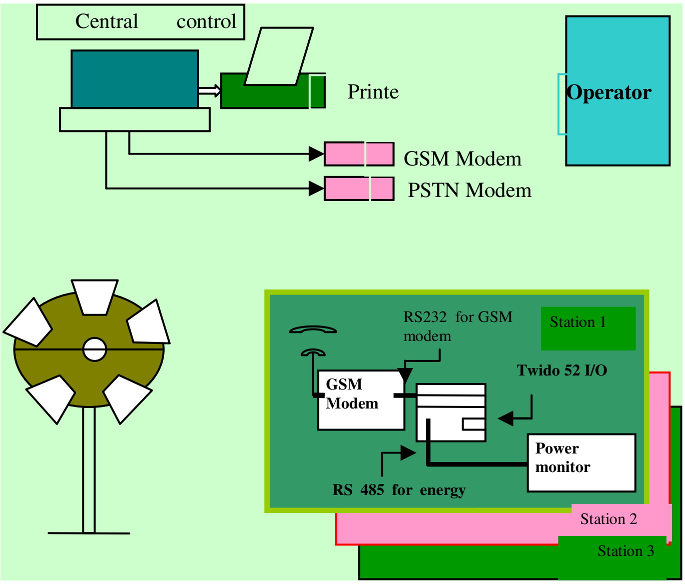


Fig. 2. Block Diagram

1.3 Messages Received at Scada Control Centre

Contactor ON/OFF All the Message are generated based on the command from SCADA/Schedule on Request from SACDA. The message generated is Contactor

ON/OFF Remote on Request from Predefined Schedule .The message generated is Contactor Light ON/OFF Schedule. On Manual Operation. The message is Contactor ON/OFF Local.

a) Limitations in these messages

- If the Message is send by SCADA and switch is not in remote mode the message remains there until any Error / Fault Message is generated. In this condition if a person puts the system in remote mode the CONTACTOR get ON / OFF
- In Schedule Mode if the RTC of the PLC is not synchronized the CONTACTOR get ON / OF at misguided time
- If a person puts the selector switch in local mode and immediately without any delay puts back in remote or off mode the message generated will be wrong ,quite possible the light is Displayed ON but actually OFF or vice-versa or the message is truncated.

b) The Message is generated to get KWH/Voltage/Current/ PF

- Energy Meter readings (KWH)
- Voltage
- Current
- Power Factor (PF)

c) Limitations in these messages

If the CT Ratio of the Meters are not set then the Readings which are coming are not correct.

d) Events OV / OC / UV / UC

The Message is generated when the values go out of range.

Limits for voltage	190-250 V for all three phases
Limits for Current	Adjustable for all three phases
Frequency of these messages are set	15 Minutes

e) Limitation in these messages

one phase is above High Limit and another is Below Low Limit . For SCADA we are changing the The Message is generated when any of the three phase is above the High Limit or Below the Low Limit . It is quite possible that color of the shown value and alarm set value range will also be available for ready reference.

f) Alarms (MCCB Trip/Contactor Fault/Meter Communication Fail)

The Message is generated when there is a fault in the Field Frequency of these messages are set for 30 min

g) Limitations in these messages

- The Message is generated when any of the signal in the field has problem.
- In SCADA the alarm will set the alarm status signal and No RESET / Acknowledge is available i.e. the alarm status bit will always be in fault mode and if a acknowledge (NO SMS)is given at SCADA the person has to wait for 30 Min. to again get the status.
- There is no message for healthiness of the system.

1.4 Messages Transmitted from SCADA

Power ON / OFF all the Messages are generated based on the command from SCADA / Schedule the Message is generated from the authorized Cell & SCADA Only

a) Limitations in these Messages

The Message executes when ever it is received at the controller. The Time Authorization is not inbuilt. Terminating the message without execution if not delivered in specific time.

b) Limitation in these messages

The Information coming is wrong as CT Ratio's are wrong time Adjust it is a request to change the schedule time from remote location

Time Synchronization is the issue if time is not properly synchronized the PLC will not act properly. Current Adjust it is a request to set the limits for Over Current / Under Current alarms. We have to set the Exact digits in the message for up to 100 amp & up to 200 amp Loads. A change In the Controller has to be done.

2 IFIX Software

IFIX and IFIX MMI are the latest Intellution® products in the IFIX family of industrial automation software. provide supervisory control and data acquisition (SCADA) functions.

2.1 IFIX Functions

a) Basic Functions

FIX and FIX MMI are software systems. The core software performs the basic functions that allow specific applications to perform their assigned tasks. The two most basic functions are data acquisition and data management. Figure 3 illustrates the basic functions.

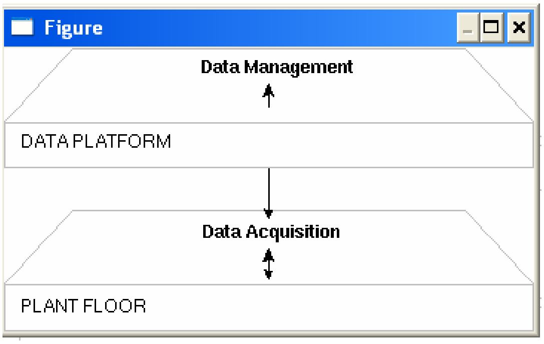


Fig. 3. Basic functions of IFIX

b) MMI and SCADA Functions

The first and most important step in automation is to use plant floor operators and technicians more efficiently. Traditional control room panels may be replaced or augmented with a computer and a graphical display running Intellution software. The computer can provide many of the same functions of the control room, including Monitoring, Supervisory control, Alarming and Control.

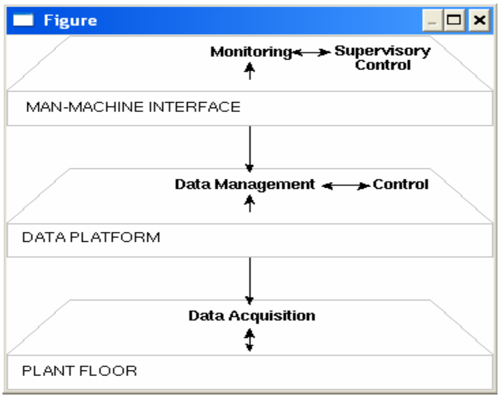


Fig. 4. SCADA functions of IFIX

3 Using the Database Builder

FIX applications read and write data to a process database. The database reads and writes information to the Driver Image Table. The Database Builder program lets you construct a data map of the portion of the process that one SCADA node interacts with. The database can

- Acquire data for data presentation in another applications.
- Generate calculated data from acquired data.
- Apply process control logic to acquired data to create a control loop.
- Receive data from applications for eventual output to the process.

The Database Builder is tool for configuring how to process each data point.

a) The Recipe Builder

The Recipe Builder is a FIX application that allows us to create, manage, and run recipes for our process. If our process requires operators to change many process database values frequently, we can use recipes to make these changes automatically. The Recipe Builder creates master and control recipes. Master recipes are templates from which many control recipes can be created.

Control recipes are production versions intended for normal use. Each recipe can initiate complex process procedures using mathematical formulas, tag groups, and variables to control a process.

4 Presenting Data

Once we have a working database, we can use the other FIX applications to present that data to operators. Data presentation tasks fall into three areas:

- Constructing the Man-Machine Interface (MMI).
- Archiving data and accessing the stored data.
- Generating reports.

5 Collecting Historical Data

The Historical Trending application provides the ability to sample real-time data at operator-specified rates. we can then use this data for long-term process analysis and optimization.

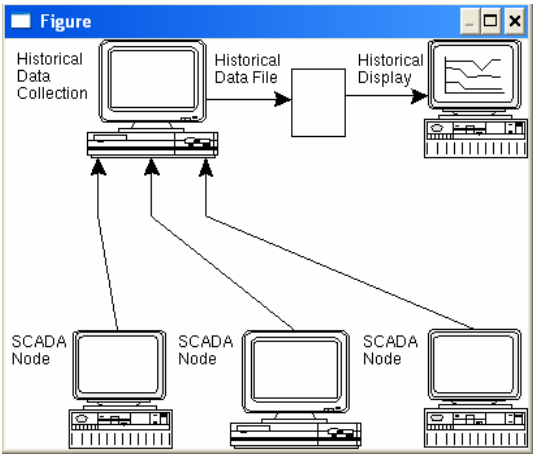


Fig. 5. Historical trending

6 Generating Reports

Most applications require the capability to periodically collect critical data in a report format that plant decision makers can review. FIX software provides a two-step approach to report generation that relies on standard data exchange protocols and spreadsheets. We can create reports using Microsoft SQL for Windows and a collection of optional, Intellution-supplied SQL macros. The Report Generator macros also allow us to

- Access process data.
- Access historical data.
- Create real-time charts.
- Schedule reports for automatic generation.

7 Optimizing the System

We can maximize the performance of FIX software by adjusting various settings. By carefully analyzing overall system architecture, and knowing how data flows from one point to another, we can pinpoint inefficiencies and unnecessary redundancies in order to re-configure our system for optimal performance. t performance is installation-specific, so we have to analyze the performance requirements for each node separately in order to accommodate each node's needs. As shown in Figure10, there are five strategic areas with data flow controls that you can modify in order to enhance overall system performance. These five strategic areas can be referenced according to the respective applications. They include:

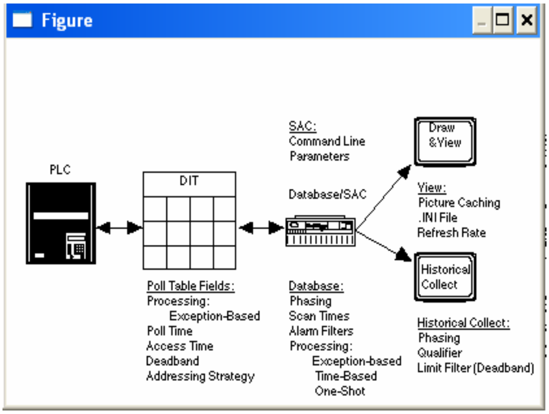


Fig. 6. IFIX SQL Components

- Poll record configuration
- Database
- SAC
- View
- Historical Collect8. IFIX ODBC SQL Software Overview

The FIX Real-Time ODBC SQL Interface software option (FIX ODBC product) allows us to systematically.

- Collect and write real-time process data to one or more relational databases.
- Read data stored in the relational database and write it back to the FIX process database.
- Delete data in relational database tables.
- Back up data and SQL commands to disk if the network fails to maintain a connection to the server or if the server fails.
- Execute backed up SQL commands automatically when the connection to the server is re-established.

8 TWIDO Controller 4.8

This has a limited number of connections built in for inputs and outputs. Typically, expansions are available if the base model does not have enough I/O.A special high speed serial I/O link is used so that racks can be remotely mounted from the processor, reducing the wiring costs for large plants. It can communicate over a wide range of media including RS-485, Coaxial, and even Ethernet for I/O control at network speeds up to 100Mbps. It uses peer-to-peer (P2P) communication between processors. This allows separate parts of a complex process to have individual control while allowing the subsystems to co-ordinate over the communication link. These communication links are also often used for HMI devices such as keypads or PC -type workstations. A rule-of thumb is that the average number of inputs installed is three times that of outputs for both analog and digital. The 'extra' inputs arise from the need to have redundant methods to monitor an instrument to appropriately control another, and from the need to use both manual command inputs to the system and feedback from the controlled system itself.

TWIDO 4.8 works by continually scanning a program. We can think of this scan cycle as consisting of 3 important steps. There are typically more than 3 but we can focus on the important parts . Typically the others are checking the system and updating the current internal counter and timer values.

Step 1-Check input status-First the PLC takes a look at each input to determine if it is on or off. In other words, is the sensor connected to the first input on? How about the second input? How about the third... It records this data into its memory to be used during the next step.

Step 2-Execute program-next the PLC executes your program one instruction at a time. Maybe your program said that if the first input was on then it should turn on the first output. Since it already knows which inputs are on/off from the previous step it will be able to decide whether the first output should be turned on based on the state of the first input. It will store the execution results for use later during the next step.

Step 3-Update output status-finally the PLC updates the status of the outputs. It updates the outputs based on which inputs were on during the first step and the results of executing your program during the second step. Based on the example in step 2 it would now turn on the first output because the first input was on and your program said to turn on the first output when this condition is true.

After the third step it goes back to step one and repeats the steps continuously. One scan time is defined as the time it takes to execute the 3 steps listed above.

9 Logic Developed

TWIDO Controllers can be programmed by either ladder logic or grafset. in this project we employed ladder logic which consists of twelve loops which are as follows

1. Switching of 1 to 4 nos., up to 500 Amp Contactors at the outgoing LV Feeders in MV/LV Substations.
2. Switching ON or OFF the individual Contactor by sending an SMS message from remote Central SCADA
3. The system should send the predefined messages back to the Zonal Staff Mobile phone and the Central SCADA according to the requirement defined for each type of messages.
4. The system should have a provision for remote and local operation with a mechanism to avoid any remote operation during the local/maintenance operations, separate for each contactor.
5. All Local operations should also be reported to the Zonal staff Mobile phone /Central SCADA
6. In case of Contactor coil failure, the system should generate a message with clear text to the zonal staff Mobile phone and Central SCADA
7. The system have an in-built Real Time Clock (RTC)
8. The system should is able to switch ON or OFF any of the Contactor at predefined time stored in the controller
9. The time schedule of the ON/OFF Operation twice a day for each of the 4 contactors should be modifiable from remote by SMS
10. It is be possible to modify/change the authorized mobile phone no. of the zonal staff.
11. After a Power Supply loss, the system generates a message for time of loss and resumption of supply, as soon as the power supply is resumed
12. The system sends all the messages to a short code (e. g., 767 etc. provided by the GSM Service Provider) and not to any mobile no. directly. This will be applicable for all the messages generated from the Field controllers and Central Control Station

10 Design Constraints and Limitations

The design has Constraints and Limitations subject to

- Capabilities and features of Software Platforms and Technologies being used
- Variations in the Performance of application at the time of database backups

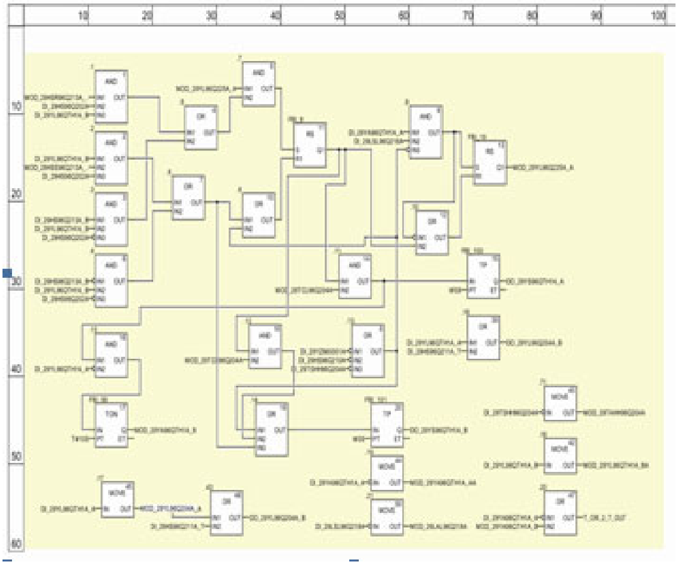


Fig. 7. Ladder logic

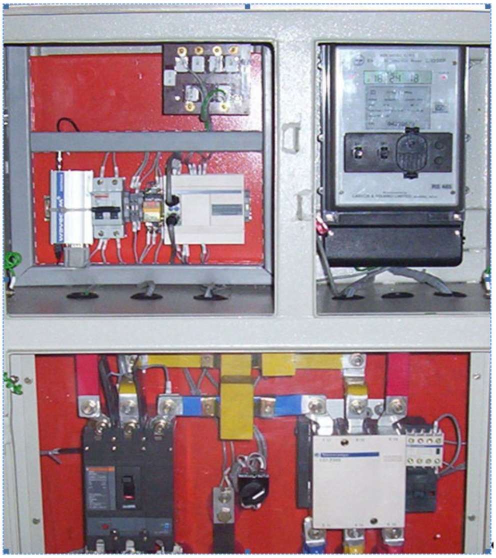


Fig. 8. Actual setup

11 Results and Discussion

We have successfully developed the logic and completed the system integration for the said scheme also tested the scheme, also we determined the optimum message transfer rate for exercising control over secondary distribution network through SMS Which comes out to be 9600 bps. Other than this we wrote a program in MATLAB Editor for obtaining optimum channel capacity for GSM Network. Integrating two different network architectures involves many details and critical issues, and it is not an easy task to extract out all of the crucial concepts and design specifics. Instead, we attempt to present a network platform and highlight some of the functional requirements for enabling relaying of calls in GSM and integration of SCADA for remote monitoring and control of secondary distribution network.

The appropriate integration of secondary systems renders the following benefits - Improved total system reliability, Optimized power system design Cost savings on all systems and plant. The system design decisions on this road towards integrated protection, control and data acquisition systems must be evaluated in many dimensions, not least of which includes customer profiles, expected revenues and capital expenditures template, modified in MS Word 2003 and saved as "Word 97-2003 & 6.0/95 – RTF" for the PC, provides authors with most of the formatting specifications needed for preparing electronic versions of their papers. All standard paper components have been specified for three reasons: (1) ease of use when formatting individual papers, (2) automatic compliance to electronic requirements that facilitate the concurrent or later production of electronic products, and (3) conformity of style throughout a conference proceedings. Margins, column widths, line spacing, and type styles are built-in; examples of the type styles are provided throughout this document and are identified in italic type, within parentheses, following the example. Some components, such as multi-leveled equations, graphics, and tables are not prescribed, although the various table text styles are provided. The formatter will need to create these components, incorporating the applicable criteria that follow.

References

1. Zheng, R., Hou, J.C., Sha, L.: Performance Analysis of Power Management Policies in Wireless Networks. IEEE Transactions on Wireless Communications 5(6) (June 2006)
2. Tynan, R., Marsh, D., o'Kane, D., O' Hare, G.M.P.: Agents for Wireless Sensor Network Power Managemen. In: IEEE International Conference on Parallel Processing Workshop (2005)
3. Tomsovic, K., Bakken, D.E., Venkatasubramanian, V., Bose, A.: Designing the Next Generation of Real-Time Control, Communication, and Computations for Large Power Systems. Proceedings of the IEEE 93, 965–979 (2005)
4. Bill, J.C.W., Mishra, R., Rastogi, N., Zhu, D., Mosse, D., Melhem, R.: Energy Aware Scheduling for Distributed Real Time Systems. In: IEEE Symposium (2003)
5. Aggelou, G.N., Tafazolli, R.: On The Relaying Capability of Next -Generation GSM Cellular Network. IEEE Personal Communication, 40–47 (February 2003)
6. Yumiba, H., Yamamoto, K., Yabusaki, M.: The Design Policy for a GSM-based IMT-2000 network. IEEE Wireless communications, 7–14 (February 2003)
7. Li, H.Y., Crossly, P.A.: Optimum Message Transfer Rate for Distribution Feeder Protection Operating Over Switched Telephone Network. IEEE Transactions on Power Delivery 17(2), 353–358 (2002)

8. Cooper, D., Jeans, T.: Narrow band, Low Data Rate Communications on TheLow Voltage Mains in the CENELE Frequencies –Noise and Attenuation. IEEE Transactions on Powere Delivery 17(3), 724–729 (2002)
9. Aggélou, G.N., Tafazolli, R.: On the Relaying Capability of Next-Generation GSM Cellular Networks. IEEE Personal Communications, 40–47 (February 2001)
10. Takahashi, E.S.C.: Application Aware Scheduling for power Management, pp. 247–253. IEEE Press, Los Alamitos (2000)
11. Dey, S., Raghunathan, A., Jha, N.K., Wakabyashi, K.: Controller Based Power Management for Control Flow Intensive designs. IEEE Transactions on computer aided design of integrated circuits and system 18(10), 1496–1508 (1999)
12. Bruce, A.G.: Reliability analysis of electric utility SCADA Systems. IEEE Transactions on Power Systems 13(3), 844–849 (1998)
13. Reyrolle, P.V.: Integration and other developments in distribution protection: an eskom perspective. In: Developments in Power System Protection, March 25-27, pp. 256–260 (1997); UK Conference Publication No. 434
14. Neorpel, A.,R., Lukander, P., Chang, L.F., Verma, V.K., Lipper, H.: Supporting PACs on a GSM MSC. IEEE Communication Magazine (1996)
15. Laitinen, M., Rantala, J.: Integration of Intelligent Network Services into Future GSM Network. IEEE Communications Magazine, 76–86 (June 1995)
16. Propst, J.E.: Calculating Electrical Risk and Reliability. IEEE Transactions on Industry Applications 31(5), 1197–1208 (1995)
17. Priscol, F.D., Muratore, F.: Study on the integration Between GSM Cellular Network and a Satellite System. IEEE Conference (1993)
18. Cory, B.J.: Computer a ids in power system engineering. Computer-Aided Engineering Journal, 217–225 (December 1988)

Test Effort Estimation-Particle Swarm Optimization Based Approach

S. Aloka, Peenu Singh, Geetanjali Rakshit, and Praveen Ranjan Srivastava

Computer Science & Information System Department, BITS PILANI – 333031 India
{alokas88,geetanjali.rakshi,singhpeenu,
praveenrsrivastava}@gmail.com

Abstract. Test Effort Estimation is an important activity in software development because on the basis of effort cost and time required for testing can be calculated. Various models are available for estimating effort but to some extent all models result in erroneous effort estimation. So there is a need to optimize the effort estimated. Meta heuristic techniques can be used for this purpose, to optimize a problem by iteratively trying to improve a solution, using some computational methods. Particle Swarm Optimization is one such technique which have been incorporated in this work to get good test effort estimates.

Keywords: Particle Swarm Optimization (PSO), Test Point Analysis (TPA), Use Case Points (UCP).

1 Introduction

Software engineering is a systematic approach for developing high-quality software in a cost-effective manner [1]. In software development process, software testing is an important and complex issue. According to literature survey, software testing consumes the maximum effort, which is nearly 50% of the total effort. Test Effort Estimation is an important activity in software development because estimating the effort beforehand enables project managers to allocate resources i.e. budget, time and staff efficiently and avoid future inconvenience [1]. Currently, many effort estimation techniques are available giving satisfactory estimates. However, with increasingly tight schedules and market competition, more accurate estimates are needed.

To handle the complex nature of software engineering, various meta-heuristic techniques[3] like Tabu Search[4] and soft computing techniques[5], are used. Particle Swarm Intelligence is one of the meta-heuristic techniques which is widely used for optimization in various fields. In this paper, results of the two existing techniques, use case points (UCP) [6] and test point analysis (TPA) [7] are optimized to achieve greater precision in test effort estimation using Particle Swarm Optimization (PSO).

The following sections discuss the related works in this field, proposed strategy to apply PSO to optimize the estimates, the results obtained on applying this on a case study, one each for Use Case Points and Test Point Analysis. Then these results are compared with those obtained from existing methods. Finally future scope for this work is discussed.

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 463–474, 2011.

2 Background

Many development effort estimation techniques, like COCOMO model [1], are available from which test effort can be estimated to be nearly 50% of development effort. However, for estimation of test effort directly, not many approaches are available at present. Few techniques which are used are test point analysis and use case points. Suresh Nageswaran proposed use case point analysis technique for test effort estimation [6]. In this technique, test effort is estimated on the basis of use cases and actors of the system by rating them into different categories. Some technical factors are also considered.

Another method is Test point analysis by Drs Eric P W M Van Veenendaal CISA and Ton Dekkers [7], where software is divided into different modules and for each module, some parameters are considered to calculate total test hours.

Results obtained from these techniques can be further optimized to give more accurate results. Various evolutionary techniques have been successfully applied earlier, like software effort estimation using neural networks [8], software effort estimation using soft computing techniques [5] and using Fuzzy Logic [9].

However, Particle Swarm Optimization is known to have some advantages over these techniques. It is much easier to understand and implement due to less number of parameters and has,less sensitivity to the nature of objective function, is less dependent on initial points in search space and quickly converges giving stable and high quality results. Its CPU and memory requirements are also less.

So, PSO algorithm has been used to optimize the effort obtained by Use Case Point Analysis and Test Point Analysis. Unlike other estimation techniques, these consider various parameters related to software testing, giving good results. The results need to be more refined, so optimization using PSO is done.

3 Particle Swarm Optimization (PSO)

Particle swarm optimization (PSO) is an optimization technique developed by Dr. Eberhart and Dr. Kennedy [14], based on social behavior of bird flocking or fish schooling [10]. PSO has many similarities with evolutionary computation techniques like Genetic Algorithms (GA). The system is initialized with a population of random solutions and searches for optima by updating generations. However, unlike GA, PSO has no evolution operators such as crossover and mutation [11]. In PSO, the potential solutions in the search space form particles in the swarm. Each particle has a position and a non-zero velocity at any instant, and is aware of the global best and the local best(self best) positions. They fly through the problem space by following the current global and local best, and approach towards optimality.

4 Use Case Points (UCP)

In Use Case Points following approach is used for effort estimation [6, 12]:

1. First of all number of actors in the system are determined and categorized into 3 levels : simple, average and complex, based on their complexities. Then weights are

assigned to each type of these actors. From this Unadjusted Actor Weight (UAW) is calculated.
2. A similar procedure is applied to use cases in the system and Unadjusted Use Case Weights (UUCW) is determined.
3. Unadjusted Use Case Points is calculated using the formula UUCP=UUCW+UAW
4. Technical and Environmental Factors (TEF) are computed.
5. Adjusted UCP is calculated as: AUCP = UUCP*[0.65+0.01*TEF]
6. Final effort is computed as: Final effort = AUCP* ratio of development man-hours needed per use case point.

5 Test Point Analysis (TPA)

Test point analysis is a technique to measure the black box test effort estimation. TPA calculates test effort estimation for the functions in the system, and also for the whole system, in terms of test points. These test points are calculated based on the importance assigned to the functions by the user [7, 13]. Quality characteristics like functionality, usability, security, efficiency, etc are taken into account in this technique with proper weights assigned to each.

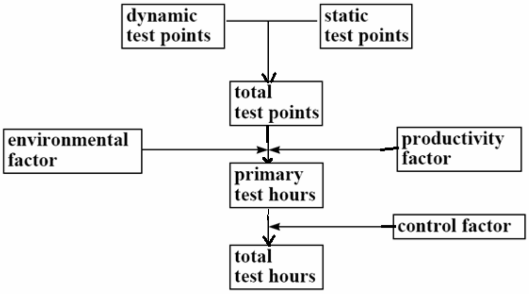


Fig. 1. TPA Technique

As shown in Fig.1. first Dynamic Test Points and Static Test Points are calculated. Next, total test points are obtained by adding dynamic and static test points. Then Primary Test Hours are calculated using Total Test Points, Productivity Factor and Environmental Factor. Finally, Total Test Hours are calculated using primary test hours and planning and control allowance.

Though both UCP and TPA give good results, there is still a lot of gap between the actual and predicted values. The results from these methods can be further refined. This is an attempt to optimize the results of these two techniques using PSO.

6 Proposed Strategy

This work has been carried out to optimize test effort estimation in order to reduce the difference between actual and predicted effort. To achieve this aim, Particle Swarm

Optimization is applied on both the methods, TPA and Use Case, in the following manner as explained in Fig.2.

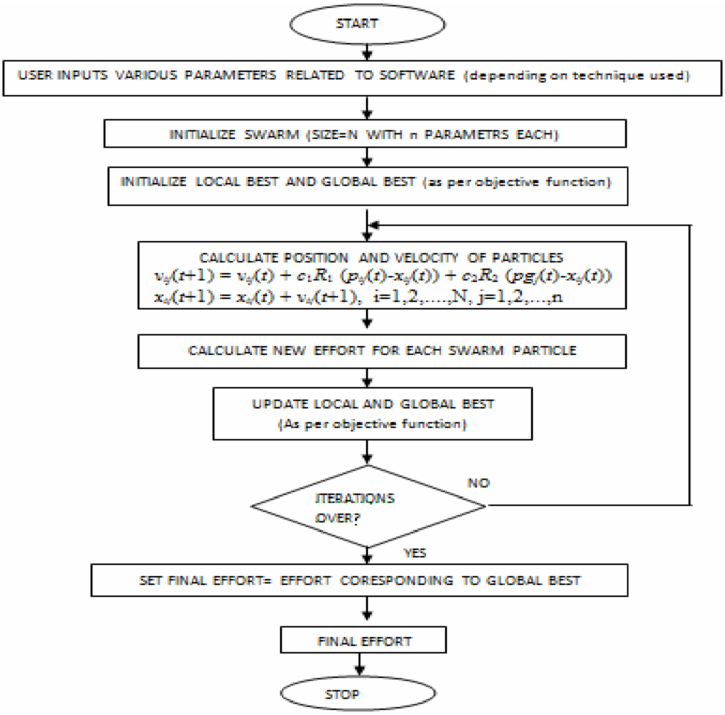

START
USER INPUTS VARIOUS PARAMETERS RELATED TO SOFTWARE (depending on technique used)
INITIALIZE SWARM (SIZE=N WITH n PARAMETRS EACH)
INITIALIZE LOCAL BEST AND GLOBAL BEST (as per objective function)
CALCULATE POSITION AND VELOCITY OF PARTICLES
$v_{ij}(t+1) = v_{ij}(t) + c_1R_1\ (p_{ij}(t)-x_{ij}(t)) + c_2R_2\ (pg_j(t)-x_{ij}(t))$
$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1),\ i=1,2,\dots,N,\ j=1,2,\dots,n$
CALCULATE NEW EFFORT FOR EACH SWARM PARTICLE
UPDATE LOCAL AND GLOBAL BEST
(As per objective function)
ITERATIONS OVER?
NO
YES
SET FINAL EFFORT= EFFORT CORESPONDING TO GLOBAL BEST
FINAL EFFORT
STOP

Fig. 2. Solution Model

The solution model flowchart explains the strategy followed in the optimization process. The steps included are initialization of swarm, application of PSO algorithm and analysis of the obtained output. The steps are explained below in detail:

1. A search space of 10 particles (where particles represent possible solutions in the search space) is considered. The study was conducted in different swarm sizes, and with a swarm size of 10 particles it was found through simulation that more optimal solutions were derived. Hence a search space of 10 particles was taken into consideration. Here particles are total effort calculated by randomly initialized weights for different ratings in that particular method (UCP or TPA).

2. Each particle is represented as a function of various parameters of the respective technique used, like UCP and TPA.

In use case 16 parameters are present (3 for actor weights, 4 for use cases and 9 for technical factors) [6] In Test Point Analysis 24 parameters related to Function Dependency, Environmental Factors, Static and Dynamic Quality characteristics, Team size and Planning and Control tools are taken into account [7].

3. These particles will move in search space to achieve global best according to PSO equations [11]:

Velocity of the particle is updated by the equation:

$$v_{ij}(t+1) = \omega v_{ij}(t) + c_1 R_1 \left(p_{ij}(t) - x_{ij}(t)\right) + c_2 R_2 \left(pg_j(t) - x_{ij}(t)\right)$$

Position of the particle is updated by the equation:

$$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1).$$

where v_{ij} and x_{ij} are vectors representing the velocity and position of the particle respectively. Here, with respect to this problem, position is the weights/values assigned to the parameters and velocity is the rate of change of these values. Suppose, n is weight assigned to simple actor in use case analysis. Then, n is the present x_{ij}. Then after iteration, if v_{ij}=2, the weight becomes n+2, that is, new x_{ij}.

c_1, c_2 are constants representing cognitive and social parameters, respectively. The combination of these two parameters determines the property of convergence of the algorithm. The portion of the adjustment to the velocity influenced by the local best is considered to be the cognitive component, and the portion influenced by the global is the Social component.

ω is the inertia factor, introduced to give best PSO performance by linearly decreasing its value as it moves towards optimized results[15].

R_1, R_2 are random numbers between 0 and 1. $p_{ij}(t)$ is the local best, which is the best value attained by an individual particle till the time t. $pg_j(t)$ is the global best, which is the best of all values obtained in the entire swarm till the time t. In each iteration, the position and the velocity of the particle as well as the local best and global best are updated according to the above equations.

According to these equations, the estimated effort value will move in the search space trying to reach the optimal value in each iteration. These equations are applied iteratively until particles converge to optimal solution, that is, where effort is closer to actual value. In every iteration, each particle has its local best (best position achieved by that particle in its lifetime). Best position from all the local best is the global best and movement of particles is dependent on both local and global best.

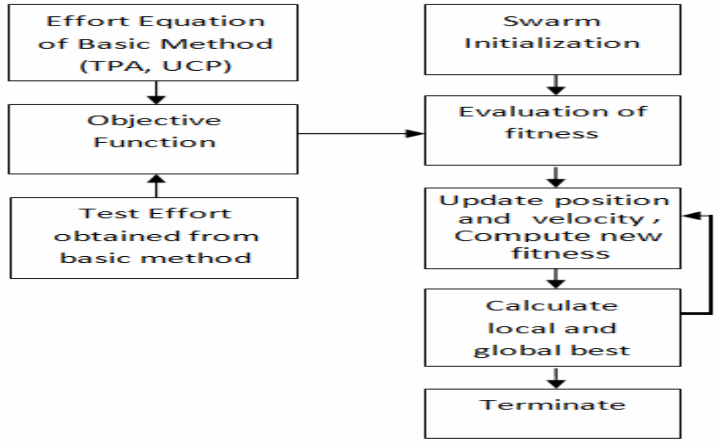

Effort Equation of Basic Method (TPA, UCP)
Objective Function
Test Effort obtained from basic method
Swarm Initialization
Evaluation of fitness
Update position and velocity, Compute new fitness
Calculate local and global best
Terminate

Fig. 3. Architecture

4. The objective is to get optimum effort closer to the one obtained by applying basic method. The objective function is selected based on what is to be optimized. Here since optimization of test effort estimation is considered, the basic equations of estimating test effort (in UCP and TPA) are taken as the objective function. So, objective function uses basic equation for effort estimation in that particular technique (TPA or Use Case). This objective function determines local and global best.

5. As an output, optimized effort is obtained.

The overall architecture of this algorithm is shown in Fig 3, which shows that optimal results can be obtained after applying PSO on swarm. It shows the objective function derived from TPA and UCP equation and effort obtained from these methods is used to evaluate fitness of the swarm particles. Then as PSO works iteratively and results converge to an optimal solution.

7 Case Study

Example 1. Use Case Points Analysis

The following example is taken from the paper "Test Effort Estimation Using Use Case Points" by Suresh Nageswaran [6]. The example considered here is that of a web-based software. The use cases as well as actors were identified from the requirements to calculate Unadjusted Actor Weights(UAW).

The details are given below as per Table 1, 2 and 3.

UUCP=UAW+UUCW=64+55=119
Adjusted UCP (AUCP) = UUCP*(.65+.01*TEF)
=119*(.65+.01*87) =180.88
Final Effort=AUCP*Conversion Factor
=180.88*13=2351.4

Taking 15% for project complexity and 10% for management, the results obtained were: Effort=2939.304 man hours =367 man days.

Table 1. Calculation of Unadjusted Actor Weight (UAW)

Actor	No. of use cases	Factor	UAW
Simple	0	1	0
Average	32	2	64
Complex	0	3	0
Total UAW			64

Table 2. Unadjusted Use Case Weights (UUCW)

Use Case	Type	Factor	UUCW
Simple	2	5	10
Average	1	10	10
Complex	1	15	15
Very Complex	1	20	20
Total UUCW			55

Table 3. Technical Factors

Factor	Assigned value	Weight	Extended Value
T1	5	3	15
T2	5	5	25
T3	2	1	2
T4	3	1	3
T5	3	2	6
T6	4	4	16
T7	2	1	2
T8	4	2	8
T9	5	2	10
Total			87

Actual effort = 390 man days.
Magnitude of Relative Error = | Actual Effort-Predicted effort |/Actual Effort * 100
=|390-367|/390*100 = 5.8%

PSO on UCP is applied by:

- Initializing the 10 particles, with 16 parameter values corresponding to actors, use cases, and technical factors in Use Case Point Analysis in the search space randomly. (In this case, an initial swarm of 10 particles was considered, as discussed earlier in proposed strategy. It may vary for different problems.)

The range of values for the parameters were bound in the intervals shown in Table 4. The initial values for the particles were taken randomly within this range.(Table 5)

Table 4. Range Taken For Weights Of Parameters

Parameters	Range
Actor	1-3
Use Cases	5-20
Technical Factors	1-5

Table 5. Initial Values of Some Particles

Parameter	Particle 1	Particle 2	Particle 3
Actor-simple	1	1	2
Actor-average	2	2	1
Actor-complex	3	2	3
Use Case-simple	5	6	10
Use Case-average	10	9	5
Use Case-complex	15	14	7
Use Case-very complex	20	19	18
T1	3	1	3
T2	5	2	2
T3	1	3	4
T4	1	4	1
T5	2	5	5
T6	4	3	3
T7	1	4	4
T8	2	2	1
T9	2	3	1

- Initial effort of all 10 particles is calculated by taking the effort equation as the objective function. This will be their initial local bests, and from these local bests, the best one is taken to be the initial global best.

For particles, initial effort is obtained as: 3633.5, 2359.5, 3633.5, 3077.62 and so on

- In each iteration, all the 16 parameters for each particle are updated using the position and velocity equations for PSO. And then, objective function is again computed to find the local best and global best. Thus in each iterations, local and global best get updated, and the swarm moves towards optimality.

In this problem 50 iterations are considered because results converge to an optimal solution within this limit. More iterations may be required, depending on the problem.

- Objective function is taken in such a way so as to maximize the effort but keeping it closer to the effort obtained by applying the basic method (UCP). So basic effort estimation equation of use case analysis is used.

- Finally after all iterations, results converge to 3032 man hours as total effort (including 25% for management and project complexity)= 379 man days

Magnitude of Relative Error = | Actual Effort-Predicted effort |/Actual Effort * 100

=|390-379|/390*100 = 2.8%

Example 2. Test Point Analysis
A project on "Random Number Generation Using Cellular Automata and LFSR.

Coupling" developed in C++ is taken for analysis. It has 11 modules and was tested in

2 months. Following are the parameters related to the project:

Total modules=11

Productivity Factor=0.7 (productivity factor is the time it takes for a tester to perform an activity, which depends on his skills and experience and is specific to an organization)

Now the well-defined variables under TPA are used for calculations:

$FDC_w = ((FI_w + UIN_w + I + C) / 20) X U$

Table 6. Calculation of FDCw

Module	FI	UIN	I	C	U	FDC_w
1	low	Low	low	Med	1	0.65
2	low	Med	low	Med	1	0.75
3	low	Med	low	Low	1	0.6
4	low	Med	Low	Low	1	0.6
5	med	Low	Med	Low	1	0.75
6	high	High	Low	Low	1	1.45
7	high	High	Low	Low	1	1.45
8	high	Med	High	High	1	1.8
9	med	Med	Med	Low	1	0.85
10	med	Med	Med	Low	1	0.85
11	high	Low	High	High	1	1.7

$QC_{dw} = \sum$(rating of QC/4) X weight factor of QC.

Table 7. Calculation of QC_{dw}

Module	Suitability	Security	Usability	Efficiency	QC_{dw}
1	Not imp	Rel unimp	Rel unimp	Rel unimp	0.187
2	Not imp	Rel unimp	Rel unimp	Rel unimp	0.187
3	Not imp	Rel unimp	Rel unimp	Rel unimp	0.187
4	Not imp	Rel unimp	Rel unimp	Rel unimp	0.187
5	Not imp	Rel unimp	Rel unimp	Rel unimp	0.187
6	Not imp	Rel unimp	Rel unimp	Rel unimp	0.187
7	Not imp	Rel unimp	Rel unimp	Rel unimp	0.187
8	Not imp	Rel unimp	Rel unimp	Rel unimp	0.187
9	Not imp	Rel unimp	Rel unimp	Rel unimp	0.187
10	Not imp	Rel unimp	Rel unimp	Rel unimp	0.187
11	Not imp	Rel unimp	Rel unimp	Rel unimp	0.187

Calculation of DTP:

Number of Dynamic Test Points (DTP) = FP X FDC_w X QC_{dw}

Table 8. Calculation of DTP

Module	FP	FDC_w	QC_{dw}	DTP
1	11.38	0.65	0.187	1.388
2	9.969	0.75	0.187	1.402
3	17.44	0.6	0.187	1.962
4	27.76	0.6	0.187	3.124
5	25.97	0.75	0.187	3.653
6	5.45	1.45	0.187	1.4827
7	5.54	1.45	0.187	1.4827
8	18.995	1.8	0.187	6.411
9	23.648	0.85	0.187	3.769
10	8.589	0.85	0.187	1.369
11	19.187	1.7	0.187	6.116
Total				32.1594

Calculation of STP (Static Test Points):

STP = FP X $\sum QC_{sw}$ / 500 = 16*0/500*23.64 = 0

Calculation of TTP (Total Test Points):

TTP = DTP+STP = 32.1594+0 = 32.1594

Calculation of Environment Factor:

Test Tools: No test tools used

Development Testing: Test plan Available

Test Basis: Documentation not developed according to standards

Development Environment: using old platform

Test Environment: Test platform is new

Test Ware: No Test Ware available

Environmental Factors = weights of (test tools+development testing+test basis+ development environment+testing environment+testware)/21=4+4+12+8+12/21 = 1.4

Calculation of Primary Test Hours:

PTH = TTP X Productivity Factor X Environmental Factor=32.1594*0.7*1.43= 32.19

Calculation of Total Test Hours:

Team size = 4

Planning and control tools: not available

Planning and Control allowance = weights of (team size + Planning and Control tools) X PTH = (3+6)*32.19 = 289.71

Total test hours TTH = PTH + Planning and Control Allowance
= 289.71+32.19 = 321.9 hrs = 40.2375 days

Actual testing time = 55 days

Magnitude of Relative Error = | Actual Effort-Predicted effort |/Actual Effort * 100
= (55-40.2375)/55*100 = 26.84%

Now PSO is applied on this technique by:

- Initializing swarm size of 10 particles, as discussed in proposed strategy.
- Random initial values to all parameters of TPA for the 10 particles are taken and total test hours are calculated, which is the initial local best. Some of the local best for particles were 485.78, 356.17, 474.357 , etc
- Initializing global best from this initial set of local best values.
- Applying iterations in which position and velocity of particles are updated each time, using the equations of PSO, resulting in new local and global best.
- Finally, after 50 iterations, results converge to total test hours=472 hour= 59 days (with 8 hours/day)

Magnitude of Relative Error= | Actual Effort-Predicted effort |/Actual Effort * 100
=|55-59|/55*100 = 7.2%

8 Comparison with Existing Approach

From the above case studies, the following results were obtained and list is in Table 9, Fig. 4 and 5:

Table 9. Comparative Results

Technique	MRE(without PSO)	MRE(after applying PSO)
Use Case Points (Example1)	5.8%	2.8%
Test Point Analysis (Example2)	26.84%	7.2%

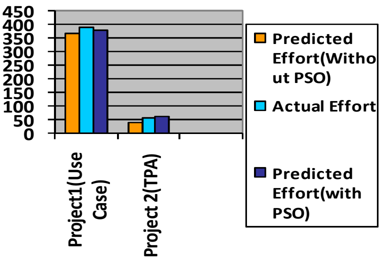


Fig. 4. Bar Graph for Comparing Effort

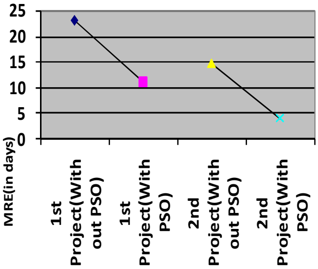


Fig. 5. Graph for Comparing Reduction in MRE

The results show that after applying PSO, the magnitude of relative error decreases, thereby making the predicted results closer to the actual ones.

9 Conclusion and Future Work

In this project, Particle Swarm Optimization (PSO) was applied on two techniques: use case points (UCP) and test point analysis (TPA) and the results led us to the conclusion that test effort estimation can be optimized by applying PSO. The results were compared with those obtained from existing methods, and were found to be closer to the actual effort.

The work done can be extended to optimize other effort estimation techniques. In this paper, only optimization of test effort estimation has been considered. PSO optimization can also be applied for estimating effort of software development. Possibility of further optimization can be explored by applying other variants of PSO.

References

1. Sommerville, I.: Software Engineering. Pearson Edition, India (2009)
2. Jalote, P.: An Integrated Approach to Software Engineering. Springer Science+Business Media, Inc., New York (2005)
3. Clarke, J., et al.: The Application of Metaheuristic Search Techniques to Problems in Software Engineering. SEMINAL-TR-01-2000 (2000)
4. Ferrucci, F., Gravino, C., Oliveto, R., Sarro, F.: Using tabu search to estimate software development effort. In: Abran, A., Braungarten, R., Dumke, R.R., Cuadrado-Gallego, J.J., Brunekreef, J. (eds.) IWSM 2009. LNCS, vol. 5891, pp. 307–320. Springer, Heidelberg (2009)
5. Sandhu, P.S., et al.: Software Effort Estimation Using Soft Computing Techniques. In: World Academy of Science, Engineering and Technology, pp. 488–491 (2008)
6. Nageswaran, S.: Test Effort Estimation Using Use Case Points. In: 14th International Software/Internet Quality Week, San Francisco (2001)
7. van Veenendaal, E.P.W.M., Dekkers, T.: Test Point Analysis: A Method for Test Estimation. In: ESCOM 1999 (1999)
8. Kaur, J., et al.: Neural Network-A Novel Technique for Software Effort Estimation. International Journal of Computer Theory and Engineering 46, 485–487 (2008)
9. Martin, C.L., et al.: Software Development Effort Estimation Using Fuzzy Logic: A Case Study. In: 6th Mexican International Conference on Computer Science (ENC 2005), Mexico, pp. 113–120 (2005)
10. The PSO website, `http://www.swarmintelligence.org/`
11. Parsopoulos, K.E., Vrahatis, M.N.: Particle Swarm Optimization and Intelligence: Advances and Applications. Information Science Reference, New York (2010)
12. Clemmons, R.K.: Project Estimation With Use Case Points. CrossTalk – The Journal of Defense Software Engineering, 18–22 (2006)
13. Chauhan, N.: Software Testing – Principles and Practices, pp. 335–340. Oxford University Press, New Delhi (2010)
14. Kennedy, J., Eberhart, R.C.: Particle swarm optimization. In: IEEE Conference on Neural Networks, Piscataway, NJ, pp. 1942–1948 (1995)
15. Shi, Y., Eberhart, R.C.: A modified particle swarm optimizer. In: IEEE International Conference on Evolutionary Computation, pp. 69–73 (1998)

Answering Cross-Source Keyword Queries over Deep Web Data Sources

Fan Wang and Gagan Agrawal

Department of Computer Science and Engineering
Ohio State University, Columbus OH 43210
{wangfa,agrawal}@cse.ohio-state.edu

Abstract. A popular trend in data dissemination involves online data sources that are hidden behind query forms, which are part of the *deep web*. Extracting information across multiple deep web sources in a domain is challenging, but increasingly crucial in many areas. Keyword search, a popular information discovery method, has been studied extensively on the surface web and relational databases. Keyword-based queries can provide a powerful yet intuitive means for accessing data from the deep web as well. However, this involves many challenges. For example, deep web data is hidden behind query interfaces, deep web data sources often contain redundant and/or incomplete data, and there is often inter-dependence among data sources. Thus, it is very hard to automatically execute cross-source queries.

This paper focuses on answering *cross-source* queries over deep web data sources. In our approach, we model a list of deep web data sources using a *graph* to capture the dependencies among them, and we consider the problem of answering cross-source queries over these deep web data sources as a graph search problem. We have developed a bidirectional query planning algorithm to generate query plans for two types of cross-source queries, which are *entity-attributes* queries and *entity-entity relationship* queries.

1 Introduction

A popular trend in data dissemination involves online data sources that are hidden behind query forms, which are part of the *deep web*. The deep web data is hidden behind query forms, and users could only access the data using their input interfaces [4]. Hundreds of large, complex, and in many cases, related and/or overlapping, deep web data sources have become available. Answering a query can require the data from multiple correlated data sources. We refer to such queries as *cross-source* queries. Currently, to answer a cross-source query, the user has to first *manually identify* data sources that contain the relevant data. Second, they need to *manually submit* online queries to numerous query forms, *keep track of* the results, and finally combine them together. This is a tedious and error-prone process. It is desirable to have a tool that can *automatically* answer such queries over the deep web sources.

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 475–490, 2011.

This paper focuses on addressing the problem of supporting *cross-source keyword queries* on a set of integrated deep web data sources. Keyword search has been a popular information discovery method over the surface web. In recent years, it has been applied on relational and graph datasets as well [17,16,3,24,30]. Our work is driven by gaining popularity of this query mechanism, but considers such queries over deep web data sources. In this paper, we consider the following scenario. We have multiple deep web sources within a particular domain, each of which has one or more query forms. A query submitted using these forms triggers a query on the back-end databases. We want to support two types of cross-source queries: 1) *Entity-Attributes Search*, where a user may submit an entity name and one or more attributes in a domain, and would like to search based on attributes of interest for the entity, and 2) *Entity-Entity Relationship Search*, where a user submits multiple entity names from a domain, and wants to know possible relationships among these names. A formal presentation of our query model is described in Section 2.1. Most complex queries can be formulated as a combination of multiple sub-queries of the above two types. As a result, we are focusing on these two types of queries in this paper. In the following, we use two examples to further illustrate these two types of queries.

Motivating Example 1: Entity-Attributes Query: We have an *entity-attribute query*, Q1={ERCC6,SNPID,"ORTH BLAST",HGNCID}, where `ERCC6` is the *entity name* representing a gene, and `SNPID`, `ORTH BLAST` and `HGNCID` are three *attributes* of which the user wants to obtain the value. Query Q1 has the following intent: given a gene name `ERCC6`, we want to find the `SNPIDs`, the `BLAST` results and the corresponding `HGNCID` of gene `ERCC6`. The query plan of answering $Q1$ is shown in Figure 1a. We could observe that Q1 is answered using this query plan as follows. To find the `SNPIDs` of `ERCC6`, we need to use gene name `ERCC6` as input to query on `dbSNP` data source to find `SNPIDs`, and we could also obtain the amino acid position information from `dbSNP`. To find the `BLAST` information, we need to take the following three steps: 1) use `ERCC6` as input to query on `Gene` data source to obtain the proteins of `ERCC6` in human species and other orthologous species; 2) use the proteins obtained from `Gene` data source as input to query on `Protein` data source to find the protein sequences; 3) use the protein sequences from step 2 and the amino acid position information obtained from `dbSNP` as input to query on `Entrez BLAST` data source to obtain the `ORTH BLAST` result. To find the `HGNCID` of `ERCC6`, we use `ERCC6` as input to query on `HGNC` data source to obtain `HGNCID`.

From this example, we see that a query path, which is determined by the *dependence* among the data sources, could guide the search.

Motivating Example 2: Entity-Entity Relationship Search: We have an *entity-entity relationship* query, Q2={MSMB,RET}. This query wants to find the relationship between two genes `MSMB` and `RET`. Our system needs to determine these two genes can be connected by a chromosome using database `Gene` or (and) `SNP500`. i.e., it turns out that the two genes are located in the same chromosome `10q11.2`. Two valid query plans of $Q2$ are shown in Figure 1(b) and Figure 1(c).

The query plan in Figure 1(b) shows that we use the two gene names MSMB and RET as input to query on Gene data source respectively, then the two sets of result from Gene data source tell us that both of the genes locate in the same chromosome. The query plan in Figure 1(c) shows that we use MSMB as input to query on Gene data source to obtain the chromosome information about the gene MSMB, then we use the chromosome as input to query on SNP500 data source. The result from SNP500 tells us that gene RET also locates on this chromosome.

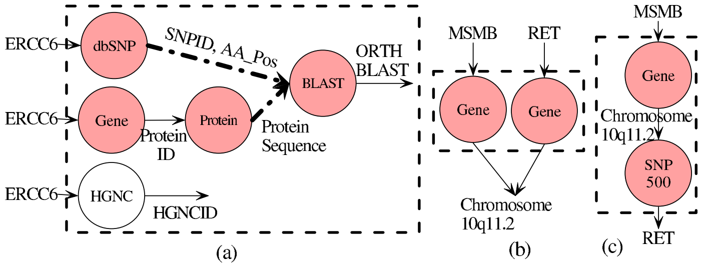


Fig. 1. Motivating Example: (a) Query Plan for Q1; (b)(c) Query Plan for Q2

Many challenges arise in answering cross-source queries on deep web data sources. First, unlike keyword search on relational databases [1,17] or graphs datasets [15,32], we only know query schemas, i.e., input and output attributes, of deep web sources, whereas the real data content is hidden in the backend web servers. Although surfacing deep web data has been used in systems like [26], this method will lead to a data consistency problem if deep web data is continuously updated. Second, answering a cross-source query involves multiple *inter-dependent* data sources. Data source inter-dependency commonly exists among the data sources. For example, in Figure 1a, BLAST data source depends on the output from both dbSNP and Protein. As a result, a valid query plan must respect data source dependencies. Third, unlike the existing work on keyword search [1,17,15,32,29,24], where the query plan must be a *connected tree*, a query plan in our scenario could be a graph with *disconnected components*. An example of this is shown in Figure 1a. Here, the shaded part (including data sources dbSNP, Gene, Protein and BLAST) and the unshaded part (including data source HGNC) are two disconnected components. Thus, the techniques related to the *Steiner Tree Algorithm*, which are used in the existing work, cannot be applied to our problem. Finally, there often is much data redundancy across deep web data sources [4]. This implies that the same data can be obtained from multiple data sources. For example, given a gene name, as in the example query Q1, three data sources can be used to obtain the SNPID information, which are dbSNP, SNP500 and Alfred. The *data redundancy* considered in this paper is the *partial redundancy* across *different* data sources, i.e., the data of some, but likely not all, of the attributes available from one sources can also be obtained from another data source. As a result, data source selection and ranking strategy are essential for our system.

The rest of the paper is organized as follows. In Section 2, we formulate the query planning problem. A novel bidirectional query planning algorithm is detailed in Section 3. Our algorithm is evaluated in Section 4. We compare our work with related efforts in Section 5 and conclude in Section 6.

2 Query Planning Problem Formulation

In this section, we give an overview of the query planning problem for the two types of queries we are considering.

2.1 Overview

A query Q is formally denoted as $Q = \{e_1, \ldots, e_m, a_1, \ldots, a_k\}$. Here, each a_i is an *attribute keyword* and each e_i is an *entity keyword.* If there are *attribute keywords* in Q, i.e., $k > 0$, Q is called an *entity-attribute query.* We could consider an *entity-attributes query* as a *selection-projection-join* SQL query, where each *attribute keywords* corresponds to the attributes in the selection clause of the a SQL query, and each *entity keyword* corresponds to entities or rows returned in the where clause of an SQL query. As a result, the *entity keyword* helps initiate answering of the query. The intention of a *entity-attribute* query is to find the values of the *attribute keywords* of the *entity.* For *entity-entity relationship* queries, all search terms are *entity keywords*, i.e., there are no *attribute keyword* and $k = 0$. The intention of a *entity-entity relationship* query is to find the *common attribute terms* of the given *entities.*

2.2 Capturing Data Source Dependencies

Before formally stating the query planning problem, we first define the *data source dependency graph.* We consider deep web data sources are connected by the inter-dependence between them and form a *dependency graph.* Graph nodes represent data sources and edges represent the inter-dependency relation between pairs of data sources. If there is a dependency edge pointing from data source node u to data source node v, it shows that the output from data source u would be used as the input values for data source v. we call u is the *parent* of v, and v is the *dependent* of u. In Figure 2, we show the inter-dependency relation between 5 data sources.

Some data source dependencies are multi-source, i.e. the input of a data source depends on the output from multiple data sources, as a result, some data source node has *composite parents.* For example, in Figure 2, two highlighted edges linked by an *arc* shows that `dbSNP` and `Entrez protein` form a *composite-parent* for `BLAST`, which means that to be able to query `BLAST`, one needs to query **both** `dbSNP` and `Entrez Protein` first.

Somewhat similar to our work, Davulcu *et al* [8] proposed a *navigation map* to capture the link structure between web pages generated by deep web data sources. The main difference of our dependence model is that we model the *inter data-source* dependencies, but they model the *intra-data source* web page dependencies.

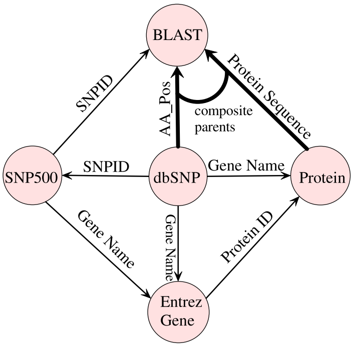


Fig. 2. Dependence Relations between Five Data Sources

2.3 Problem Formulation and Cost Model

Given a query Q and a data source dependency graph DG introduced as above, the query planning problem is formulated as follows. *We want to find a* subgraph *SubG from DG, such that all search terms in Q are covered by SubG*. Here, *covering* a search term can have different meaning depending upon the type of the term. For an attribute term a_i, we want to obtain its value from certain data sources, so a_i is mapped to data source(s) with an *output attribute* that matches it. For an entity term e_i, there are two considerations. First, e_i could help initiate answering of a query. Thus, we map e_i to data sources with an *input attribute* that covers the corresponding type of the entity. For example, in $Q1$ and $Q2$ in Section 1, using a domain dictionary we can know that `ERCC6` and `MSMB` are gene names. As a result, we can map them to the data sources whose input attributes contain the attribute `Gene Name`. The second consideration is as follows. In an *entity-entity relation* query, we can consider e_i to be the value of an output attribute. As shown in Figure 1c, the gene name `RET` is connected with `MSMB` because `RET` is the value of an output attribute of the data source `SNP500`. Similarly, `MSMB` is connected with `SNP500` through the chromosome `10q11.2`. From this consideration, we map e_i to data sources with an *output attribute* that covers the corresponding type of the entity.

Clearly, there may be multiple subgraphs of the dependency graph that satisfy the conditions for a valid query plan we have stated above. Among these query plans, we want to identify the query plan with the *least execution* time, and the one which is likely to give the *highest quality* of results. In our approach, we combine these two considerations into a single *cost model.* Though the details of these cost models are available from an associated technical report [34], the basic ideas are as follows. For every data source D included in a candidate query plan, we associate an *access cost*, and a *quality cost.* The access cost includes the response time of D, which is obtained by *off-line data source profiling*, i.e., by issuing pilot queries on a data source, and obtaining the average response time. In the future, our work can incorporate more advanced access cost estimation methods [12,6]. The second component, *quality cost*, captures the fact that different data sources in a domain may contain similar data, but with different data quality or a higher trust factor [4]. For example, a data source may be likely to return more data elements in response to a given query, or alternatively, data sources maintained by large and reputed institutions are likely to be trusted more by the users. To capture this together with the access cost, we associate a higher cost with data sources that may have a lower quality or may otherwise not be preferred by the users. This information is clearly domain specific and our system assumes that this information is captured and is available.

Now, our query planning problem becomes: *We want to find a* subgraph *SubG from DG, such that all search terms in Q are covered by SubG, and furthermore, among all subgraphs meeting these requirements, SubG has the lowest sum of access and quality costs.*

Given the query planing problem formulated as above, in an associated technical report [34], we have established the following result.

Lemma 1. *The query planning problem is NP-hard.*

Proof. Lemma 1 can be proved by reducing the standard set cover problem to our query planning problem. The detailed proof is in an associated technical report [34]. □

3 Bidirectional Planning Algorithm

In this section, we propose a heuristic bidirectional planning algorithm to find a minimal cost query plan for a given query.

3.1 Algorithm Overview

At the beginning of the algorithm, we map the search keywords onto the data sources which *cover* them according to mapping method we introduced in Section 2.3. We define the data source nodes that cover *attribute terms* as the *target nodes* and the data source nodes that cover *entity terms* as the *starting nodes*. A query plan ultimately connects a subset of target nodes with a subset of starting nodes, such that all search terms are covered. We explore the query plan in a bidirectional manner. We perform backward exploration from the target nodes to connect them with starting nodes. To accelerate this process, we also do forward exploration from the starting nodes. In this way, the bidirectional exploration can meet *mid-way*. To find the query plan with cost as least as possible, we apply the following two heuristics.
Heuristic 1: Always try to add the data source with the least cost into the query plan.
Heuristic 2: If multiple data sources need to be connected to form a query plan, connecting them via the shortest path.

3.2 Bidirectional Exploration

Initially we add all *starting nodes* to a forward exploration queue, and all *target nodes* to a backward exploration queue. Initially, if a node covers a search term, the distance of reaching this search term from the node is 0, otherwise the distance is infinity. Then, the algorithm tries to find a sub-graph (with minimal cost) to connect the target node set with the starting node set. At each iteration of the sub-graph exploration, the algorithm always selects the node with the *least cost*, CN, from the two queues, using the cost model described above (applying the first heuristic). If CN belongs to the forward queue, all out-going neighbors of CN will be explored using *forward exploration.* If CN belongs to the backward queue, all in-coming parents of CN will be explored using *backward exploration.* During the backward or forward exploration, since new nodes are explored, the distance from a data source to search terms could be updated. This is done using *edge exploration.* Finally, when every search term can be reached from a starting node with finite distance, a query plan is found and this query plan is a graph with disconnected components, each of which is rooted at the corresponding starting node. The details of forward exploration, backward exploration and edge exploration are introduced as follows.

Forward and Backward Exploration: Forward exploration explores the edge e between CN (predecessor) and one of CN's descendants. Backward exploration explores the edge e between CN (descendant) and one of CN's predecessor. Regardless of the direction of the exploration, for an edge e, we denote the predecessor node as u and the descendant node as v. Since some data source dependencies in our dependency graph are multi-source, when e is explored, we need to consider two cases. In the first case, the predecessor u is a single node. We just directly perform an *edge exploration* on e. For example, in Figure 2, suppose we are now on node `dbSNP`, and we want to do a forward exploration to `Entrez Gene`. Since `dbSNP` is the single predecessor of `Entrez Gene`, the exploration can be done directly. All search terms covered by `Entrez Gene` now can be reached from `dbSNP` via `Entrez Gene`, so that the distance from `dbSNP` to the search terms originally covered by `Entrez Gene` should be updated using the *edge exploration* function. In the second case, u is involved in a composite parent node with respect to v. In this case, all nodes in the composite parent node should be explored previously so that the edge between the composite parent node and v can be explored. This is because the accessibility of the dependent node v depends on the accessibility of all its predecessors in the composite parent node. As a result, if any node in composite parent node is not accessible currently, we cannot access v. In order to explore edge e, we need all the unexplored nodes in the composite parent node to be explored. From this point of view, these unexplored nodes become *target nodes* and, therefore, are added to the backward queue. For example, in Figure 2, suppose we are on `dbSNP`, and we want to do a forward exploration to `BLAST`. Since `dbSNP` and `Protein` are composite parents for `BLAST`, we need both of `dbSNP` and `Protein` been explored so as to explore `BLAST`. If `Protein` has not been explored, we can consider `Protein` as a new *target node* and add it to the backward queue.

The backward exploration is executed in similar way as the forward exploration. After a backward exploration, suppose from v to u, we add node v to the forward queue to explore the frontier of v.

Edge Exploration: To build the sub-graph as the final query plan, paths (sequence of graph edges) must be explored to connect target nodes with starting nodes. The goal of edge exploration is to build such paths and update the path distance information, i.e. the distance from starting nodes to search terms, whenever a new node is explored. Here, we apply the second heuristic which is that we always connect pair of nodes through the shortest path. An edge with a shorter distance is preferred and the shorter distance is propagated to the starting nodes. Suppose the two end nodes of an edge e are u (predecessor) and v (descendant). The edge exploration is performed in two steps. The first step is the *local distance update* on the predecessor node u, and the second step is *distance propagation* to u's ancestors. If u is a single node, we first update the shortest distance from u to any search term could be reached via v (local distance update step). Then, we propagate the updated distance to u's ancestors (distance propagation step). This is the standard edge relaxation in Dijkstra algorithm.

If u is a composite parent node, i.e., u contains multiple nodes which form a composite parent node for v, we use a different strategy. We first locally update the shortest distance from each node in the composite parent node u to any search term could be reached via v (local distance update step). For the distance propagation step, we differentiate u's ancestors into two types. The first type of u's ancestors are the *Shared Ancestors* or SA, which are the common ancestors shared by more than one node in the composite parent node u. The second type of ancestors are the *Unshared Ancestors* or UA, which are the ancestors for a single node in u only. An important feature for a shared ancestor, sa, is that the shortest distance from sa to a search term could be reached via u depends on the *longest* path among all the paths connecting sa to each node u_i in u. As a result, we perform distance propagation on *unshared ancestors* as normal. For *shared ancestors*, we first compute a batch of shortest distances from sa to u using every node u_i, which is a descendent of sa, in u. Then, we propagate the *longest one* among the batch of shortest distances to sa. Taking Figure 2 as an example, suppose all edges have distance of 1 except for the edge between `BLAST` and `dbSNP` having a distance of 2. The search term `ORTH BLAST` is covered by `BLAST` and we want to update the distance of reaching the search term `ORTH BLAST` from data source `Entrez Gene`, i.e., obtain the distance from node `Entrez Gene` to node `BLAST`. Because `dbSNP` and `Protein` form a composite parent node for `BLAST`, although the distance from `BLAST` to `Entrez Gene` via `Protein` is 2, the final distance should be 3 which is through `dbSNP`.

This algorithm may seem similar to the Steiner tree algorithm used in the context of keyword search on relational databases (Kacholia *et al.* [19]). Given a set of nodes N, the Steiner tree algorithm finds a subgraph which achieves *pair-wised* connection for nodes in set N. However, the connection requirement between the *target nodes* and the *starting nodes* we have is different from the steiner tree problem. In our experiment, we compare our bidirectional query planning algorithm with the Steiner tree algorithm and show that for most queries, the query plans generated by our algorithm runs faster (more efficient) than the plans generated by the Steiner tree algorithm.

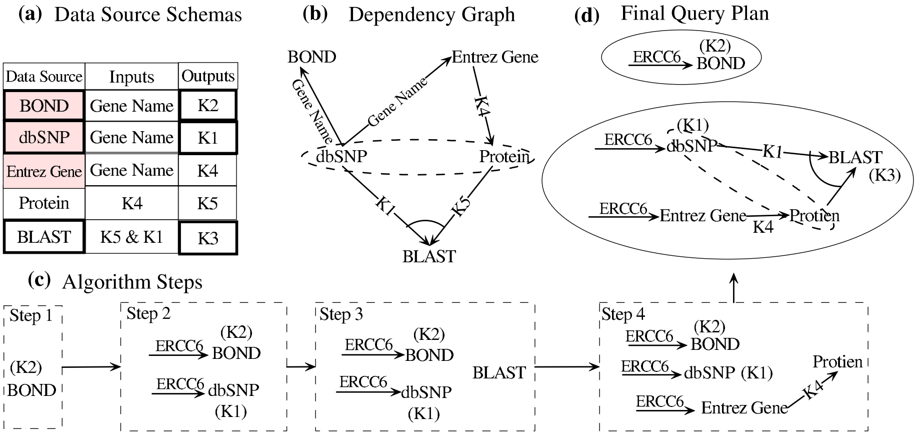
3.3 Query Planning Algorithm Running Example

We use a simple example to illustrate the main ideas in our algorithm. We focus on the general idea in our description, and the actual execution of the algorithm is more complex than what we discuss here.

We have a query Q={ERCC6,K1,K2,K3} and five data sources dbSNP, Entrez Gene, Protein, BOND and BLAST. Q is an *entity-attribute* query with the following intention: given a gene name ERCC6, we want to find the values of the attributes K1, K2 and K3 (actual names are replaced by K1, K2 and K3, to increase readability). The running of the query planning algorithm on this example is shown in Figure 3.

The algorithm first maps search terms to data sources as introduced in Section 2.3. The starting nodes (shaded in schema table in figure 3a) are BOND, dbSNP and Entrez Gene, because their input attributes match with Gene Name which is the type of the entity keyword ERCC6 in Q. The starting nodes are added to the forward exploration queue. The target nodes (with thick border in figure 3a) are BOND, dbSNP and BLAST, because their output attributes covers K1, K2 and K3. The target nodes are added to the backward exploration queue.

Now, we move to the exploration phase. Suppose BOND has the highest score, then it is selected and added to the plan (K2 covered) as shown in the step 1. We have a finite distance from a starting node, BOND, to K2. In the step 2, dbSNP



Data Source	Inputs	Outputs
BOND	Gene Name	K2
dbSNP	Gene Name	K1
Entrez Gene	Gene Name	K4
Protein	K4	K5
BLAST	K5 & K1	K3

Fig. 3. Example Illustrating the Algorithm on the Query {ERCC6, K1, K2, K3} (a) Data Source Query Schemas;(b) Dependency Relation Graph Model; (c) Bidirectional Planning Steps; (d) Final Query Plan

is selected as a highly ranked node and added to the query plan as shown in the step 2. Now, K1 and K2 are covered and reached by two starting nodes with finite distance. In the next step, BLAST is selected from the backward queue and we need to perform a backward exploration. The parent of BLAST is a hyper-node composed of dbSNP and Protein as highlighted by a dotted border circle in Figure 3b. We need to make sure that both dbSNP and Protein, are explored, otherwise this backward exploration can not be performed. Since Protein has not been explored, this exploration is skipped and Protein is added to the backward queue as a new target node (step 3). In step 4, Protein is selected from the backward queue to perform backward exploration to Entrez Gene. At the last step, since both dbSNP and Protein are explored, we do a forward exploration from Protein to BLAST (K3 covered) as shown in figure 3d. The distance information of K3 is also propagated to the staring nodes dbSNP and Entrez Gene. Now, all three terms are covered and we have finite distances from starting nodes to search terms. The final query plan is a hyper-graph with two disconnected components (figure 3d, two disconnected components highlighted by two solid border circles) and it is a sub-graph of the dependency graph in figure 3b.

4 Performance Evaluation

In this section, we described the experiments we conducted to evaluate the performance of our algorithm.

4.1 Experiment Setup

Our evaluation was done using 12 biological deep web databases we have integrated, which includes dbSNP[1], Entrez Gene[1], Protein[1], BLAST[1], SNP500[2], Seattle[3], SIFT[4], BIND[5], Human Protein[6], HGNC[7], Mouse SNP[8], and ALFRED[9]. The input and output schema of the data sources are extracted using a previously created wrapper. The dependency graph is constructed by analyzing the correspondence between the output and input attributes of different data sources. We created 20 queries for our evaluation. The queries we use for evaluation throughout this section are based on our collaboration with a biologist focusing on SNP-related studies [35]. All queries in our experiment have the same structure as the motivating examples in Section 1.

[1] http://www.ncbi.nlm.nih.gov/projects/SNP
[2] http://snp500cancer.nci.nih.gov/home_1.cfm
[3] http://pga.gs.washington.edu/
[4] http://blocks.fhcrc.org/sift/SIFT.html
[5] http://www.bind.ca
[6] www.hprd.org
[7] www.genenames.org
[8] http://mousesnp.roche.com/
[9] http://alfred.med.yale.edu/alfred/

4.2 Query Planning Algorithm Evaluation

In our evaluation, we compare up to three algorithms. The first is one of the bidirectional query planning algorithm we have developed in this paper. The second is an exhaustive search algorithm (OPT), which searches the entire sub-graph space exhaustively to find the minimal cost query plan, and has an exponential complexity. Finally, we also compare our algorithm against the Steiner tree algorithm.

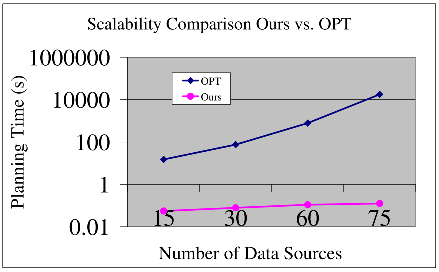
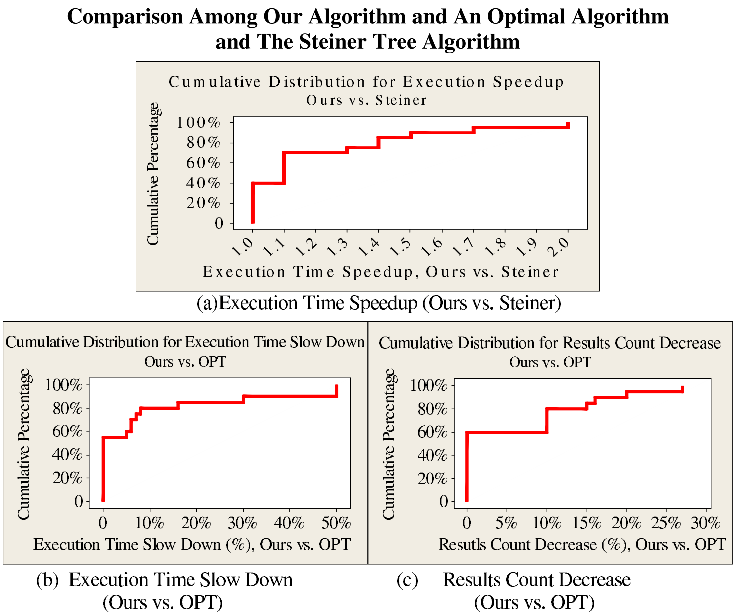


Fig. 4. Scalability Comparison between Our Algorithm and Exhaustive Algorithm

Scalability Evaluation. To compare the scalability between our algorithm and the exhaustive algorithm, we record the query planning time varying the number of data sources. Since our goal was to evaluate scalability, and because we did not need the parsers to obtain the query results, we experimented with up to 75 data sources. The results are shown in Figure 4. This figure is plotted with a logarithmic scale, where x-axis is the number of data sources and the y-axis is the query planning time in seconds. We observe that the exhaustive algorithm scales very poorly, and in fact, when the number of data sources is 75, it takes about 9 hours to find the query plan for a single query. In comparison, our algorithms have very good scalability, and take less than 1 second even with 75 data sources. This experiment shows that the exhaustive algorithm is not practical for real systems that integrate 50 or more deep web sources, and need to support interactive queries.

Evaluating Bidirectional Query Planning Algorithm. We use 20 queries. For each of them, we generate query plan using each of the above three algorithms. We compare the three algorithms in two aspects: *query plan execution time* and *query plan quality*. In our case, the latter is simply evaluated by counting the number of data records returned in response to a query, since a better data source is expected to contain more information. The results are shown in Figure 5.

Figure 5(a) shows the *cumulative histogram* for the query execution time speedup of our planning algorithm over the steiner tree algorithm. The x-axis shows the speedup, and the y-axis shows the *cumulative percentage* of queries



(a)Execution Time Speedup (Ours vs. Steiner)

(b) Execution Time Slow Down (Ours vs. OPT)

(c) Results Count Decrease (Ours vs. OPT)

Fig. 5. Query Planning Algorithms Comparison on Execution Time and Result Count

that have equal or lesser speedup. *In other words, in a cumulative histogram we present, a point* (x, y) *denotes that* $y\%$ *of the queries have a value less than or equal to* x.. We observe that for about 60% of the queries, the query plans generated by our bidirectional query planning algorithm achieves a speedup over the plans from the steiner tree algorithm. Furthermore, for about one third of the queries, the execution speedup achieved is larger than 30%. This is because the Steiner algorithm requires pair-wise node connection, and therefore, it may have to include some unnecessary data sources. In terms of number of returned results, our algorithm and the steiner tree algorithm are very similar.

Figure 5(b) and Figure 5(c) show the comparison between our algorithm and the exhaustive search algorithm in terms of query plan execution time and query plan quality, respectively. From Figure 5(b), we have the following observations. First, for about 60% of the queries, the query plans from our algorithm have exactly the same execution time as the optimal plan generated by the exhaustive search algorithm. Second, for more than 80% of the queries, the execution slow down of the plans from our algorithm is no more than 20%. From Figure 5(c), we have similar observations. For about 60% of the queries, our plans have the same result quality as the optimal plans, and for over 90% of the queries, the result quality from our plans is no more than 20% lower than the optimal one. These results show although in some cases (less than 20% of the queries), our plans are not as good as the optimal plans, for a vast majority of the cases, our query plans are as efficient and effective as the optimal plans. Considering the previous results (Figure 4), where we show that lack of scalability of the optimal algorithms, these results establish that our method will be practical for real systems.

5 Related Work

We now compare our work with existing work on related topics, including query planning, mediator systems, keyword search on relational databases, and deep web mining.

Query Planning: Query planning has been extensively studied in databases. Raschid and co-workers have developed a navigational-based query planning strategy for mining biological data sources [5,23]. Their navigational model captures the link structure between database objects. The key difference in our work is we focus on schema level dependencies between data sources.

Much work on query planning is based on the *Bucket Algorithm* [10,9,18,7]. In this work, it is assumed that the query, expressed as a predicate, specifies the databases that need to be queried. In our work, the query only contains search terms and does not specify any databases of interest. Our system selects the best data sources based on data source dependencies and query schemas. At the same time, query planning is also performed by our system.

Talukdar *et al* [29] proposed a system, similar to other work in [21,1,17,15,32,24], which uses Steiner Tree algorithm to find query plan which must be a *tree*. However, we formulate our query planning problem as a subgraph set cover problem and a query plan could be a graph. Srivastava *et al* [28] presented a query planning algorithm which focuses on minimizing the query's running time. Furthermore, they assume one attribute can only be provided by one data source, which is clearly distinct from ours. Tran *et al* [31] proposed a web data source search system, Search-WebDB. In their system, multiple data sources relevant to a query are queried in a distributed fashion simultaneously. Their query planning strategy is not applicable to our scenario, because the inter-dependencies between data sources in our case requires that the data sources must be accessed in a *specific order* according to the dependencies.

The work proposed in this paper is different from our previous deep web query planning work [33] in the following aspects. First, in this work, we considered two types of queries which are entity-attributes search and entity-entity relationship search. Second, in this paper, we considered the query planning problem as a graph problem, and proposed an effective and efficient heuristic planning algorithm based on a detailed domain ontology and several ranking functions.

Query Mediation Systems: Use of mediators is one of the classical approaches for information integration. There have been several well-developed systems in this area, include SIMS [2], Information Manifold [22], TSIMMIS [11], and Med-Maker [27]. The query plans from mediator systems have often been generated based on pre-specified rules or axioms. For example, the Mediator Specification (MS) rules are used in TSIMMIS and MedMaker, and the SIMS Axioms are used in SIMS. There are two key differences in our approach. First, query plans in our system are automatically generated based on data source dependencies and their relevance to the user query. Second, our data source model and the cost metric are very distinct, and as a result, the query planning formulation and algorithms are different.

Keyword Search on Relational Databases and databases with ontology: Recently, keyword search over relational databases has attracted a lot of attention [17,16,19,25,1,3,30]. In relational database keyword search, data tuples are represented as nodes, and foreign keys are represented as edges. In our case, since we don't have the access to the database behind deep web data sources, our graph model is in the metadata level. Furthermore, query answering plans are *trees* in above work which is distinct from our scenario. NAGA [20] is a search system answering query using a knowledge ontology. In NAGA, queries are expressed as graphs and the query answer is identified by searching for a subgraph which matches the query graph. This is clearly different from us because our query is keyword-like query and we search for subgraphs covering all search terms.

Deep Web Mining: Lately, there has also been much effort on mining useful information from the deep web [13,14,26]. This work has been focused on database integration, schema matching, and hidden data crawling. The focus of our work is distinct, i.e. answering complex cross-source queries over multiple inter-dependent deep web data sources. Our work assumes that a set of relevant deep web sources have been found and integrated for a particular domain.

6 Conclusions

In this paper, we considered answering cross-source queries over scientific deep web data sources. Our algorithm can support two type of queries, which are entity-attributes query and entity-entity relation query. We use a graph model to capture the dependencies between data sources. Finding a query plan over the dependence graph is NP-hard, and thus we proposed a heuristic bidirectional planning algorithm to find a query plan which could be a graph with disconnected components.

Our query planning algorithm is capable of finding a near optimal query plan for a query with the least execution time and highest quality. We compared our algorithm with an optimal exhaustive search algorithm, and the steiner tree algorithm. We find that our algorithm has good scalability, and can generate near optimal query plans for over 90% of the queries in terms of both query execution time and result quality. Furthermore, for over 60% of the queries, the query plans from our algorithm have speedup over the plans from the steiner tree algorithm.

References

1. Agrawal, S., Chaudhuri, S., Das, G.: Dbxplore: A system for keyword-based search over relational databases. In: Proceedings of the 18th International Conference on Data Engineering, p. 5 (2002)

2. Arens, Y., Knoblock, C.A., Shen, W.-M.: Query Reformulation for Dynamic Information Integration. Journal of Intelligent Information Systems - Special Issue on Intelligent Information Integration 6(2/3), 99–130 (1996)
3. Aditya, B., Bhalotia, G., Chakrabarti, S., Hulgeri, A., Nakhe, C., Parag, P., Sudarshan, S.: Banks: Browsing and keyword searching in relational databases. In: Proceedings of the 28th International Conference on Very Large Data Bases, vol. 28, pp. 1083–1086 (2002)
4. Bergman, M.K.: The deep web: Surfacing hidden value. Journal of Electronic Publishing 7 (2001)
5. Bleiholder, J., Khuller, S., Naumann, F., Raschid, L., Wu, Y.: Query planning in the presence of overlapping sources. In: Proceedings of the 10th International Conference on Extending Database Technology, pp. 811–828 (2006)
6. Braga, D., Ceri, S., Daniel, F., Martinenghi, D.: Optimization of multi-domain queries on the web. In: Proceedings of the VLDB Endowment, vol. 1, pp. 562–673 (2008)
7. Cali, A., Martinenghi, D.: Querying data under access limitations. In: Proceedings of the 24th International Conference on Data Engineering, pp. 50–59 (2008)
8. Davulcu, H., Freire, J., Kifer, M., Ramakrishnan, I.V.: A layered architecture for query dynamic web content. In: Proceedings of the 1999 SIGMOD Conference, pp. 491–502 (1999)
9. Doan, A., Halevy, A.: Efficiently ordering query plans for data integration. In: Proceedings of the 18th International Conference on Data Engineering, p. 393 (2002)
10. Florescu, D., Levy, A., Manolescu, I.: Query optimization in the presence of limited access patterns. In: Proceedings of the 1999 ACM SIGMOD international conference on Management of Data, pp. 311–322 (1999)
11. Garcia-molina, H., Papakonstantinou, Y., Quass, D., Sagiv, Y., Ullman, J.D., Vassalos, V., Widom, J.: The TSIMMIS Approach to Mediation: Data Models and Languages. Journal of Intelligent Information Systems 8, 117–132 (1997)
12. Gruser, J.-R., Raschid, L., Zadorozhny, V., Zhan, T.: Learning response time for websources using query feedback and application in query optimization. VLDB Journal 9, 18–37 (2000)
13. He, B., Zhang, Z., Chang, K.C.-C.: Knocking the door to the deep web: Integrating web query interfaces. In: Proceedings of the 2004 ACM SIGMOD International Conference on Management of Data, pp. 913–914 (2004)
14. He, H., Meng, W., Yu, C., Wu, Z.: Automatic integration of web search interfaces with wise_integrator. The International Journal on Very Large Data Bases 12, 256–273 (2004)
15. He, H., Wang, H., Yang, J., Yu, P.S.: Blinks: Ranked keyword searches on graphs. In: Proceedings of the 2007 ACM SIGMOD International Conference, pp. 305–316 (2007)
16. Hristidis, V., Gravano, L., Papakonstantinou, Y.: Efficient ir-style keyword search over reltional databases. In: Proceedings of the 29th International Conference on Very Large Data Bases (2003)
17. Hristidis, V., Papakonstantinou, Y.: Discover: Keyword search in relational databases. In: Proceedings of the 28th International Conference on Very Large Data Bases, pp. 67–681 (2002)
18. Ives, Z.G., Florescu, D., Friedman, M., Levy, A.: An adaptive query execution system for data integration. In: Proceedings of the 1999 ACM SIGMOD International Conference on Management of Data, pp. 299–310 (1999)

19. Kacholia, V., Pandit, S., Chakrabarti, S., Sudarshan, S., Desai, R., Karambelkar, H.: Bidirectional expansion for keyword search on graph databases. In: Proceedings of the 31st International Conference on Very Large Data Bases, pp. 505–516 (2005)
20. Kasneci, G., Suchanek, F.M., Ifrim, G., Elbassuoni, S., Ramanath, M., Weikum, G.: Naga: Harvesting, searching and ranking knowledge. In: Proceedings of the 2008 ACM SIGMOD International Conference, pp. 1285–1288 (2008)
21. Kementsietsidis, A., Neven, F., Craen, D.V.d., Vansummeren, S.: Scalable multi-query optimization for exploratory queries over federated scientific databases. Proceedings of the VLDB Endowment 1, 16–27 (2008)
22. Kirk, T., Levy, A.Y., Sagiv, Y., Srivastava, D.: The Information Manifold. In: Proceedings of the AAAI 1995 Spring Symp. on Information Gathering from Heterogeneous, Distributed Enviroments, pp. 85–91 (1995)
23. Lacroix, Z., Raschid, L., Vidal, M.-E.: Efficient techniques to explore and rank paths in life science data sources. In: Proceedings of the 1st International Workshop on Data Integration in the Life Sciences, pp. 187–202 (2004)
24. Li, G., Ooi, B.C., Feng, J., Wang, J., Zhou, L.: Ease: An effective 3-in-1 keyword search method for unstructured, semi-structured and structured data. In: Proceedings of the 2008 ACM SIGMOD International Conference, pp. 903–914 (2008)
25. Liu, F., Yu, C., Meng, W., Chowdhury, A.: Effective keyword search in retional databases. In: Proceedings of the 2006 ACM SIGMOD International Conference on Management of data, pp. 563–574 (2006)
26. Madhavan, J., Ko, D., Kot, L., Ganapathy, V., Rasmussen, A., Halevy, A.: Google's Deep Web Crawl. VLDB Endowment 1, 1241–1252 (2008)
27. Papakonstantinou, Y., Garcia-molina, H., Ullman, J.: Medmaker: A mediation system based on declarative specifications. In: Internation Conference on Data Engineering, pp. 132–141 (1996)
28. Srivastava, U., Munagala, K., Widom, J., Motwani, R.: Query optimization over web services. In: Proceedings of the 32nd International Conference on Very Large Data Bases, pp. 355–366 (2006)
29. Talukdar, P.P., Jacob, M., Mehmood, M.S., Crammer, K., Ives, Z.G., Pereira, F., Guha, S.: Learning to create data-integrating queries. Proceedings of the VLDB Endowment 1, 785–796 (2008)
30. Tata, S., Lohman, G.M.: Soak: Doing more with keywords. In: Proceedings of the 2008 ACM SIGMOD, pp. 889–901 (2008)
31. Tran, T., Wang, H., Haase, P.: Searchwebdb: Data web search on a pay-as-you-go integration infrastructure. In: WWW (2009)
32. Varadarajan, R., Hristidis, V., Raschid, L.: Explaining and reformulating authority flow queries. In: Proceedings of the 2008 IEEE ICDE International Conference, pp. 883–892 (2008)
33. Wang, F., Agrawal, G., Jin, R.: Query planning for searching inter-dependent deep-web databases. In: Ludäscher, B., Mamoulis, N. (eds.) SSDBM 2008. LNCS, vol. 5069, pp. 24–41. Springer, Heidelberg (2008)
34. Wang, F., Agrawal, G., Jin, R.: A system for relational keyword searches over deep web data sources. Technical Report OSU-CISRC-03/08-TR10, The Ohio State University (March 2008)
35. Wang, F., Agrawal, G., Jin, R., Piontkivska, H.: Snpminer: A domain-specific deep web mining tool. In: Proceedings of the 7th IEEE International Conference on Bioinformatics and Bioengineering, pp. 192–199 (2007)

Unified Modeling Technique for Threat Cause Ranking, Mitigation and Testing

Gaurav Varshney, Ramesh Chandra Joshi, and Anjali Sardana

Department of Electronics & Computer Engineering,
Indian Institute of Technology Roorkee,
Roorkee, Uttarakhand, India
{aheadpec,rcjosfec,anjlsfec}@iitr.ernet.in

Abstract. This paper describes a unified modeling technique applied after threat identification step of threat modeling process, for ranking the causes of a threat which is then used for threat mitigation and testing. The paper presents a unique approach that starts with enumeration of causes for each possible threat over the system with construction of threat cause model that diagrammatically describes the causes and sub-causes responsible for the occurrence of a threat. The paper suggests an approach for the ranking of both threats and their causes for effective mitigation. After applying threat cause mitigation strategy, testing of system towards a threat is verified by checking the security at the perimeter of the cause model for that threat. This unique technique assures that ensuring all sub-causes at lowest level of abstraction impossible will make the system safe towards a particular threat. Unlike other techniques this technique is unified as it starts with a threat model for each individual threat, that enumerates causes of their occurrence and then the same is used for mitigation and testing. Hence this strategy can ensure security when applied to all threats over the system.

Keywords: threat modeling, threats, causes, sub-causes, unified modeling technique.

1 Introduction

Today threat modeling is a major step that is followed in the system or software development process [1] [2]. The aim of threat modeling is to identify the threats over each independent module of a system that is being developed, understanding and categorizing threats so that an efficient mitigation strategy can be planned [6][2]. Active threat modeling process then tries to minimize the threats by applying the mitigation strategies, so that the vulnerability of developed modules for known threats can be minimized. The methodology also includes testing or validation, which counter checks the efficiency of the applied mitigation policy used for various modules [1]. It is observed that having a clear idea of the causes for threats over a module increases the understanding and selection of an efficient threat mitigation strategy. A very few policies exist in present scenario such as "Hierarchical Tree Approach" [6] that can really categorize causes for threats. The contribution and work in this paper is focused

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 491–500, 2011.

on proposing an unconventional unified modeling technique that can be used for threat cause ranking, mitigation and testing. The technique is called unified as it includes a cause model which is first used for ranking of causes for threats and then for choosing and determining appropriate mitigation strategies for each cause at lowest level of abstraction and then the same model is used in testing for evaluating reliability. The aim of using this model is to describe the causes for a threat at their lowest level of abstraction.

The paper is organized as follows: section 2 covers the previous work in threat modeling and provides a brief introduction to threat modeling process. In section 3 we have proposed our methodology for threat modeling, and discussed how it can be applied to a generic system. In section 4 we have concluded our work, with a description of intuitive advantages of our approach. At last section 5 provides an area for future work and section 6 includes references.

2 Related Work

The threat modeling process is not too old. The approach finds its existence since late 1990's, to increase the reliability of a system by minimizing the possible threats to which a system is vulnerable to [1] .The process of threat modeling comprises of various steps. There are five steps in the threat modeling process:

2.1 Decompose Application

The first step of threat modeling process is to decompose the application or the underlying system into individual components or modules that can be individually analyzed [2] [3]. This decomposition is required to analyze the vulnerabilities that exist over them. It also identifies trust boundaries, data flow, entry-exit points and privileged code [3] [4] [5].

2.2 Threat Identification

Threat identification is used in the process of threat modeling to identify the vulnerabilities that exists in the previous systems or that can exist in the new system if the system is going to be developed starting from scratch[1] [11]. This step can include many techniques to better represent a system in the form of independent and individual modules. One of these is through diagramming, by creating data flow diagrams of the whole system that shows how the data traverses throughout the system [1] [2]. The visualization of data flow helps to identify the places where in the security of the system can be compromised. The identification step lists all these weak points of a system or software [6]. These vulnerabilities that exist in the system can be actually exploited by an attacker for carrying out any unwanted and unauthorized work and hence these threats require proper attention for their mitigation [7].

However by analyzing this, we can say that there exists a trust in threat modeling process which is that "next step relies completely over the results of the previous steps". If in a case the previous step gives wrong result the overall threat modeling process will be of no worth to the system. For example, if threat identification lefts any threat undetected or unidentified in the system, threat categorization will categorize only those threats given to it by threat identification process and so threat mitigation can mitigate out only those threats. Until a thorough testing all the things looks good which is not the truth. Hence to save time, resources and manpower there should be a cent percent reliability in these steps. We also think that a simulated accuracy measure can be calculated for the results of threat modeling process depending upon the accuracy that each step is providing to its next step.

2.3 Threat Categorization

The next step that is applied over the results of threat identification is the categorization of the threats. This comprises of the identification of threats according to their behavior. Now actually behavior is nothing but the type of threat. Like STRIDE approach is used by Microsoft corporation in their threat modeling process that categorizes the threats into Spoofing, Tampering, Repudiate, Information disclosure, Denial of service, Elevate privilege[1] [2] [4] [6]. After this, the risk associated with threats is evaluated using DREAD model [2]. By categorizing the Threats into various categories, it is easy to plan a mitigation strategy for them as each type of threat requires a different kind of attention. We are going to discuss in our paper that how can we use the outcomes of threat categorization and rank the causes for Threats.

2.4 Mitigations

After enumeration or categorization of the Threats over the system the Threat mitigation step plans the mitigation strategies that are required for handling each different types of threat[1] [2] [6]. The application of mitigation strategies removes all identified and enumerated threats existing over the system. But while performing the mitigation for the known threats a ranked approach may do better in which instead of mitigating all threats in order, they can be mitigated based on their ranking.

2.5 Testing

Testing phase test the security of system after the application of mitigation strategies [1] [2]. If the system is found secure the process stops, else the process is carried out iteratively until a specified amount of accuracy of system towards enumerated threats is achieved.

3 Our Methodology

Our methodology is applied for each threat detected during the process of threat modeling over a module or a system.

3.1 Assumptions

While describing our technique we assume that system has gone through Threat identification and the result from this step are available in the form of threats with their causes and sub-causes. Let us assume that there exists a system S over which threat identification process is applied which reveals that the system is vulnerable to 'n' number of threats as follows: A1, A2, A3, A4…….An. Also each threat can have a variable number of causes for its existence, and causes can also have a number of sub-causes. Causes are denoted by C1, C2, and C3……Cm where m represents the number of causes for existence of a particular threat and hence can vary. Sub-causes are subscripted accordingly like if a cause C1 has three sub-causes then they are represented as C11, C12, and C13. In the description of our technique we will consider a single threat from 'n' threats over the system.

3.2 Threat Cause Model

Threat cause model is represented as a set of concentric circles representing threat, causes or sub-causes. The representation used for threat, its causes and sub-causes in the model are given below:

Table 1. Representation of entities

Entity	Representation
Threat	Innermost Circle
Causes	Concentric Circle (Diameter greater than innermost circle)
Sub-causes	Concentric Circle (Diameter greater than cause circle diameter)

3.2.1 Representing Threat

Threat whose cause is to be analyzed is represented with an innermost circle. So if one of the threat over a system is A1 then it will be represented as shown in Figure 1.

Fig. 1. Representing threat

3.2.2 Representing Causes at First Level of Abstraction

Causes that are responsible for the occurrence of the threat A1 are represented by concentric circles having diameter greater than that chosen for representing threat. Depending upon number of causes the new concentric region is equally divided for each cause. So if threat A1 have its causes as C1, C2 and C3 then it will be represented as shown in Figure 2:

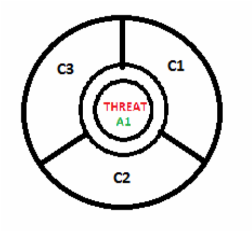


Fig. 2. Representing causes

3.2.3 Representing Sub-causes at Second Level of Abstraction

Sub-causes are nothing but a more precise division and description of a cause. Sub-causes are represented using concentric cirles with diameter greater than the cause whose sub-cause they are. The concentric arc for each cause is further extended and then divided in a number equal to its sub-causes. The representation is shown below in Figure 3 threat A1 has its causes as C1, C2, C3 with cause C1 divided into two sub-causes C11 and C12, similarly C2 is divided into C21, C22, C23 and C3 as C31, C32 as:

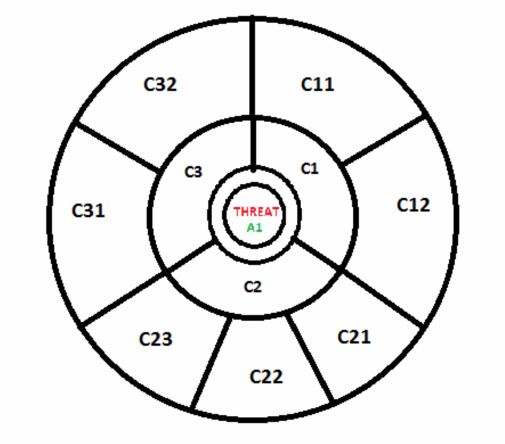


Fig. 3. Representing sub-causes (1)

It is not necessary in practial that every cause has its sub-causes. It might be the situation that for one of the cause we reached the lowest level of abstraction, and hence no further subdivision is possible. Hence the arc corresponding to those causes whose further subdivision is not possible are not extended in the cause model. The representation for such a situation is given in Figure 4 threat A1 has cause for its occurrence as C1, C2, C3 and C2 does not have its sub-causes as it cannot be defined more precisely, or we can say that it has reached its lowest level of abstraction for its description.

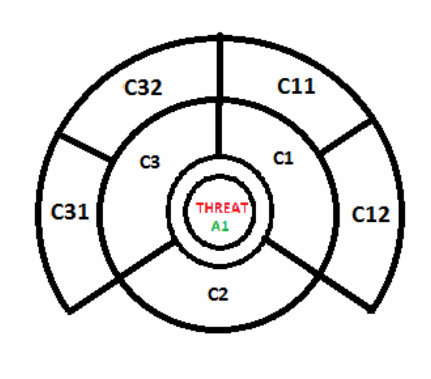


Fig. 4. Representing sub-causes (2)

Only those causes are subdivided into sub-causes whose detailed description has still not reached their lowest level of abstraction. By lowest level of abstraction we mean a detailed description of the base cause for which a mitigation strategy exists. The final model built using these steps will present a threat over the system with its possible causes and sub-causes. The outermost circular arc for each cause represents the sub-causes that are to be mitigated by applying suitable mitigation strategies. The mitigation of sub-causes ensures that the particular cause will not occur and will not be reponsible for the occurrence of that particular threat. When all such innermost causes have been mitigated, we have our system secured from the threat we are working upon.

3.3 Threat Cause Ranking

Our model depicts a clear description of sub-causes that can be mitigated to remove the innermost causes for the threats but while applying mitigation strategies, a ranked approach is better than a traditional unranked one. For example, if one threat has a number of causes for its occurrence, then a good approach is to remove causes in the descending order of their possibility rather than removing causes in a random order. Now how to calculate possibility is a major concern and we are going to present a technique for that in our text here. When we were doing research we found that a good plan of mitigation is the one which increases the security of system or decreases its overall vulnerability. And hence we concluded that probabilistic behavior can be used as a good threat cause ranking mechanism.

To calculate possibility, we assigned probability values to various causes and sub-causes in our final threat cause model achieved in the previous step i.e. figure.3. We assigned a probability value of 1 to each circular region. So now the innermost region i.e. the threat itself has a probability value of 1, similarly moving outwards from innermost to outermost region we assigned probability value 0.33 to each cause at the first level of abstraction. This is because we are taking an aspect that every cause is equally probable for the occurrence of the threat. As we move along to next level of abstraction we again divide the probability of the cause equally among the various sub-causes. If we have more levels then we can again divide the probability of the sub-cause equally among its next level sub-causes until outermost circle has been reached. The scenario for figure.3 is as shown below:

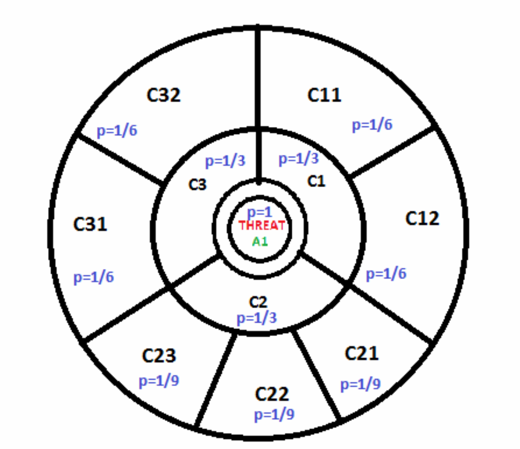


Fig. 5. Threat model after ranking causes and sub-causes

We are going to define two terms "probability" and "possibility". The reason is that we are taking each cause for a threat as equiprobable so "how we can rank causes". For this we say a cause is more possible than another if the individual probability values assigned to its sub-causes at lowest level of abstaction is lower than the individual probability values assigned to sub-causes of a cause at its lowest level of abstraction. Because this cause is more possible as more ways exists for its occurrence than any other cause for the occurrence of the same threat. Hence a more possible cause for a threat is the one which is having the same probability but higher possibility of occurrence. This technique can rank the base causes at different levels. Now if a threat mitigation startegy on a system just want to quickly decrease the probability of overall threats to the system it can mitigate those threats which are less possible. Threat ranking is also done based on the possibility that we used in case of ranking the causes of a threat. A threat whose possibility value i.e probability of its sub-cause at lowest level of abstraction in cause model is lower, is ranked higher than the other one.As each base cause is equiprobable so mitigating low possible cause will be quick as for that we have to deal with low number of detailed threat sub-causes. Like for example in Figure 5 for a quick mitigation strategy that do not require total mitigation but just wants to decrease the probability of a particular threat over its system then it is good to mitigate causes C1 and C3, as removal of them will decrease the probability of threat occurrence by a value of 0.66 using minimum number of mitigation strategies to handle C11, C12, C31,C32 and that is 4, instead of taking C1 or C3 with C2 which will ultimately cause a same decrease in probability of threat by a value of 0.66 but requires more number of mitigation strategies which is 5 in this case. Here we had assumed that a single mitigation strategy is required for the removal of sub-causes at the lowest level of abstraction. However a mitigation strategy that want the threat occurrence to be less possible will try to remove those causes for which several sub-causes exist for its occurrence. Because the occurrence of such a cause is more possible and hence removing or mitigating such a cause by applying mitigation strategy over its sub-causes at lowest level of abstraction will decrease the overall possibility of a given threat.

3.4 Threat Mitigation

Threat mitigation strategies are used to find and apply appropriate techniques through which the destruction caused by the threats over the system can be minimized or removed. This is a very important part of software threat modeling to mitigate the threats over the system. In our model the mitigation is done according to the ranking one has opted for. Means either one can go according to more possibility or he can start mitigation of the causes based on their respective probability.

While using any of these scenarios, we try to mitigate all the sub-causes at lowest level of abstraction for each cause, at first level of abstraction. For each sub-cause we will find out an appropriate way using which the chances of occurrence of that sub-cause can be minimized or reduced to zero. That strategy will be placed beside the sub-cause in the model indicating that one can use the given strategy to make the sub-cause impossible. One can use such a strategy to remove all the sub-causes of a cause at first level of abstraction. Now the reason why we are removing only the sub-causes at the lowest level is that we have divided the cause into its various sub-causes that specify its existence, so if we can remove those sub-causes we are in a surety that the particular cause will never occur. And in this way if we can remove all causes at first level of abstraction for a threat we can surely say that the particular threat will never occur as there is no possibility of even a single cause for the given threat. When such a strategy is applied to all the threats of a system then the system can be secured.

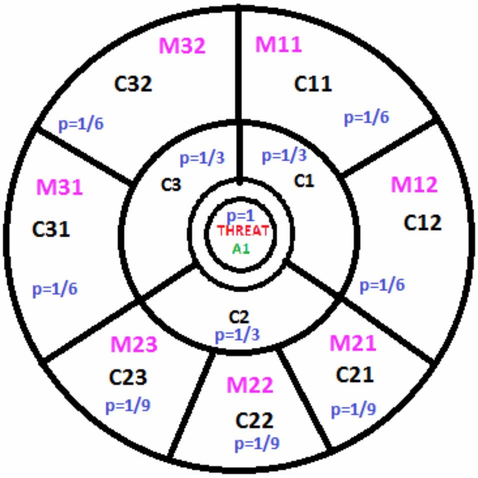


Fig. 6. Threat model with various mitigation startegies applied

Now if we have sub-causes at lowest level which are to be removed as C11, C12, C21, C22, C23, C31, C32, and the mitigation strategies to be applied so that these sub-causes will not occur are M11, M12, M21, M22, M31, M32, M33 then the figure.5 will look like as given in Figure 6. At this stage one has applied mitigation strategies for securing his system to various threats and has completed the part of threat mitigation. Still this is not the end of threat modeling process as before launching the system to work in real world it should be thoroughly tested for its safety towards threats.

3.5 Testing

While testing the system finally to a particular threat we test our final threat model for that threat by acting as an intruder. This means that we will attack over the system in such a manner so that any possible sub-cause at lowest level of abstraction in the cause model is made to exist, whose existence is already removed in mitigation process by the application of a particular strategy. If such happens this means that the existence of that sub-cause is not successfully removed yet, because of the inefficient strategy that is applied over it in the mitigation process. For such cases, the sub-causes are recorded and threat mitigation process is applied again by choosing more efficient strategy to make the existence of those sub-causes impossible in the system for future. Now if everything goes well one could not be able to show the existence of any of the sub-causes at the perimeter of the model. This means in other way that one could not be able to penetrate into the model by application of any strategy and hence this will make sure that the particular threat will never occur over the system.

In this way testing is applied to all possible threats over the system by using threat model for each one of them. If our system is secured from all the possible threats we have nothing to worry about, but if some of the threats over the system cannot be mitigated even after repetitive application of this process, we say that our system is partially secured. One of the testing results according to our model is system security percentage (SSP). SSP is defined as the ratio of number of threats completely mitigated to total numbers of threats detected during the threat identification process. So if the number of threats over a system is N1 and the number of threats mitigated during the threat mitigation process is N2, then the system security percentage is given as

$$SSP \quad = \quad N2/N1 \tag{1}$$

4 Conclusion

Unified threat modeling is a technique that helps to analyze and mitigate the possible threats, risks and vulnerabilities existing on a system, so as to increase its security and reliability. We in this paper proposed this technique for threat modeling in terms of threats and their causes. The technique is modular, as it is applied over each and every individual threat on a system and is complete, because for every threat it individually models various causes for their occurrences. It is easily understandable and as it does not assume any specific system for its application, the technique is generalized and can be applied over any system or subsystems. Hence this technique can be effectively used for modeling and mitigating threats over any type of system. However the validation of the technique is under study.

5 Future Work

We discussed the technique of unified threat modeling for modeling threats on a generic system; however we had not presented validation of our technique in the real world scenario. So our future research work will be focused on validating this technique; that will help us in evaluation and comparison with the present traditional approaches.

References

1. Shostack, A.: Experiences Threat Modeling at Microsoft (2008), http://www.homeport.org/~adam/modsec08/Shostack-ModSec08-Experiences-Threat-Modeling-At-Microsoft.pdf
2. Howard, M., Leblane, D.: Threat Modeling: Writing Secure Code, 2nd edn., vol. ch. 4. Microsoft Press (2002)
3. Meier, J.D., Mackman, A., Wastell, B.: Threat Modeling Web Applications. Microsoft Patterns & Practices, Microsoft Corporation (2005), http://msdn.microsoft.com/en-us/library/ff648006.aspx
4. Meier, J.D., Mackman, A., Dunner, M., Vasireddy, S., Escamilla, R., Murukan, A.: Threat Modeling: Improving Web Application Security:-Threats and Countermeasures. Microsoft patterns & practices, Microsoft Corporation, ch. 3, file:///F:/threat%20modeling/Threat%20Modeling.htm
5. Abi-Antoun, M., Wang, D., Torr, P.: Checking Threat Modeling Data Flow Diagrams for Implementation Conformance and Security. In: ASE 2007, November 7 (2007) ; short paper program
6. Threat Modeling: A Process to Ensure Application Security.: SANS Institute InfoSec Reading Room, http://www.sans.org/reading_room/whitepapers/securecode/threat-modeling-process-ensure-application-security_1646
7. Ambler, S.W.: Introduction to Security Threat Modeling, http://www.agilemodeling.com/artifacts/securityThreatModel.htm
8. Abdullah, S., Hussain, T., Khan, G.F.: Enhancing C4I Security using Threat Modeling. In: 12th International Conference on Computer Modelling and Simulation (2010)
9. Chen, Y., Boehm, B., Sheppard, L.: Value Driven Security Threat Modeling Based on Attack Path Analysis. In: Proceedings of the 40th Hawaii International Conference on System Sciences (2007)
10. Ebenezer, A., Oladimeji., S.S., Lawrence, C.: Security Threat Modeling and Analysis: A Goal-Oriented Approach. In: 10th IASTED International Conference on Software Engineering and Applications (SEA 2006), Dallas, Texas, USA (2006)
11. The Importance of Threat Modeling White Paper: Information Risk Management. In: IRM PLC (December 2007)
12. Threat Model Analysis. Microsoft Corporation, http://msdn.microsoft.com/en-us/library/aa561499v=bts.70.aspx
13. SDL Process: Introduction 2008. Microsoft Corporation, http://msdn.microsoft.com/en-us/library/cc307406.aspx

Automatic Face Recognition Using Multi-Algorithmic Approaches

S.M. Zakariya[1], Rashid Ali[2], and Manzoor Ahmad Lone[3]

[1,2] Department of Computer Engineering, Zakir Hussain College of Engineering and Technology, Aligarh Muslim University, Aligarh, India
[3] Department of Computer Science and Engineering Baba Ghulam Shah Badshah University, Rajouri, J & K, India
s.zakariya@gmail.com, rashidaliamu@rediffmail.com, mahmadlone@gmail.com

Abstract. Face recognition system has been evolving as a convenient biometric mode for human authentication. Face recognition is the problem of searching a face in the reference database to find a face that matches a given face. The purpose is to find a face in the database, which has highest similarity with a given face. The task of face recognition involves the extraction of different features of the human face from the face image for discriminating it from other persons. Many face recognition algorithms have been developed and have been commercialized for applications such as access control and surveillance. For enhancing the performance and accuracy of biometric face recognition system, we use a multi-algorithmic approach, where in a combination of two different individual face recognition techniques is used. We develop six face recognition systems based on the six combinations of four individual techniques namely Principal Component Analysis (PCA), Discrete Cosine Transform (DCT), Template Matching using Correlation and Partitioned Iterative Function System (PIFS). We fuse the scores of two of these four techniques in a single face recognition system. We pperform a comparative study of recognition rate of these face recognition systems at two precision levels namely at top- 5 and at top-10. We experiment with a standard database called ORL face database. Experimentally, we find that each of these six systems perform well in comparison to the corresponding individual techniques. Overall, the system based on combination of PCA and DCT is giving the best performance among these six systems.

Keywords: Face Recognition, PCA, DCT, Template Matching, PIFS, Multi-Algorithmic approach, ORL face database, recognition rate.

1 Introduction

A face recognition system would allow a user to be identified by simply walking past a surveillance camera. Human beings often recognize one another by unique facial characteristics. In the face recognition problem, a given face is compared with the faces stored in a face database in order to identify the person [1]. Facial recognition is

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 501–512, 2011.

one of the most successful forms of human surveillance. Facial recognition technology is one of the fastest growing fields of the biometric industry. Several studies have been reported in the last 10 years given in [2, 3, 4, and 11] that compare those algorithms. Research organizations are working on the development of more accurate and reliable systems. Among these, the prominent approaches are those based on Principal Component Analysis (PCA), Template Matching, Neural Network, Model Matching, Partitioned Iterated Function System (PIFS), Wavelet and Discrete Cosine Transform (DCT). For enhancing the accuracy and to boost the performance of the face recognition system a combination of these algorithms can be used.

In this paper, we discuss the four different face recognition techniques based on PCA, DCT, Template Matching using Correlation; PIFS and use a multi-algorithmic approach for the face recognition systems based on the different combinations of these individual techniques. The goal is to find which combination of these techniques performs better in terms of recognition rate.

This paper is organized as follows. In section 2, we discuss in brief the previous studies performed in the area of face recognition systems. In section 3, first we give an overview of the face database (ORL) used, then in section 4 we discuss the implementation steps for the face recognition systems based on Principal Component Analysis, Discrete Cosine Transform, Template Matching using Correlation and Partitioned Iterative Function System techniques. In section 5 we discuss the multi-algorithmic approach for the face recognition systems based on the different combinations of techniques discussed in section 4. In section 6 we show our experimental results and also analyse the results at two levels (Top 5 and Top 10 IDs). Finally, we conclude and give future directions.

2 Related Work

Some of the important studies on face recognition systems are discussed as below.

PCA also known as Eigen face method, In PCA method the images are projected onto the facial value so called eigenspace [6, 15]. The eigenvectors are ordered, each one accounting for a different amount of the variation among the face images. Each image location contributes more or less to each eigenvector. PCA approach reduces the dimension of the data by means of data compression basics [3] and reveals the most effective low dimensional structure of facial patterns. This reduction in dimensions removes information that is not useful [4] and precisely decomposes the face structure into uncorrelated components known as eigenfaces.

Face recognition based on template matching [14] represents a face in terms of a template consisting of several masks enclosing the prominent features e.g. the eyes, the nose and the mouth. In [17] a face detection method based on half face-template is proposed.

Recognition technique formulated on Partitioned Iterated Function System (PIFS) [7] makes use of the fact that human face shows region-wise (fractal) self-similarity, which is utilized for encoding the face to generate the PIFS code. Recognition is performed by matching these PIFS codes. In [20] the face recognition system based on partitioned Iterated Function System is proposed, in which face recognition based on PIFS representation and matching is carried out in the PIFS code domain. In [18] Bayesian approach to face recognition based on wavelet transform is proposed.

DCT is used to extract the features from images. To get the feature vector representing a face, its DCT coefficients are determined and only a subset of the DCT coefficients is retained. This feature vector contains low to mid frequency DCT coefficients, as these are the ones containing highest information [5]. In [19] illumination normalization is proposed by exploiting the correlation of discrete cosine transform (DCT) low-frequency coefficients to illumination variations.

3 ORL Face Database

The ORL face database was originally published by Cambridge University. ORL database contains a set of faces taken between April 1992 and April 1994 at the Olivetti Research Laboratory in Cambridge, U.K. [23]. Since 1994, ORL has been used to benchmark many face identification systems.

It consists of 400 face images taken from 40 people, 10 images per person. For each person, it contains face images under different lighting conditions, facial expressions, and poses. All the images are against a dark homogeneous background with the subjects in an up-right, frontal position, with tolerance for some tilting and rotation of up to about 20 degrees. The images are in bitmap file format (bmp), grayscale with a resolution of 92x 112 pixels. There are variations in images of different persons like persons have beard, persons have glasses, persons have moustaches etc.

4 Individual Techniques of Face Recognition

In this section, we have discussed implementation steps and also show the corresponding results in the given subsection of the following face recognition techniques: PCA, DCT, Template matching using correlation and PIFS. For implementing face recognition system based on the above techniques, we follow the following three steps: 1: Face preprocessing, 2: Feature vector extraction, 3: Recognition step.

4.1 Principal Component Analysis

Figure 1 shows the flowchart of PCA based face recognition system. First of all, we have to perform some preprocessing tasks. Here, we are resizing the images because original images have more values by which eigenface computation becomes complex. ORL face database images size is 92×112. After resizing the face image size is 64×64, and the resultant output are extracted by each individual technique.

4.2 Discrete Cosine Transform

The Discrete Cosine Transform (DCT) converts an image from spatial domain to frequency domain [19]. To recognize a particular input query face, the system compares the face's feature vector to the feature vector of the database faces using Euclidean Distance nearest neighbor classifier [5].In this paper, 2D DCT has been used to extract feature vectors from face image. Since, DCT has a property of accumulating image information in just few coefficients instead of using whole (say 64x64 pixel) image only 12x12 DCT coefficients are taken.

4.3 Template Matching

Template matching is the process of locating the position of a sub image inside a larger image. The sub image is called the template and the larger image is called the search area. The template matching process involves shifting the template over the search area and computing the similarity measure between the template and the window in the search area over which the template lies. The correlation between two signals (cross correlation) is a standard approach to feature detection [22] as well as a building block for more sophisticated recognition techniques.

4.4 Partitioned Iterative Function System

All images of natural or man made objects show region wise self similarity although they may not be globally self similar. Such images can be represented by Partitioned Iterative function System (PIFS) [21].

5 Multi Algorithmic Approach of Face Recognition

In multi-algorithmic approach of biometric face recognition, different algorithms are employed to recognize a query face. As pointed out in [10], fusing the scores of several techniques applied on the same data (face image) is a good approach to improve the overall accuracy of a biometric system [18]. Therefore, in this paper, we use a multi-algorithmic approach using different combinations of four face recognition techniques based on Principal Component Analysis (PCA), Discrete Cosine Transform (DCT), Template Matching using Correlation (Corr) and Partitioned Iterated Function System (PIFS). For combining two techniques, we fuse the scores of two individual techniques in order to increase the performance of the system [10]. A total of six combinations are formed namely PCA & DCT, PCA & Corr, PCA & PIFS, DCT & Corr, DCT & PIFS, and PIFS & Corr.

5.1 Recognition by the Pair of Two Combinations

In multi-algorithmic approach, first we extract the PCA feature vector, DCT feature vector, Templates and the PIFS code of the reference face database images as discussed in previous section. Reference face image database when subjected to Principal Component Analysis, PCA reference feature vector database is obtained. Likewise when the reference face image database is subjected to DCT and PIFS, DCT reference feature vector database and reference PIFS code database is obtained respectively. In Template Matching the templates of eyes, nose and mouth are created and stored as reference Template database. Six different combinations namely (i)PCA + DCT, (ii)DCT + Corr, (iii)DCT + PIFS, (iv)PIFS + Corr, (v)PCA + Corr, and (vi)PCA + PIFS of the four individual techniques are implemented. The schematic block diagram of the PCA+DCT approach is shown in the Figure 2. The schematic diagrams for the other five combinations are similar. We are not showing the schematic block diagram of the other approaches here due to space limitations.

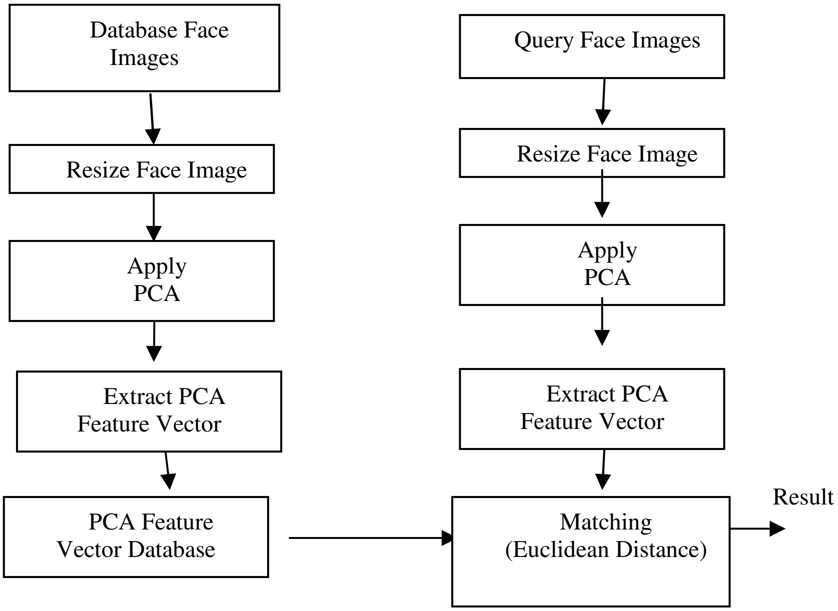

Database Face Images
Resize Face Image
Apply PCA
Extract PCA Feature Vector
PCA Feature Vector Database
Query Face Images
Resize Face Image
Apply PCA
Extract PCA Feature Vector
Matching (Euclidean Distance)
Result

Fig. 1. Flow chart of face recognition system based on PCA

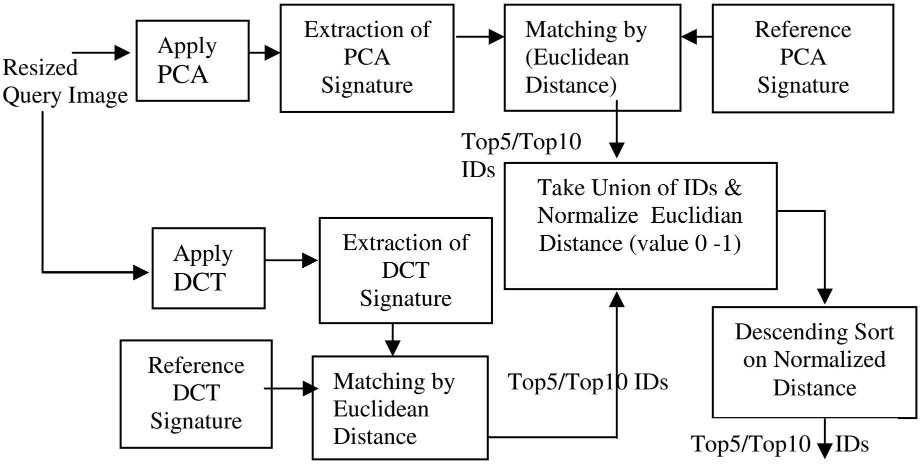
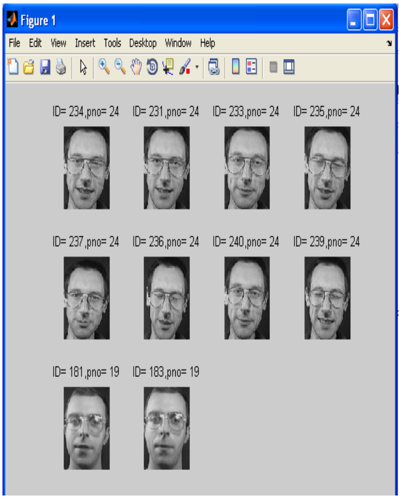

Resized Query Image
Apply PCA
Extraction of PCA Signature
Matching by (Euclidean Distance)
Reference PCA Signature
Top5/Top10 IDs
Take Union of IDs & Normalize Euclidian Distance (value 0 -1)
Apply DCT
Extraction of DCT Signature
Reference DCT Signature
Matching by Euclidean Distance
Top5/Top10 IDs
Descending Sort on Normalized Distance
Top5/Top10 IDs

Fig. 2. Schematic block diagram of the PCA+DCT Combination

6 Results and Discussion

In this work on multi-algorithmic face recognition, we implemented six face recognition systems by combining four individual techniques in the group of two. We also compare the performance of these six face recognition systems with the performance of the four systems each based on a single individual techniques. The experimental results are obtained on Olivetti Research Laboratory (ORL) face image database [23]. This ORL face database acts as the reference face database and contains 250 face images. There are 10 face images each of 25 different persons. These systems are evaluated on the basis of recognition rate obtained at two levels, namely (i) at top 5-IDs and (ii) at top 10-IDs.Each image in the reference face image database is made as query. The top 5-IDs and top 10-IDs are retrieved corresponding to each query image based on minimum Euclidean distance .Then, the average recognition rate for each system is determined at top 5-IDs and top 10-IDs. Top-10 results of the six face recognition systems based on multi-algorithmic approach are shown in figure 3, figure 4, figure 5, figure 6, figure 7 and figure 8. In each of these figures the first image is the query image.

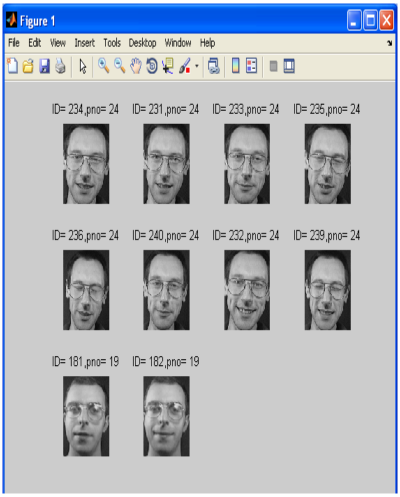


ID= 234,pno= 24 ID= 231,pno= 24 ID= 233,pno= 24 ID= 235,pno= 24

ID= 237,pno= 24 ID= 236,pno= 24 ID= 240,pno= 24 ID= 239,pno= 24

ID= 181,pno= 19 ID= 183,pno= 19

Fig. 3. PCA+DCT System Result



ID= 234,pno= 24 ID= 231,pno= 24 ID= 233,pno= 24 ID= 235,pno= 24

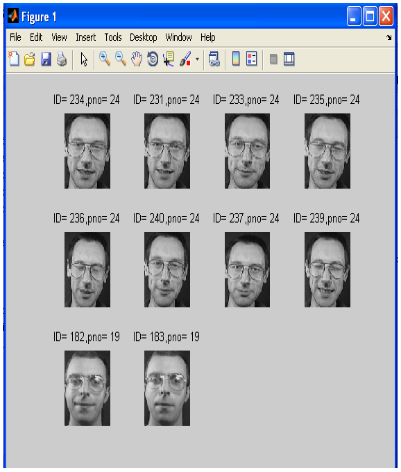
ID= 236,pno= 24 ID= 240,pno= 24 ID= 232,pno= 24 ID= 239,pno= 24

ID= 181,pno= 19 ID= 182,pno= 19

Fig. 4. DCT + Corr System Result

Top ten close matches (Top 10-IDs)

We have analyzed the results at two levels, at Top 5 IDs and at Top 10 IDs. Table 1 and Table 2 show the average face recognition rate at Top 5 IDs with four individual technique and six recognition system at Top 5 IDs by the combination of two individual algorithms. We find that the DCT based face recognition systems has higher recognition rate as compared to other three systems at this level, and also find the PCA+DCT has the highest recognition rate at this level, graphically shown in Figure 9 and Figure 10.



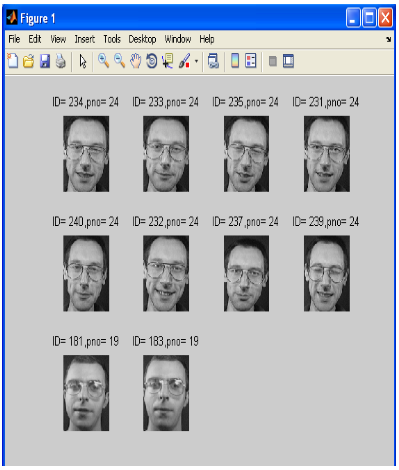
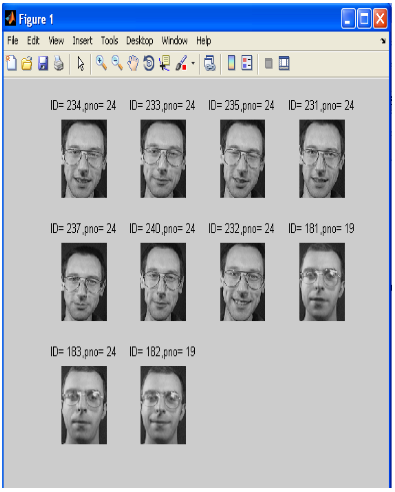


Fig. 5. DCT+PIFS System Result

Fig. 6. Corr + PIFS system Result

Top ten close matches (Top 10-IDs)



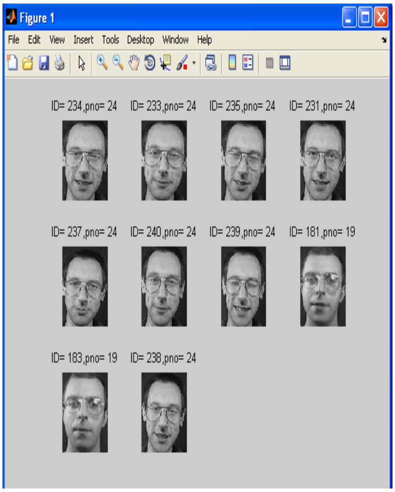


Fig. 7. PCA + Corr System Result

Fig. 8. PCA + PIFS System Result

Top ten close matches (Top 10-IDs)

Table 1. Recognition rate at top5-IDs of the four recognition systems

S. No.	Face recognition Methods	Recognition Rate (%)
1	PCA	0.9056
2	Corr	0.8964
3	PIFS	0.8853
4	DCT	0.9324

Table 2. Recognition rate at top 5-IDs of the six recognition system

S. No.	Face Recognition systems	Recognition Rate (%)
1	PCA+DCT	0.9424
2	PCA+PIFS	0.9240
3	PCA+Corr	0.9260
4	DCT+PIFS	0.9380
5	DCT+Corr	0.9410
6	Corr+PIFS	0.9100

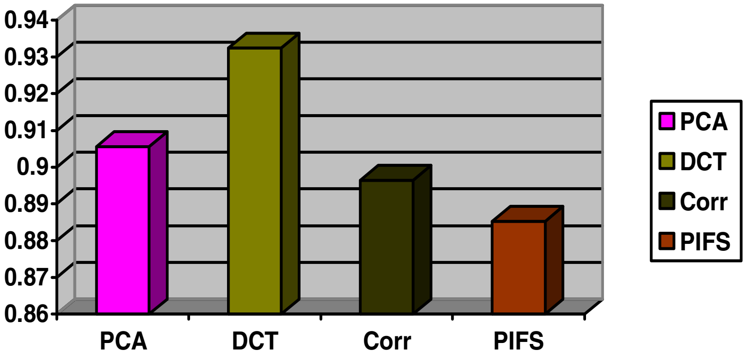


Fig. 9. Face Recognition rate of four individual systems at top 5-IDs

Table 3 and Table 4 show the average face recognition rate at Top 10 IDs with four individual technique and six recognition system at Top 10 IDs by the combination of two individual algorithms. We find that the DCT based face recognition systems has higher recognition rate as compared to other three systems at this level, and also find the PCA+DCT has the highest recognition rate at this level, graphically shown in Figure 11 and Figure 12.

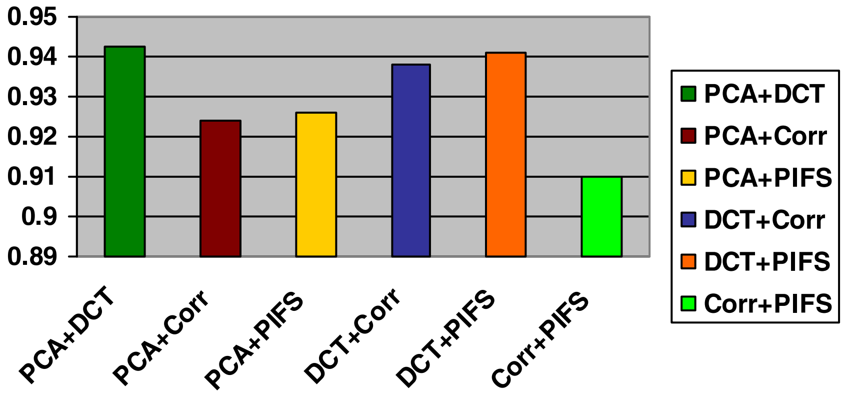


Fig. 10. Face Recognition rate of six systems based on combination of two individual at top 5-IDs

Table 3. Recognition rate at top 10-IDs of the four recognition systems

S. No.	Face recognition Methods	Recognition Rate (%)
1	PCA	0.7184
2	Corr	0.7084
3	PIFS	0.6808
4	DCT	0.7420

Table 4. Recognition rate at top 10-IDs of the six recognition system

S. No.	Face Recognition Systems	Recognition Rate (%)
1	PCA+DCT	0.7750
2	PCA+PIFS	0.7400
3	PCA+Corr	0.7500
4	DCT+PIFS	0.7520
5	DCT+Corr	0.765
6	PIFS+Corr	0.7300

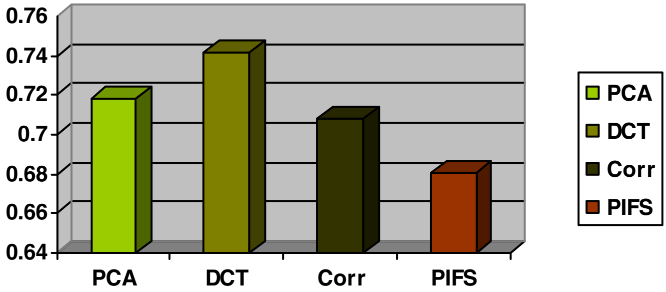


Fig. 11. Face Recognition rate of four individual techniques at top 10-IDs

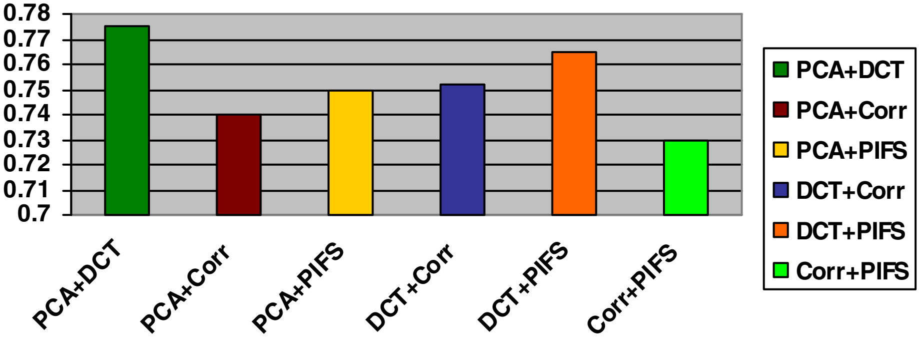


Fig. 12. Face Recognition rate of six systems based on combination of two individual at top 10-IDs

7 Conclusion

In this work, we reported the development of six multi-algorithmic face recognition systems based on four individual techniques namely PCA, DCT, Correlation and PIFS with the combination of two individual algorithms. In the multi-algorithmic approach, we combine these four individual techniques in a pair of two to obtain six combinations namely PCA+DCT, PCA+Corr, DCT+Corr, DCT+PIFS, PIFS+Corr, and PIFS+PCA. We performed our experiments with the standard ORL face database. Experimentally, we find that these combinations based systems provide better results than the corresponding individual techniques based system. The obvious reason for this is that the some IDs are returned by first system but not by the 2nd system in the pair and vice versa. When we combine these two techniques, these IDs got combined and the recognition rate in both the cases i.e. for the top5-IDs and for the top10-IDs increases. Out of these six combinations, the PCA+DCT based system has the highest recognition rate in both the cases i.e. for the top5-IDs and for the top10-IDs. In future we can implement in the combination of three and four individual algorithms.

References

1. Sirivich, D.L., Kirby, M.: Low-Dimensional Procedure for the Characterization of Human Faces. Journal of the Optical Society of America A: Optics, Image Science, and Vision, 519–524 (1987)
2. Gross, R., Shi, J., Cohn, J.: Quo vadis Face Recognition: Third Workshop on empirical Evaluation Methods in Computer Vision. Carnegie Mellon University, Pittsburgh (2001)
3. Blackburn, D., Bone, M., Phillips, P.: Facial Recognition Vendor Test 2000 Evaluation Report, Publish in National Institute of Science and Technology, Gaithersburg (2000)
4. Phillips, P.J., Moon, H., Rizvi, S., Rauss, P.: FERET Evaluation Methodology for Face Recognition Algorithms. In: IEEE Transaction on Pattern Analysis and Machine Intelligence (PAMI 2000), pp. 1090–1103. IEEE Computer Society Press, Los Alamitos (2000)
5. Hafed, Z.M., Levine, M.D.: Face Recognition Using the Discrete Cosine Transform. International Journal of Computer Vision 43(3), 167–188 (2001)
6. Turk, M., Pentland, A.: Eigenfaces for Recognition. Journal of Cognitive Neuroscience, 71–86 (1991)
7. Chandran, S., Kar, S.: Retrieving Faces by the PIFS Fractal Code. In: Proceedings of the Sixth IEEE Workshop on Applications of Computer Vision (WACV 2002), Orlando, Florida, pp. 8–12 (December 2002)
8. Chellapa, R., Wilson, C.L., Sirohey, S.A.: Human and Machine Recognition of Faces: A survey. Proceedings of the IEEE 83(5) (May 1995)
9. Gosthtasby, A., Gage, S.H., Bartholic, J.F.: A Two-stage Coss Correlation Approach to Template Matching. IEEE Transactions on Pattern Analysis and Machine Intelligence, 374–378 (1984)
10. Lemieux, A., Parizeau, M.: Flexible multi-classifier architecture for face recognition systems. Vision Interface, 1–8 (2003)
11. Yang, M.-H., Kriegman, D.J., Ahuja, N.: Detecting Faces in Images: A Survey. IEEE Transactions on Pattern Analysis and Machine Intelligence 24(1), 34–58 (2002)
12. Bryliuk, D., Starovoitov, V.: Access Control by Face Recognition using Neural Networks and Negative Examples. In: 2nd International Conference on Artificial Intelligence, Crimea, Ukraine, pp. 428–436 (September 2002)
13. Wiskott, L., Fellous, J.M., Krueuger, N., von der Malsburg, C.: Face Recognition by Elastic Bunch Graph Matching. IEEE Trans. on Pattern Analysis and Machine Intelligence 19(7), 775–779 (1997)
14. Brunelli, R., Poggio, T.: Face Recognition: Features versus Templates. IEEE Transactions on Pattern Analysis and Machine Intelligence 15(10), 1042–1052 (1993)
15. Chin, T.-J., Suter, D.: A Study of the Eigenface Approach for Face Recognition, Technical Report of Monash University, Dept. Elect. & Comp. Sys. Eng. (MECSE 2004) Australia, pp.1–18 (2004)
16. Nazeer, S.A., Omar, N., Khalid, M.: Face Recognition System using Artificial Neural Networks Approach. In: IEEE International Conference on Signal Processing, Communications and Networking, pp. 420–425 (2007)
17. Chen, W., Sun, T., Yang, X., Wang, L.: Face detection based on half face-template. In: Proc. of the IEEE Conference on Electronic Measurement and Instrumentation, pp. 54–58 (2009)
18. Liping, N., Yanbin, Z., Yuqiang, D., Yuan, L.X.: Combined Face Recognition Using Wavelet Transform and Bayesian. In: Proc. of the IEEE International Conference on Information and Computing Science, pp. 337–340 (2009)

19. Vishwakarma, V., Pandey, P., Gupta, S.: A Novel Approach for Face Recognition Using DCT Coefficients Re-scaling for Illumination Normalization. In: Proc. of the IEEE International Conference on Advanced Computing and Communications (ADCOM 2007), pp. 535–539 (2007)
20. Potgantwar, A.D., Bhiruid, S.G.: Web Enabled based Face Recognition using Partitioned Iterated Function System. International Journal of Computer Applications 1(2), 30–35 (2010)
21. Fan, C.: Matching Scheme based on PIFS of Compression for Image Retrieval. In: Proc. of the IEEE International Conference on Robotics and Biomimetics, pp. 2027–2031 (December 2007)
22. Gonzalar, R.C., Woods, R.E.: Digtal Image Processing, 3rd edn. Addiotion – Wesly, Readings (1992)
23. The Database of Faces, `http://www.cl.cam.ac.uk/research/dtg/attarcive/Facedatabase.html`

Assignment and Navigation Framework for Multiple Travelling Robots

Anshika Pal, Ritu Tiwari, and Anupam Shukla

Soft computing and Expert System Laboratory
ABV-Indian Institute of Information Technology and Management, Gwalior, India
anshika@iiitm.ac.in, ritutiwari@iiitm.ac.in,
anupamshukla@iiitm.ac.in

Abstract. Assignment Problem was well studied in the past 50 years, and is of great value in operations research and engineering. With the growing size of these problems and the new complexities introduced over the years, multi robot task assignment problems have become an important focus of assignment research. The work presented in this paper considers the scenario where multiple destination sites are available. The task of the controller is to assign a robot to each site as soon as possible in such a way so that robots can reach their destination with minimum travelling distance. Efficient algorithms for solving problem of this type have important applications in industries and home automation. Our main contributions are twofold: (a) A wave front based path planning method to compute the cost for the performance of each robot on each target (destination); and (b) An assignment algorithm for the assignment of robots to targets so that the sum of the total cost so obtained is as minimum as possible. The proposed approach has been tested through computer simulation. Experimental results demonstrate that our algorithm runs fast and produces near-optimal solutions on randomly generated instances.

Keywords: Optimization Theory, Operation Research, Assignment Problem, Path Planning, Mobile Robot, Wave Front Algorithm, Hungarian Algorithm.

1 Introduction

There has been significant recent interest in the use of multiple robotic devices to achieve tasks that a single robot could not achieve alone. There are various approaches to controlling multiple robotic devices. One is to have a centralized base station, either mobile or fixed, that collects position and other information on all of the robots and commands them to move to certain locations [1].

Assignment problems are fundamental in combinatorial optimization and roughly consist of finding a minimum weight matching in a weighted bipartite graph. They arise frequently in operations research, computer vision, as well as robotics, where graphs are recently emerging as a natural mathematical description for capturing interconnection topology [2-6]. Depending on the form of the cost function, assignment problems can be classified as linear or quadratic. Optimal solutions to the linear assignment problem can be computed in polynomial time using the Hungarian

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 513–523, 2011.

algorithm [7]. The quadratic assignment problem, however, is NP-hard [8], and suboptimal solutions are achieved by means of various relaxations. Approaches are either purely discrete [9-10] or continuous [11], based on the solution of differential equations that always converge to a discrete assignment.

In robotics, the assignment problem naturally arises in settings involving destination or target allocation. Depending on whether the discrete assignment is addressed simultaneously with the continuous navigation strategies or is solved independently in advance, approaches can be either online or offline. An online approach is proposed in [12], where the space of permutation-invariant multi robot formations is represented using complex polynomials whose roots correspond to the configurations of the robots in the formation. The proposed approach is open loop and centralized, since it requires global knowledge of the environment. On the other hand, in [13], a polynomial-time algorithm is developed that computes offline a suboptimal assignment between agents and destinations based on a "minimum distance to the goal" policy.

Motivated by the growing popularity of robot system and related optimization algorithms, this paper focuses on the multi robot assignment and navigation problem. The work presented in this paper, not only obtained a feasible solution, but also an efficient one.

The paper is organized as follows. The model assumptions and formulation of the problem is given in section 2. The proposed approach is described in section 3. Experimental results are presented in section 4. Finally conclusion is given in section 5.

2 Assumptions and Problem Formulation

2.1 Assumptions

The assumptions are divided into two parts: (1) the geometry of the environment; and (2) the characteristics and capabilities of a mobile robot.

1. *Environment Assumptions*
   The environment is a 2D plane and is occupied by stationary obstacles. The knowledge of all destinations and initial robot locations is given, as shown in fig.1. It is assumed that there are only a finite number of obstacles in the environment. The working space is partitioned into a grid of square cells, and a M X N board is gotten. If there is no obstacle in a cell, the cell is called free, otherwise called obstacle cell.

2. *Mobile Robot Assumptions*
   The mobile robot is given the coordinates of the start, and its destination position. The mobile robot has a memory to store position data and intermediate results. We assume the robot uses 45^0 as the unit for turning, since we only allow the robot to move from one cell to one of its eight neighbors.

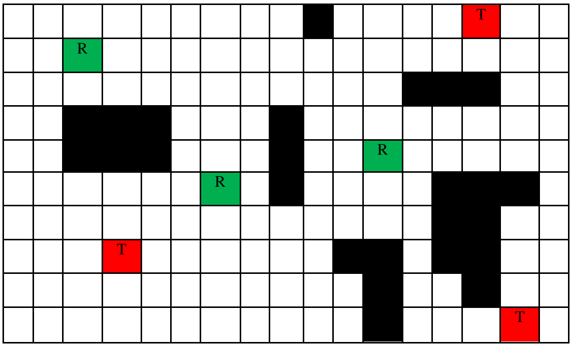


Fig. 1. An overview of the different state values of the grid cells in the environment. Cells representing obstacles are colored black, target points are colored red and cells that represent robot initial position are colored green. Accessible cells remain white.

2.2 Problem Formulation

The problem of immediate interest is as follows: given a number of destination locations, $\{T_1, T_2, \dots, T_n\}$, and a team of identical robots $\{R_1, R_2, \dots, R_n\}$, find the assignment guaranteeing that distinct destinations are assigned to distinct robots, while significantly reducing the assignment cost.

3 Proposed Approach

The operational overview of the system from a mission perspective, illustrated in fig.2. The locations of the targets are identified from the world map. The path planner is used to calculate the cost of the targets. Finally target assignment module determines the optimal combinations of robots and targets.

3.1 Focused Wave Front Method for Path Planning

The wave front based path-planning method considers the task of path planning as finding paths from the goal location back to the start location. This planning procedure works well in the environment with a uniform grid by propagating the waves through free space from the goal cell. Generally, from the goal cell, the values are spread to the start cell with adding one. A cell will be assigned with a value when it is not an obstacle and not assigned a given value yet. The value of each cell of the grid map will be measured based on this formula [14]:

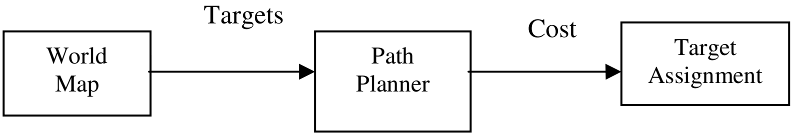


Fig. 2. Operational flow of the system

$$Map(i,j) = \begin{cases} Min\big(neighborhood(i,j)\big) + 1 & Empty \\ Nothing & Fill \end{cases} \qquad (1)$$

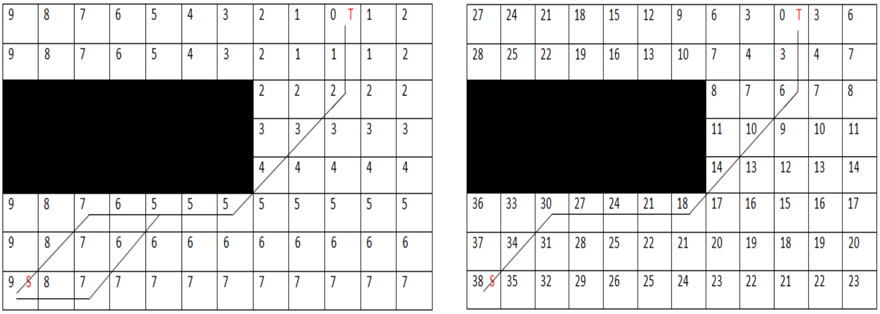
Where i, j are the row and the column number of the grid map and the neighborhood (i, j) is the vicinities around the (i, j) cell. The neighboring can be in 4 or 8 directions around the (i, j) cell. The Algorithm, then, will calculate the value of each cell by adding +1 to it when there is no obstacle there (according to the least value of neighboring cells) and assign it to the cell. In each stage only the cell which hasn't any value will get a value and those cells got a value in previous stage will not change. The process terminates when reaching the start point. After finishing the wave expansion, the robot initiates its movement toward the target point. The robot chooses from its 8 surrounding directions the cell with least value and moves toward that cell until it has reached the target point.

A simple illustration of wave front planning is shown in fig.3(a) using eightfold neighboring (i.e robot has the ability of moving in 8 directions). In this example, the number of transitions from the start to the goal are shown in the grid. The robot could simply follow the decreasing values in order to reach the goal, and could do so along a number of different paths in this example.

In case of adding one to the value of all neighbor cells (as shown in fig.3(a)), there is the selection problem for the shortest path because of the same value in neighborhood. This problem can be solved in [15] by assigning the different value according to the cell location. The values of orthogonal and diagonal neighbor cells are spread from the goal cell to the start cell, with adding three and four, respectively. Then, there are no cells having the same value in the planned path. Fig. 3(b) displays an example of this modified wave expansion planning method.

As shown from the fig.3 (a) and (b), these methods require that every cell in the map be filled by values to determine the path. This works well when there are a small number of cells. However, when planning a path using a regular grid map over a large area this becomes very time expensive.

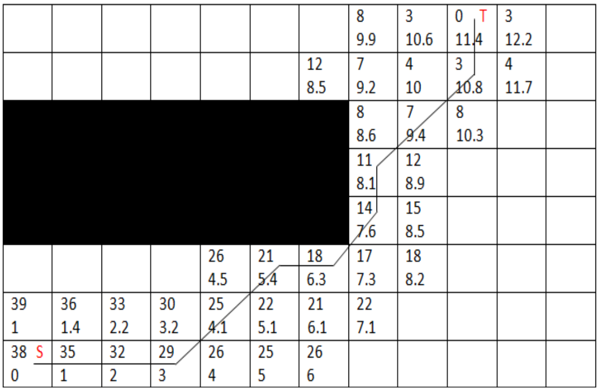
Our main idea is, during expansion only focus on the region which provides navigations towards the source in early stage without full expansions. Therefore, it only updates the values of all cells which are nearer to the source so that it can reach the source very quickly, and the expansion process terminates. Our idea is simple and time efficient.



9	8	7	6	5	4	3	2	1	0 T	1	2
9	8	7	6	5	4	3	2	1	1	1	2
							2	2	2	2	2
							3	3	3	3	3
							4	4	4	4	4
9	8	7	6	5	5	5	5	5	5	5	5
9	8	7	6	6	6	6	6	6	6	6	6
9 S	8	7	7	7	7	7	7	7	7	7	7

27	24	21	18	15	12	9	6	3	0 T	3	6
28	25	22	19	16	13	10	7	4	3	4	7
							8	7	6	7	8
							11	10	9	10	11
							14	13	12	13	14
36	33	30	27	24	21	18	17	16	15	16	17
37	34	31	28	25	22	21	20	19	18	19	20
38 S	35	32	29	26	25	24	23	22	21	22	23

Fig. 3. 'S' is start and T is target, (a) Simplified Wave Front Planning (left), two of the possible path are shown by Black line segments, (b) Modified Wave Front Planning (right)



							8 9.9	3 10.6	0 T 11.4	3 12.2	
						12 8.5	7 9.2	4 10	3 10.8	4 11.7	
							8 8.6	7 9.4	8 10.3		
							11 8.1	12 8.9			
							14 7.6	15 8.5			
				26 4.5	21 5.4	18 6.3	17 7.3	18 8.2			
39 1	36 1.4	33 2.2	30 3.2	25 4.1	22 5.1	21 6.1	22 7.1				
38 S 0	35 1	32 2	29 3	26 4	25 5	26 6					

Fig. 4. Focused Wave Front Planning

For every cell ‘c’ encountered during the wave expansion, it maintains two values ‘weight’ and ‘cost’:

Weight: this value is assigned according to the cell location. The values of orthogonal and diagonal neighbor cells are spread from the goal cell to the start cell, with adding three and four, respectively as proposed by the author [15].

Cost: this is the cost of getting from cell ‘c’ to start point. Here cost is the Euclidean distance of two 2D points.

The working steps are as follows:

Focused Wave Front Algorithm (FWF)

```
1. Initialize target with weight zero and cost with d(T,S), d is Euclidean cost, T is the
   target, and S is the source point.
2. While source not reached do,
3. Select a cell ‘c’ with minimum cost
4. Find those neighbors of ‘c’ that hasn’t got any value in previous stages.
5. Assign them weight and cost values accordingly.
```

Hence this algorithm only focuses on the cells from where the distance of the source is minimized. Once this is complete, we can simply follow the first value in reverse order until to reach the goal, avoiding any cell that is a obstacle. The process is illustrated in fig.4 (upper value in the cell represents the weight and the lower one is the cost). However, the path planned by this algorithm is not most optimal but time efficient.

3.2 Assignment Algorithm

This algorithm determines which pair of robot and target (destination) location to be assigned, i.e., answers the question who goes where?

The goal is to assign a robot to every target in such a way to minimize the total cost in the sense of robot to target travelling distance. Therefore, it is necessary to design an optimal solution to assign future locations fairly.

Consider a ‘n x n’ cost matrix ‘C’ which represents the cost of all individual assignments of robots to targets. Here, each entry in the cost matrix can be the length of the path the corresponding robot has to travel to reach the designated target points. The distances are defined according to the path planning algorithm discussed in section 3.

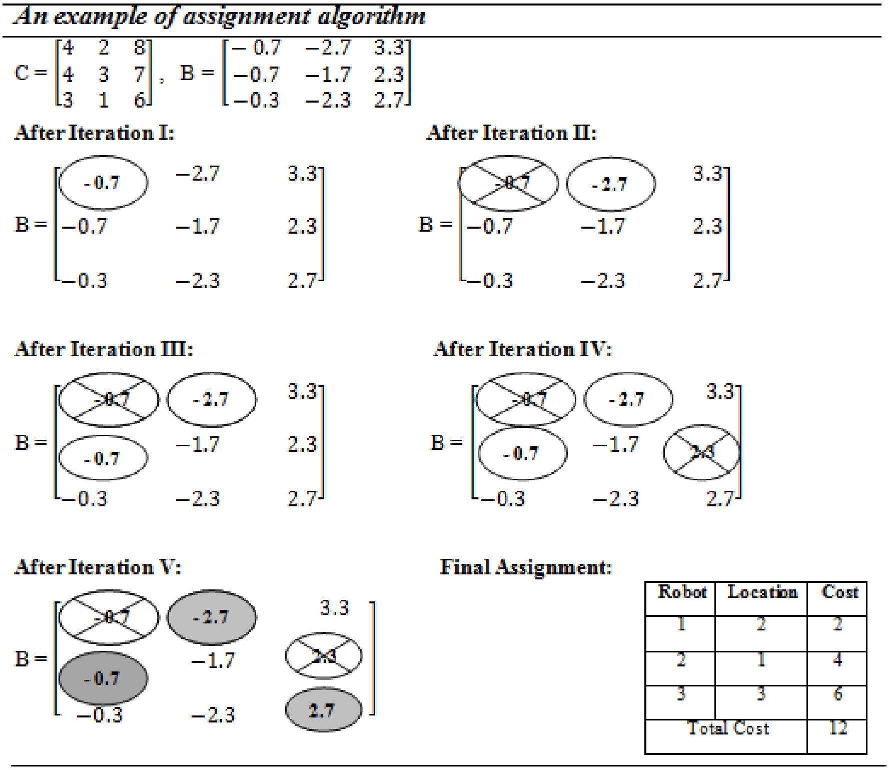
$$\mathrm{C} = \begin{bmatrix} C_{11} & C_{12} & C_{13} & \dots & C_{1n} \\ C_{21} & C_{22} & C_{23} & \dots & C_{2n} \\ C_{31} & C_{32} & C_{33} & \dots & C_{3n} \\ \dots & \dots & \dots & \dots & \dots \\ C_{n1} & C_{n2} & C_{n3} & \dots & C_{nn} \end{bmatrix}$$

This algorithm will automatically avoid collision with obstacles and walls and have the optimal cost, which will be referred to as real path cost. The method, which is able to find the optimal solution with the minimal cost given this matrix, can be summarized by the following steps:

Assignment Algorithm

```
1. Compute a reduced cost matrix ‘B’ by subtracting from each element the average
   cost of that row.
2. If each column in ‘B’ has an entry which is marked by circle, then go to step 8.
3. Sequentially select a column j from ‘B’, which does not contain any element that is
   marked by circle.
4. Find the unmarked minimal element $min_j$ from column j, suppose the index of
   corresponding row is r.
5. If row r contains an element $W_k$ ( in column k), which is marked by circle, than
      a. Compute,
         $cost\_min_j = \frac{1}{n}\sum_{i=1}^{n} C_{ij} - C_{rj}$ and $cost\_W_k = \frac{1}{n}\sum_{i=1}^{n} C_{ik} - C_{rk}$
      b. If $(cost\_min_j > cost\_W_k)$ then, Mark element $min_j$ by circle and $W_k$ by
         cross.
      c. Else, Mark element $W_k$ by circle and $min_j$ by cross.
6. Else, mark element $min_j$ by circle
7. Goto step 2.
8. Circled positions determine the possible assignment combinations.
```

In this algorithm optimal combinations are marked by ‘circles’ and the combinations that are not feasible are marked by ‘crosses’. Initially all entries in the matrix are in unmarked state. This assignment ensures that the total sum of path lengths of robot moving to their assigned target is minimized. A numerical example is shown in fig. 5.



An example of assignment algorithm

$$C = \begin{bmatrix} 4 & 2 & 8 \\ 4 & 3 & 7 \\ 3 & 1 & 6 \end{bmatrix}, \quad B = \begin{bmatrix} -0.7 & -2.7 & 3.3 \\ -0.7 & -1.7 & 2.3 \\ -0.3 & -2.3 & 2.7 \end{bmatrix}$$

After Iteration I:

$$B = \begin{bmatrix} -0.7 & -2.7 & 3.3 \\ -0.7 & -1.7 & 2.3 \\ -0.3 & -2.3 & 2.7 \end{bmatrix}$$

After Iteration II:

$$B = \begin{bmatrix} \cancel{-0.7} & -2.7 & 3.3 \\ -0.7 & -1.7 & 2.3 \\ -0.3 & -2.3 & 2.7 \end{bmatrix}$$

After Iteration III:

$$B = \begin{bmatrix} \cancel{-0.7} & -2.7 & 3.3 \\ -0.7 & -1.7 & 2.3 \\ -0.3 & -2.3 & 2.7 \end{bmatrix}$$

After Iteration IV:

$$B = \begin{bmatrix} \cancel{-0.7} & -2.7 & 3.3 \\ -0.7 & -1.7 & \cancel{2.3} \\ -0.3 & -2.3 & 2.7 \end{bmatrix}$$

After Iteration V:

$$B = \begin{bmatrix} \cancel{-0.7} & -2.7 & 3.3 \\ -0.7 & -1.7 & \cancel{2.3} \\ -0.3 & -2.3 & 2.7 \end{bmatrix}$$

Final Assignment:

Robot	Location	Cost
1	2	2
2	1	4
3	3	6
Total Cost		12

Fig. 5. An Assignment Example

4 Experiments

The proposed approach is simulated in Java. The robot and target positions are calculated at random in a virtual grid world of size 50X50, two simulation environment is shown in fig. 6. The map was taken as an input in form of an image. The image depicted the obstacles as black regions and the accessible area as the white region. The objective of the experiments is to mutually assign targets to robots, i.e., no two robots may be assigned to same target. The simulations have been accomplished using a variety of scenarios in which the number of robots and targets ranged from 5 to 8. The wave front (WF) expansion proposed by [15] for rectangular map and Hungarian method is used for the comparison of proposed focused wave front path planning and assignment modules respectively. Fig. 7 and 8 show the snapshots of the solution obtained using our assignment algorithm and the path followed by the robots using WF and FWF algorithms. Two types of experiments are performed, first experiment evaluates the performance of the path planning algorithm and second experiment evaluates the performance of the assignment algorithm. Ten simulations have been run per case, where the position of the robots and tasks has been calculated at random.

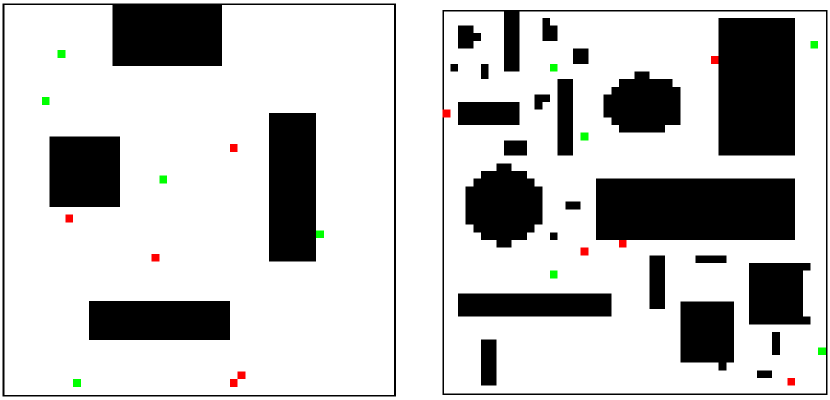


Fig. 6. Simulation environments, map m1(left) and m2(right), on which algorithm were tested, red dots represents targets and green are robots

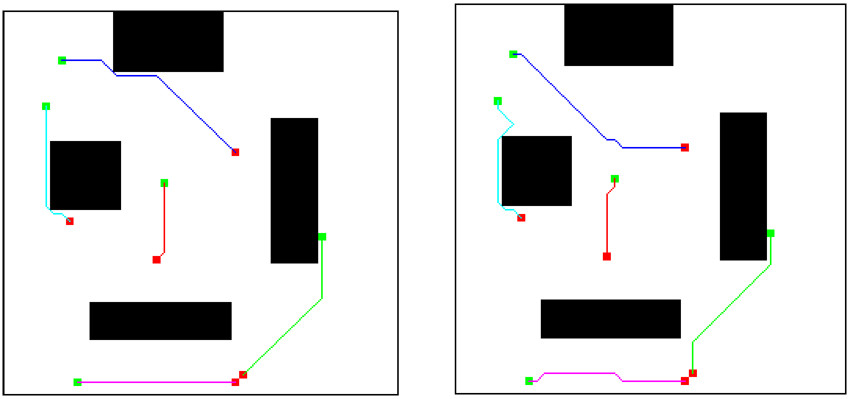


Fig. 7. One of the random mission obtained using WF(left) and FWF(right) on m1

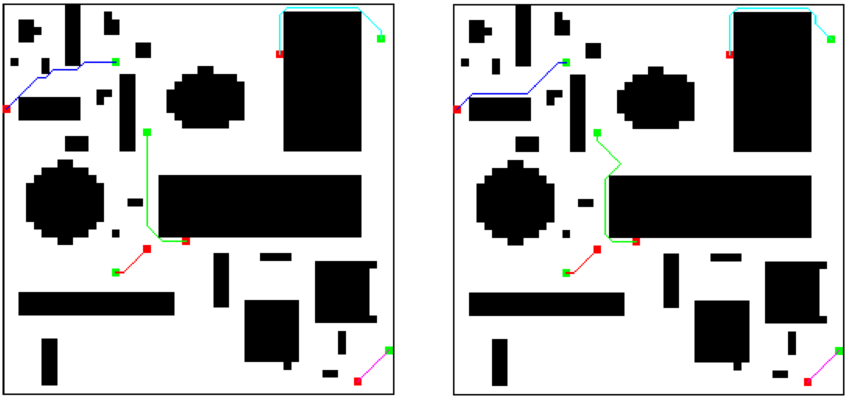


Fig. 8. One of the random mission obtained using WF(left) and FWF(right) on m1

4.1 Evaluation of Path Planning Algorithm

The aim of this part is to compare the capability of the wave front (WF) expansion proposed by [15] for rectangular map and the focused wave front (FWF) expansion to plan a valid path. Through experiments we can find out the influence of the cost function on the planned path. Performance of each of the algorithm is measured in terms of (a) average distance travelled by each robot to reach their assigned target; (b) average time to plan the valid path towards the targets; and (c) average number of turns on the planned path. Fig.9 shows that there is not much difference in both approaches in terms of the distance travelled by the robots. The great improvement is to be found in terms of time (fig. 11). The reason is that FWF algorithm is focuses only on the desired region instead of the entire workspace. However, the number of turns is more in FWF (fig. 10). The reason is that the search is performed only in the focused region thats why the movement choice is limited and turns increases. The path planned by this algorithm is not most optimal but time efficient.

4.2 Evaluation of Assignment Algorithm

This analysis, verifies the effectiveness of the assignment algorithm presented in section 3.2. Fig.12 shows the cost of the assignment using proposed assignment method and Hungarian method with FWF as a path planner. The results obtained from these simulations shows that there is not much difference between these two assignment methods. However the cost of Hungarian assignment is quite less, but this is not more significant. Our assignment method is simple and easy to implement.

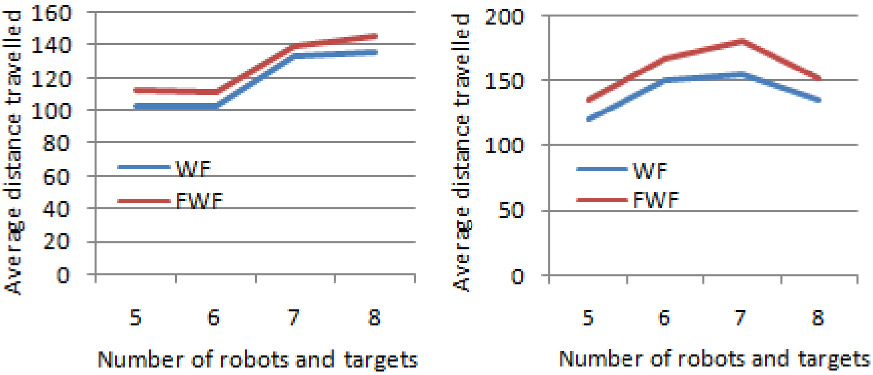


Fig. 9. Average distance travelled by robots in map m1(left) and m2(right)

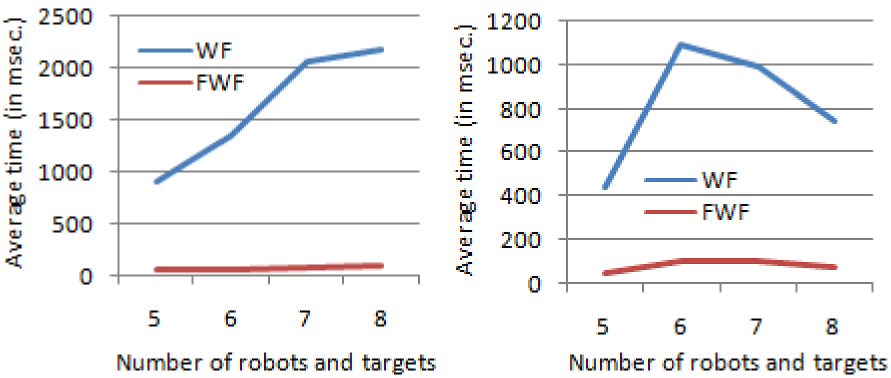


Fig. 10. Average time taken by path planner in map m1(left) and m2(right)

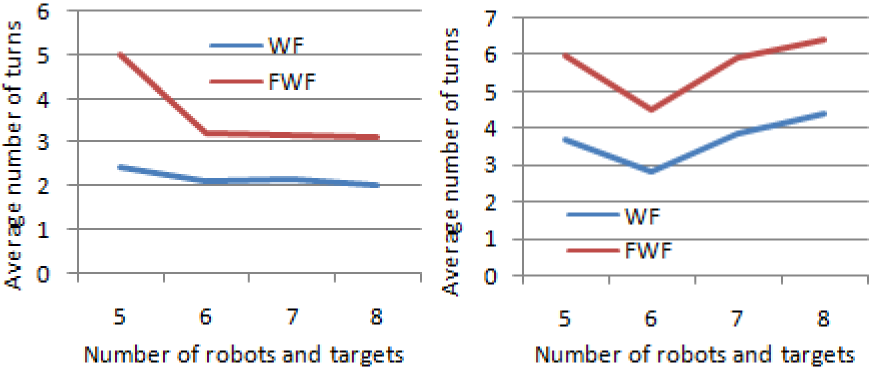


Fig. 11. Average number of turns on the planned path in map m1(left) and m2(right)

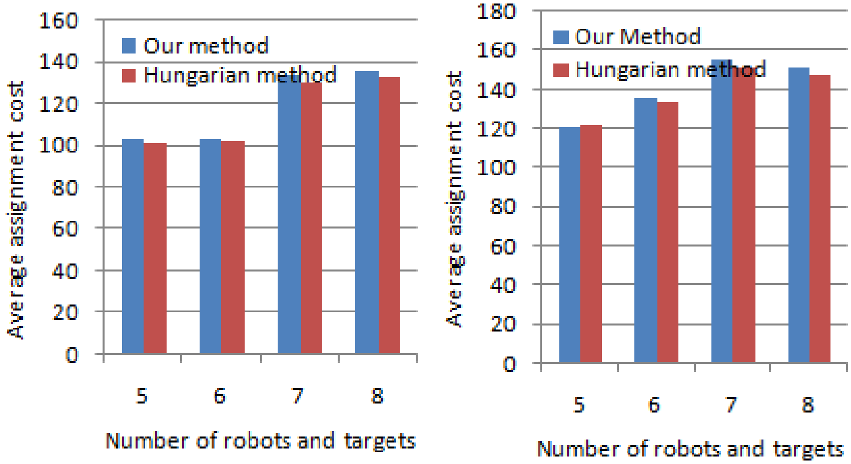


Fig. 12. Average assignment cost in comparison with hungarian method over 10 simulations per case on map m1 (left) and m2 (right

5 Conclusion and Future Work

In this paper, we have considered the integration of target assignment algorithm with a path planning algorithm. By combining these two problems, the robot's objective can easily obtained with satisfactory performance. The overall system was shown to always guarantee the mutual exclusion property of the final assignment. However the planned path using FWF algorithm is not most optimal but time efficient. This algorithm can be applied in the applications where optimality sacrifices for fast operation. Finally we have run experiments through simulation. Results show that the proposed framework works well in different situations and varying environments. Further works will concern a deeper study on the properties of algorithm.

References

1. Kwok, K.S., Driessen, B.J., et al.: Analyzing the Multiple-target-multiple-agent Scenario using Optimal Assignment Algorithms. SPIE Intelligent Systems & Automated Manufacturing, 13–17 (1997)
2. Jadbabaie, A., Lin, J., Morse, A.S.: Coordination of groups of mobile autonomous agents using nearest neighbor rules. IEEE Trans. Autom. Control 48(6), 988–1001 (2003)
3. Olfati-Saber, R., Murray, R.M.: Consensus problems in networks of agents with switching topology and time-delays. IEEE Trans. Autom. Control 49(9), 1520–1533 (2004)
4. Lafferriere, G., Williams, A., Caughman, J., Veerman, J.J.P.: Decentralized control of vehicle formations. Syst. Control Lett. 54(9), 899–910 (2005)
5. Balch, T., Arkin, R.C.: Behavior-based formation control for multirobot teams. IEEE Trans. Robot. Autom. 14(6), 926–939 (1998)
6. Ren, W., Beard, R.: Consensus seeking in multi-agent systems under dynamically changing interaction topologies. IEEE Trans. Autom. Control 50(5), 655–661 (2005)
7. Kuhn, H.W.: The hungarian method for the assignment problem. Naval Res. Logist. 2(1-2), 83–97 (1955)
8. Garey, M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-Completeness. W.H. Freeman, New York (1979)
9. Almohamad, H.A., Duffuaa, S.O.: A linear programming approach for the weighted graph matching problem. IEEE Trans. Pattern Anal. Mach. Intell. 15(5), 522–525 (1993)
10. Wolkowicz, H.: Semidefinite programming approaches to the quadratic assignment problem. In: Nonlinear Assignment Problems: Algorithms and Applications (Combinatorial Optimization), pp. 143–174. Kluwer, Norwell (2000)
11. Zavlanos, M.M., Pappas, G.J.: A dynamical systems approach to weighted graph matching. In: Proc. 45th IEEE Conf. Decis. Control, San Diego, CA, pp. 3492–3497 (December 2006)
12. Kloder, S., Hutchinson, S.: Path planning for permutation-invariant multirobot formations. IEEE Trans. Robot. 22(4), 650–665 (2006)
13. Ji, M., Azuma, S., Egerstedt, M.: Role-assignment in multi-agent coordination. Int. J. Assist. Robot. Mechatron. 7(1), 32–40 (2006)
14. Nooraliei, A., Nooraliei, H.: Path planning using wave front's improvement methods. In: International Conference on Computer Technology and Development (2009)
15. Oh, J.S., et al.: Complete Coverage Navigation of Cleaning Robots using Triangular Cell Based Map. IEEE Transactions on Industrial Electronics 51(3) (2004)

Cryptanalysis of Enhancements of a Password Authentication Scheme over Insecure Networks

Manoj Kumar[1], Mridul Kumar Gupta[2], and Saru Kumari[3]

[1] Department of Mathematics, R. K. College, Shamli,
Muzaffarnagar Uttar Pradesh, India
[2] Department of Mathematics, Choudhary Charan Singh University,
Meerut, Uttar Pradesh, India
[3] Department of Mathematics, Agra College, Agra, Uttar Pradesh - India
yamu_balyan@yahoo.co.in, mkgupta2002@hotmail.com,
saryusiirohi@gmail.com

Abstract. In 2005, Liao et al. pointed out some weaknesses in Das et al.'s dynamic ID-based scheme. They proposed a slight modification to Das et al.'s scheme to improve its weaknesses. In 2008, Gao-Tu, and in 2010, Sood et al., found vulnerabilities in Liao et al.'s scheme; and independently proposed its security enhanced versions. However, we identify that Gao-Tu's scheme is insecure against user impersonation attack, server counterfeit attack, man in the middle attack, server's resource exhaustion attack and does not provide session key agreement. We also demonstrate that Sood et al.'s scheme is still vulnerable to malicious user attack in different ways and user's password is revealed to the server. Besides both the schemes have no provision for revocation of lost or stolen smart card. Our cryptanalysis results are important for security engineers, who are responsible for the design and development of smart card-based user authentication systems.

Keywords: Authentication, Password, Smart Card, Network Security, Cryptanalysis.

1 Introduction

Remote user authentication has become an important technique in modern computer network systems. This technique is used to validate the legitimacy of a remote login user. Due to the convenience, secure storage, and secure computation of smart card, many smart card based remote user authentication schemes have been proposed [1, 2, 3, 4, 15, 17, 18, 19, 21, 22]. Most of these schemes have the following properties: (1) the server has no password table; (2) users can freely choose their own passwords; (3) it demands only low communication and computation cost; (4) mutual authentication is provided between a user and a server.

However, these schemes did not protect the user's identity, even though user anonymity is an important issue in many e-commerce applications. In 2004, Das et al. proposed a smart card based user authentication scheme [5] by adapting a dynamic identification being changed in each login request to achieve the user anonymity. In

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 524–532, 2011.

2005, Liao et al. pointed out that Das et al.'s scheme has three weaknesses: it cannot protect against guessing attacks; it cannot achieve mutual authentication; password can be revealed by the remote systems [6, 14, 16, 20,23]. Therefore, they proposed a slight modification to Das et al. scheme to improve these weaknesses. Excepting Liao-Lee-Hwang scheme, a number of papers [7, 8, 9,10] have proposed to improve Das et al. scheme. However, all of these improved schemes can't achieve mutual authentication, some of them even lost the merit of dynamic ID, and most of them have also little security leak in password change phase.

In 2008, Gao-Tu [11] analyzed Liao et al.'s scheme and pointed out that any user can login into remote system using an arbitrary password even if he don't know the correct password; and demonstrated some attacks such as smart card loss attacks, message forgery attacks, masquerade attack and denial of service attacks on the scheme. They also proposed an improvement to Liao et al.'s scheme to resolve its shortcomings.

In 2010, Sood et al. [12] found that the Liao et al.'s scheme is vulnerable to malicious user attack, Ku and Chang's impersonation attack [13], Awasthi's stolen smart card attack [14] and offline password guessing attack. They also pointed out that Liao et al.'s scheme does not maintain the user's anonymity and its password change phase is insecure. To remedy these pitfalls, they also presented a scheme which they claimed to inherit the merits of different dynamic identity based authentication schemes and resolve the aforementioned problems.

Unfortunately, this paper describes that Gao-Tu's scheme is insecure against user impersonation attack, server counterfeit attack, man in the middle attack, server's resource exhaustion attack and does not provide session key agreement. We also demonstrate that Sood et al.'s scheme is still vulnerable to malicious user attack in different ways which we shall point out more clearly later. Besides both the schemes have no provision for revocation of lost or stolen smart card.

The remaining sections of this paper are organized as follows. Section 2 is about the notations. Section 3 reviews Gao-Tu scheme and describes its weaknesses. Section 4 reviews Sood et al.'s scheme and describes its security flaws. Finally section 5 concludes the article.

2 Notations

- U: the user.
- ID: the identity of U.
- PW: the password of U.
- T_U: timestamp of U.
- SC: the smart card of U.
- S: the smart card.
- x: the secret key of S.
- y: S's secret number stored in each user's smart card .
- T_S: timestamp of S.
- ΔT: the maximum time interval for communication delay.
- U_A: the attacker.
- U_K: the malicious user.

- T_K: timestamp of U_K.
- $\oplus$: the bitwise XOR operation.
- h(.): a cryptographic hash function.
- $\Rightarrow$: a secure channel.
- $\rightarrow$: a common channel.
- ||: the string concatenation.

3 Review and Security Analysis of Gao-Tu's Scheme

3.1 Review of Gao-Tu's Scheme

This scheme is composed of the registration phase, verification phase and the password change phase.

Registration Phase
1) U selects his ID and PW and computes h(ID || PW).
1) $U \Rightarrow S$: {h(ID || PW)}
2) S computes $M = h(ID \| PW) \oplus h(x)$ and $N = h^2(ID \| PW) \oplus h^2(x)$.
3) S personalizes U's smart card with the parameters (M, N, y, h(.)).
4) $S \Rightarrow U$: {SC}

Verification Phase: In this phase, S authenticates U when U login to S. After the successful verification of the login request, S allows U to access the system. All steps of this phase are shown as follows:

1) U inserts his SC into the card reader, and keys in ID and PW.
2) SC computes h(ID || PW), $L = h(h(ID \| PW) \oplus M)$ and $K = h^2(ID \| PW)) \oplus N$.
3) If $K \neq L$, SC will notice an invalid login request and prompt U to input ID and PW again (not more than three times or this card would be prohibited); otherwise
4) SC computes $CID = h(ID \| PW) \oplus h(N \oplus y \oplus T_U)$.
5) $U \rightarrow S$:{CID, N, T_U}.
6) On receiving the login request {CID, N, T_U}, S records the current timestamp T_S, and verifies if $(T_S - T_U) \leq \Delta T$. If so, S computes $h(ID \| PW)^* = CID \oplus h(N \oplus y \oplus T_U)$ and $N^* = h(h(ID \| PW)^*) \oplus h^2(x)$.
7) If $N^* = N$, S records the current timestamp T_S^* and computes $D = h(h(ID \| PW)^* \oplus h(x) \oplus T_S^*)$.
8) $S \rightarrow U$: {D, T_S^*}.
9) On receiving {D, T_S^*} at the time T_U^*, U verifies if $(T_U^* - T_S^*) \leq \Delta T$. If it holds, U computes $h(M \oplus T_S^*)$ and compares with the received D. If both are equal, U confirms the authenticity of S.

Password Change Phase: This phase is invoked whenever U wishes to change his password. The phase works as follows:

1) U inserts his SC into the card reader, and keys in his original ID and PW.
2) SC computes h(ID || PW), $L = h(h(ID \| PW) \oplus M)$ and $K = h^2(ID \| PW)) \oplus N$.

3) If $K \neq L$, SC prompts the original PW incorrect, and rejects changing password; else, calls for a new password PW_{NEW}, which user wants to change to.
4) SC computes $h(ID \| PW_{NEW})$, $M_{NEW} = M \oplus h(ID \| PW) \oplus h(ID \| PW_{NEW})$, $N_{NEW} = N \oplus h^2(ID \| PW) \oplus h^2(ID \| PW_{NEW})$.
5) SC replaces M and N with M_{NEW} and N_{NEW}.

3.2 Security Analysis of Gao-Tu's Scheme

User impersonation attack: A malicious user U_K can extract the values y, h(x) and $h^2(x)$ from his SC; all these three values are common to all users. U_K intercepts a login request $\{CID, N, T_U\}$ of U, computes $h(N \oplus y \oplus T_U)$, extracts $h(ID \| PW) = CID \oplus h(N \oplus y \oplus T_U)$. Now U_K can easily compute a valid login request $\{CID = h(ID \| PW) \oplus h(N \oplus y \oplus T_K), N, T_K\}$. As T_K is fresh and ID & PW are valid identity and password of U, obviously U_K would successfully pass the authentication phase. This attack is mounted by U_K without knowing ID and PW of U and without having access to SC of U.

Server Counterfeit Attack: This subsection presents that a legal but malicious user U_K intends to counterfeit a legal server to spoof U. Similarly U_K can extract secret value y from his SC. In this attack U_K is required to make a legal but fake response $\{D^* = h(h(ID \| PW) \oplus h(x) \oplus T_K), T_K\}$. First U_K eavesdrops a login request $\{CID, N, T_U\}$ sent by U. Next U_K computes $h(ID \| PW) = CID \oplus h(N \oplus y \oplus T_U)$ and selects a valid timestamp T_K (it is possible for U_K to obtain a legal timestamp T_K since U_K can consecutively monitor all transaction behaviors involved with U). Then U_K computes a fake value $D^* = h(h(ID \| PW) \oplus h(x) \oplus T_K)$ and issues $\{D^*, T_K\}$ as response to U. Once U receives $\{D^*, T_K\}$, he checks the validity of T_K and D^*. Obviously all verification will pass and U believes that he currently communicates with legal server (actually is U_K). As a result the server counterfeit attack is not prevented in Gao-Tu's scheme.

Man in the Middle Attack: According to the above two subsections we know that the user impersonation attack and the server counterfeit attack can easily be invoked by a legal but malicious user U_K. Therefore it is possible for U_K to construct a man in the middle attack.

Server's Resource Exhaustion Attack: S stores nothing with itself corresponding to a user. Moreover, upon receiving a login request, S do not checks the validity of ID and PW of U separately. The value (related to U) playing key role in, personalizing SC/ creation of login request/ creation of response message from S to U/ verification of U by S/ verification of U by SC, is $h(ID \| PW)$ rather than ID & PW separately. Taking opportunity of this fact U_K can successfully login into S as an arbitrary user without getting registered to it. For this he selects an arbitrary value Z of bit length same as that of $h(ID_K \| PW_K)$ where ID_K & PW_K are identity and password respectively of U_K. Next U_K computes $N^* = h(Z) \oplus h^2(x)$ and $CID^* = Z \oplus h(N^* \oplus y \oplus T_K)$, where U_K extracts $h^2(x)$ and y from his SC and T_K is currently chosen timestamp. Then he sends $\{CID^*, N^*, T_K\}$ as login request to S. On receiving $\{CID^*, N^*, T_K\}$, S computes $Z^* = CID^* \oplus h(N^* \oplus y \oplus T_K)$ and $N^{**} = h(Z^*) \oplus h^2(x)$; checks whether

$N^* = N^{**}$ (which will obviously be true). Consequently S accepts the login request, computes $D^* = h((Z^*) \oplus h(x) \oplus T_S)$ and sends $\{D^*, T_S\}$. As a result S's resources will get exhausted for an unregistered user. Such an attack can cause great loss to S in case of secure transaction, e.g. online banking and e-commerce, etc.

Attack on user's Anonymity: U inserts his SC into the card reader to login onto S and submits its ID & PW. The SC first verifies the ownership of SC and then computes $CID = h(ID \| PW) \oplus h(N \oplus y \oplus T_U)$ and sends $\{CID, N, T_U\}$ to S. The value N remains same for different login requests belonging to same user. Hence login requests belonging to same user can be traced out and can be interlinked to derive some information related to the user.

Moreover, if U_A manages to record two login requests belonging to same user: $\{CID_1, N, T_{U1}\}$ and $\{CID_2, N, T_{U2}\}$. Then U_A computes $CID_1 \oplus CID_2 = [h(ID \| PW) \oplus h(N \oplus y \oplus T_{U1})] \oplus [h(ID \| PW) \oplus h(N \oplus y \oplus T_{U2})] = h(N \oplus y \oplus T_{U1}) \oplus h(N \oplus y \oplus T_{U2})$. Now he guesses a value y^*, computes $A = h(N \oplus y^* \oplus T_{U1}) \oplus h(N \oplus y^* \oplus T_{U2})$; and checks if $A? = CID_1 \oplus CID_2$. He repeats this process until the equivalence $A = CID_1 \oplus CID_2$ guarantees his correct guess of y. After this success he uses any of recorded login requests to extract $h(ID \| PW) = CID_1 \oplus h(N \oplus y \oplus T_{U1})$. Now U_A can easily mount user impersonation attack, server counterfeit attack and man in the middle attack in similar way as described for U_K.

Absence of Session Key Establishment: To maintain the conversation privacy after authentication, messages encryption is imperative. There are no session keys established in Gao-Tu's scheme; therefore, privacy cannot be ensured. The authors might argue that the session key establishment is not the focus of the scheme or that the related parties can run another protocol to establish a session key for messages encryption; however, we disagree with these contentions. Firstly we deem that the session key establishment is an essential requirement when designing an authentication scheme. The scheme designers should not neglect it. Secondly, employing another protocol to establish a session key is inefficient due to the increased calculation and communication costs.

4 Review and Security Analysis of Sood et al.'s Scheme

4.1 Review of Sood et al.'s Scheme

In this section we examine Sood et al.'s scheme composed of the registration phase, login phase, verification and session key agreement phase and the password change phase.

Registration Phase:

1) $U \Rightarrow S$: {ID, PW)}.
2) S computes $N = h(PW) \oplus h(y_u \| ID) \oplus h(x)$, $B = y_u \oplus h(PW)$, $V = h(ID \| PW) \oplus PW$ and $D = h(y_u \| ID)$; where y_u is a random value chosen by S in such a way that the value D must be unique for each user.

3) S stores $y_u \oplus x$ and $ID \oplus h(x)$ corresponding to D in its database.
4) S personalizes U's smart card with the parameters (N, B, V, h(.)).
5) $S \Rightarrow U$: {SC}.

Login Phase: U inserts his SC into a card reader to login on to S and keys in his ID^* and PW^*.

1) SC computes $V^* = h(ID^* \| PW^*) \oplus PW^*$ and checks if V^* ?= V.
2) If so, then the legality of U is verified; SC computes $y_u = B \oplus h(PW)$, $h(x) = N \oplus (PW) \oplus h(y_u \| ID)$, $CID = h(y_u \| ID) \oplus h(h(x) \| T_U)$, and $M = h(h(x) \| h(y_u) \| T_U)$.
3) $U \rightarrow S$:{CID, M, T_U}.

Verification and Session Key Agreement Phase:

1) On receiving the login request {CID, M, T_U}, S records the current timestamp T_S, and verifies if $(T_S - T_U) \leq \Delta T$. If so, S computes $D^* = CID \oplus h(h(x) \| T_U)$.
2) S finds D corresponding to D^* in its database and extracts y_u and ID from it.
3) S computes $M^* = h(h(x) \| h(y_u) \| T_U)$ and checks if M^* ?= M; the equivalence authenticates the legality of U and the login request is accepted else the connection is interrupted.
4) U and S agree on the common session key as $h(ID \| y_u \| h(x) \| T_U)$ to maintain the conversation privacy of the subsequent messages.

Password Change Phase: U can change his password without the help of S. The phase works as follows:

1) U inserts his SC into the card reader, and keys in his original identity ID^* and password PW^*.
2) SC computes $V^* = h(ID^* \| PW^*) \oplus PW^*$ and checks if V^* ?= V.
3) If so, then the legality of U is verified; SC prompts U to enter a new password PW_{NEW}, which user wants to change to
4) SC computes, $N_{NEW} = N \oplus h(PW) \oplus h(PW_{NEW})$, $B_{NEW} = B \oplus h(PW) \oplus h(PW_{NEW})$ and $V_{NEW} = h(ID \| PW_{NEW}) \oplus PW_{NEW}$
5) SC replaces N,B and V with N_{NEW}, B_{NEW} and V_{NEW}.

4.2 Security Analysis of Sood et al.'s Scheme

User Impersonation Attack: A legal but malicious user U_K can easily purchase a valid SC and can extract the value h(x) from his SC (such extraction is also admitted by Sood et al. themselves). In the following, we demonstrate that U_K can impersonate any valid user to communicate with S.

In a normal session, U_K first eavesdrops a login request {CID, M, T_U} which is sent by U. Next U_K obtains $h(y_u \| ID) = CID \oplus h(h(x) \| T_U)$.Then U_K mounts an offline guessing attack using $M = h(h(x) \| h(y_u) \| T_U)$ to guess y_u until he is successful. For U_K cost is not a concern in making such guess or extraction because computational cost of the method are very low; only hash operations and exclusive-or calculations are employed. Afterwards, whenever U_K wishes to login as U he computes $CID^* = h(y_u \| ID) \oplus h(h(x) \| T_K)$, $M^* = h(h(x) \| h(y_u) \| T_K)$ and sends {CID^*, M^*, T_K} with the current timestamp T_K. Thus the scheme does not resist user impersonation attack.

Session Key Computation Attack: This subsection presents that a legal but malicious user U_K can guess the secret agreed on session key between U and S. Similarly as in the previous subsection U_K possess h(x), $h(y_u \| ID)$ and y_u. Next he can guess ID using $h(y_u \| ID)$; then corresponding to the eavesdropped login request {CID, M, T_U} employed in obtaining $h(y_u \| ID)$ he successfully computes the session key $h(ID \| y_u \| h(x) \| T_U)$. From now, then U_K has access over all secret conversation between U and S.

Man in the Middle Attack: According to the above subsections we know that the user impersonation attack and the session key computation attack can be invoked by a legal but malicious user U_K. Therefore it is possible for U_K to construct a man in the middle attack.

Server's Resource Exhaustion Attack: In Sood et al.'s scheme S maintains a database to store entries corresponding users but there is no provision for its offsite back-up and periodic inspection to detect any malicious handling. Therefore if U_K having extracted h(x) from his own SC somehow successfully guesses the secret key x of S and luckily manages access to S's database. Next suppose U_K inserts an arbitrary triplet $\{D^* = h(y^* \| ID^*), y^* \oplus x, ID^* \oplus h(x)\}$ into the server's database, where y^* and ID^* are arbitrary values chosen by U_K of bits same as that of y_k and ID_K respectively. U_K then stores y^* and ID^* for future use. Corresponding to this triplet a valid login request $\{CID^* = h(y^* \| ID^*) \oplus h(h(x) \| T_K)$, $M^* = h(h(x) \| h(y^*) \| T_K), T_K\}$ can be computed by U_K. Consequently S's resources will be exhausted without getting payment of smart card/registration.

Leak of Verifier Attack: U_K can extract h(x) from his SC and hence can guess S's secret key x. Then being able to steal the verification table stored at S, he can use the stolen verifiers to impersonate a participant of the authentication protocol. For this U_K can extract y_u and ID; computes $D_U = h(y_u \| ID)$ corresponding to U. Consequently he can compute a valid login request {CID, M, T_K} and make fool of both U and S.

Insider Attack: It is never prudent to submit password to the server in plaintext form as done by Sood et al. because such practice invites insider attack. The insider of S gets the password of U and he can impersonate U to access other servers. Here insider of S has no need of even guessing the password of U. Sometimes it proves to be a catastrophic attack when the situation is related to money related transactions like online banking, e-commerce etc.

Common Drawback of both the Schemes: It is one of the requirements of smart card-based authentication schemes that in case of lost smart cards, there should be provision in the system for invalidating the further use of lost smart card, otherwise an attacker can impersonate valid registered user [4]. Through keeping record of valid card identifier of each registered user, the authentication system can distinguish the valid card from the invalid one. Unfortunately Sood et al.'s scheme and Gao-Tu's scheme overlook this feature and there is no prerequisite to revoke the lost smart card. This flaw would become more catastrophic if an adversary has got lost smart card and has revealed password of a valid user by any means to login into the system for performing secure transaction, e.g. online banking and e-commerce, etc. Thus the two schemes fail to provide the important feature of smart card-based authentication for revoking the lost smart cards without changing the user's identities [4].

5 Conclusion

Smart card-based user authentication technology has been widely deployed in various kinds of applications, such as remote host login, withdrawals from automated cash dispensers, and physical entry to restricted areas. In 2008, Gao-Tu and later in 2010, Sood et al., found vulnerabilities in Liao et al.'s scheme; they independently proposed its security enhanced versions showing immunity against pointed out attacks. However, we have demonstrated that scheme proposed by Gao-Tu suffers from serious impersonation problems; and Sood et al.'s scheme also fails to resist attacks it claimed to. For this reason neither Gao-Tu's scheme nor Sood et al.'s scheme is secure for practical application. It is important that security engineers should be made aware of this, if they are responsible for the design and development of smart card-based user authentication systems.

References

1. Chang, C.C., Wu, T.C.: Remote Password Authentication with Smart Cards. IEE Proceedings-E 138(3), 165–168 (1991)
2. Hwang, M.S., Li, L.H.: A New Remote User Authentication Scheme using Smart Cards. IEEE Trans. on Cons. Elect. 46(1), 28–30 (2000)
3. Juang, W.S.: Efficient Password Authenticated Key Agreement using Smart Cards. Computers and Security 23(2), 167–173 (2004)
4. Fan, C.I., Chan, Y.C., Zhang, Z.K.: Robust Remote Authentication Scheme with Smart Cards. Computers and Security 24(8), 619–628 (2005)
5. Das, M.L., Saxena, A., Gulati, V.P.: A dynamic ID-based remote user authentication scheme. IEEE Trans. on Cons. Elect. 50(2), 629–631 (2004)
6. Liao, I.E., Lee, C.C., Hwang, M.S.: Security enhancement for a dynamic ID-based remote user authentication scheme. In: International Conference on Next Generation Web Services Practices, pp. 437–440. IEEE CS Press, Los Alamitos (2005)
7. Chien, H.Y., Chen, C.H.: A Remote Authentication Scheme Preserving User Anonymity. In: Proc. 19th Inter. Conf. Advd. Infmn. Netw. and Applns.,Taipei, Taiwan, vol. 2, pp. 245–248 (2005)
8. Zhang, X., Feng, Q.Y., Li, M.: A Modified Dynamic ID-Based Remote User Authentication Scheme. In: Proc. of Inter. Conf. on Communcs. Circuits and Syst., vol. 3, pp. 1602–1604 (2006)
9. Misbahuddin, M., Ahmed, M.A., Rao, A.A., Bindu, C.S., Khan, M.A.M.: A Novel Dynamic-ID Based Remote User Authentication Scheme. In: Annual India Conf., pp. 1–5 (2006)
10. Liao, I.E., Lee, C.C., Hwang, M.S.: A Password Authentication Scheme over Insecure Networks. Journal of Computer and System Sciences 72, 727–740 (2006)
11. Gao, Z.X., Tu, Y.Q.: An Improvement of a Dynamic ID-Based Remote User Authentication Scheme with Smart Card. In: Proc. Of the 7thWorld Congress on Intelligent Control and Automation, pp. 4562–4567 (2008)

12. Sood, S.K., Sarjee, A.K., Singh, K.: An Improvement of Liao et al.'s Authentication Scheme using Smart Card. In: IEEE 2nd International Advance Computing Conf., pp. 240–245 (2010)
13. Ku, W.C., Chang, S.T.: Impersonation Attack On A Dynamic ID-Based Remote User Authentication Scheme using Smart Cards. IEICE Transactions on Communications E88-B(5), 2165–2167 (2005)
14. Manoj, K.: On the Security Vulnerabilities of a Hash Based Strong Password Authentication Scheme, Cryptology ePrint Archive: a publication of The International Association for Cryptologic Research (IACR), Santa Rosa Administrative Center, University of California, Santa Barbara, CA, 93106-6120, USA, Report (2009), http://www.eprint.iacr.org/2009/560
15. Manoj, K., Gupta, M.K., Kumari, S.: A Remote Login Authentication Scheme with Smart Cards Based on Unit Sphere. Indian Journal of Computer Science and Engineering 11(3), 192–198 (2010) ISSN: 0976-5166
16. Manoj, K.: Security Vulnerabilities of a Novel Remote User Authentication Scheme Using Smart Card Based on ECDLP. Contemporary Computing, Communications in Computer and Information Science 95(5), 252–259 (2010) ISSN: 1865-0929
17. Manoj, K.: An Enhanced remote user authentication scheme with smart cards. International Journal of Network Security 10(3) (2010) ISSN 1816-353X (Print) ISSN 1816-3548 (Online)
18. Manoj, K.: A New Secure Remote User Authentication Scheme with Smart Cards. International Journal of Network Security 11(2), 88–93 (2010) ISSN 1816-353X (Print), ISSN 1816-3548(Online)
19. Manoj, K.: New Remote User Authentication Scheme with Smart Cards. IEEE Trans. Consumer Electronic 50(2), 597–600 (2004) ISSN: 0098-3063
20. Manoj, K.: Some Remarks on a Remote User Authentication Scheme Using Smart Cards with Forward Secrecy. IEEE Trans. Consumer Electronic 50(2), 615–618 (2004) ISSN: 0098-3063
21. Yeh, K.-H., et al.: Two Robust Remote User Authentication Protocols Using Smart Cards. Journal of System and Software (2010), doi:10.1016/j.jss.2010.07.062
22. Wang, Y.Y., Liu, J.Y., Xiao, F.X., Dan, J.: A More Efficient and Secure Dynamic ID-based Remote User Authentication Scheme. Computer Communications 32, 583–585 (2009)
23. Hsiang, H.C., Shih, W.K.: Weaknesses and Improvements of the Yoon–Ryu–Yoo Remote User Authentication Scheme Using Smart Cards. Computer Communications 32, 649–652 (2009)

Designing QoS Based Service Discovery as a Fuzzy Expert System

Rajni Mohana and Deepak Dahiya

Dept. of Computer Science and Engineering, JUIT Waknaghat, Solan, H.P, India
rajni.mohana@juit.ac.in, deepak.dahiya@juit.ac.in

Abstract. Service Discovery is an important element which requires finding a set of suitable webservice candidates faster for the service requester among those published by the service provider. Among large number of functionally-equivalent, it is difficult for users to choose a best service to be invoked. This paper proposes a new webservice reference platform which has service discovery element behaving as a fuzzy expert system. The proposed webservice reference model makes the service discovery element automatic .

Keywords: Webservices, Fuzzy Clustering, Fuzzy Expert System, PSO, Quality of Service.

1 Introduction

Service oriented architecture is a reusable library of services for common business and IT function where the components are service users and/or service providers. However, what essentially characterizes an SOA is the webservice [1]. The basic webservices architecture consists of specifications (SOAP, WSDL, and UDDI) service that supports the interaction of a webservice requester with a webservice provider and the potential discovery of the webservice description.

2 Related Work

A fuzzy expert system is application software that performs a task that would be performed by a human expert [4]. It simply uses a collection of fuzzy membership functions and rules, instead of Boolean logic, to reason about data. There are many ways presented by the earlier researchers to rank the services [2][3][4][5].

3 Proposed Webservice Reference Model

The proposed webservice model is a fuzzy expert system which is adaptive in nature. It trains itself according to the dataset and generates a rule base, which will be used by the reference engine to rank the webservices.

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 533–534, 2011.

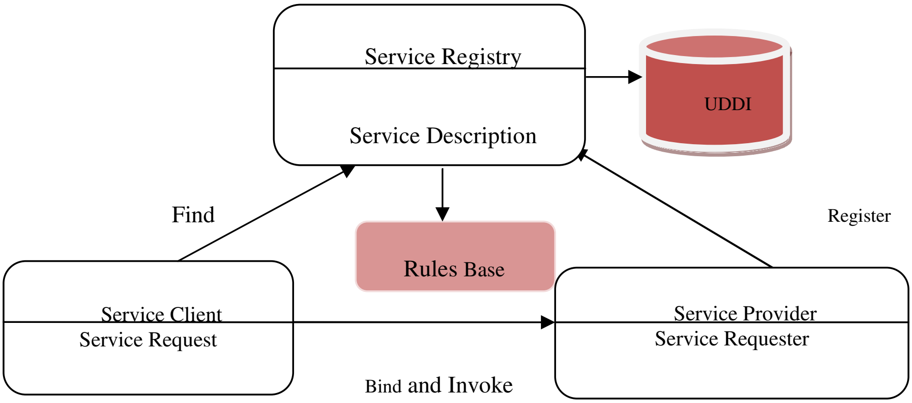


Fig. 1. Proposed webservice reference model

The approach described uses an algorithm to generate the rules based on Fuzzy clustering and Particle Swarm Optimization (PSO) [4][5].

4 Conclusion

The proposed web reference model is an expert system that can automatically rank the new webservice that is registered by a service provider. This architecture is adaptive in nature as any change in QoS of a webservice will change the rules generated by the algorithm. The rule base is generated using fuzzy clustering and PSO. The number of rules is also optimized by PSO.

References

1. o'Brien, L., Merson, P., Bass, L.: Quality Attributes for Service-Oriented Architectures. In: Proc. of the International Workshop on Systems Development in SOA Environments SDSOA 2007(2007), doi:10.1109/SDSOA.2007.10
2. Yu, Q., Liu, X., Bouguettaya, A., Medjahed, B.: Deploying and managing Webservices: issues, solutions, and directions. The VLDB Journal — The International Journal on Very Large Data Bases 17(3), 537–572 (2008)
3. Tran, V.X., Tsuji, H.: QoS based Ranking for Web Services: Fuzzy Approaches. In: Proc. of 4th International Conference on Next Generation Web Services Practices. IEEE, Los Alamitos (2008) doi: 10.1109
4. Zadeh, L.A.: Fuzzy Sets. J. of Information and Control 8(3), 338–353 (1965)
5. Mohana, R., Dahiya, D.: Optimized Service Discovery using QoS based Ranking: A Fuzzy Clustering and Particle Swarm Optimization Approach. In: To be published in the IEEE Proceedings of the 35th IEEE Computer Software and Applications Conference (IEEE COMPSAC 2011), Munich, Germany (2011)

Image Interpolation Using E-spline

Ram Bichar Singh[1], Kiran Singh[2], Kumud Yadav[2], and Amrita Bhatnagar[2]

[1] Radha Raman Institute of Technology & Science, Bhopal
[2] Skyline Institute of Engg & Technology, Gr.Noida (U.P)

Abstract. This paper introduces a new fast method for the calculation of exponential B-splines sample at regular intervals. This new method is fast and it also considered polynomial spline as special case. This algorithm is based on a combination of FIR and IIR filters which enables a fast decomposition and reconstruction of a signal. In this paper we have tried to get the interpolation function which uses the symmetric exponential functions of 4th order. We are considering the real part of these functions which is used for interpolation of real signals corresponding to different exponential parameter that leads to less band limited signals when they are compared with polynomial B-spline counterparts. These characteristics were verified with 1-D and 2-D examples. We are also going through all the interpolation methods which are already in use.

Keywords: Nearest Neighbour Interploation,Cubic Interpolation,Kernel of B-spline approximation.

1 Introduction

B-splines sometimes are referred to as cubic splines while cubic interpolation is also known as cubic convolution, high resolution spline interpolation and bi-cubic spline interpolation. Here, we propose E-spline method for image expansion and compare to other methods such as Linear and cubic spline. Here, we propose E-spline method for image expansion and compare to other methods such as Linear and cubic spline.The goal of this study is not to determine overall best method but to present a comprehensive catalogue of interpolation methods using E-spline, to define general properties and requirements of E-spline techniques.

1.1 Interpolation Methodologies

In this research we use various interpolation methodologies as:

Ideal interpolation, Nearest Neighbor Interpolation, Linear Interpolation, Quadratic Approximation, Quadratic Interpolation, B-spline Approximation, B-spline Interpolation, Cubic Interpolation, Gaussian Interpolation.

2 Results

Here we perform on lena image (64x64). After performing, E-spline interpolation with varying α .We get the smooth area corresponding to the face of Lena and higher

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 535–537, 2011.

frequency parts which are the back ground. Interpolation versus Approximation: in this E-spline methods are well suited for the images containing high frequency components. Gaussian kernels are not suited for this method. These kernels have been compared on various images of Lena. In each case, the efficiency and accuracy of a particular interpolation technique was evaluated by analyzing its Fourier properties, visual quality and run time measurement.

2.1 Runtime Measurements

The runtime of the various interpolation schemes were measured on the standard machine. Sources have been compiled using MATLAB 6.5. The rotation is quite time consuming in MATLAB environment. It shows that simple interpolation methods such as nearest neighbor ,linear and 2x2 cubic interpolation are fairly fast and requires less time than the rotations of pixel coordinates. Gaussian interpolation required more time due to the evaluation of the exponential function necessary to determine weights. Here are the results of all interpolated image and we compare all interpolated image to original one. The resultant image is shown below:

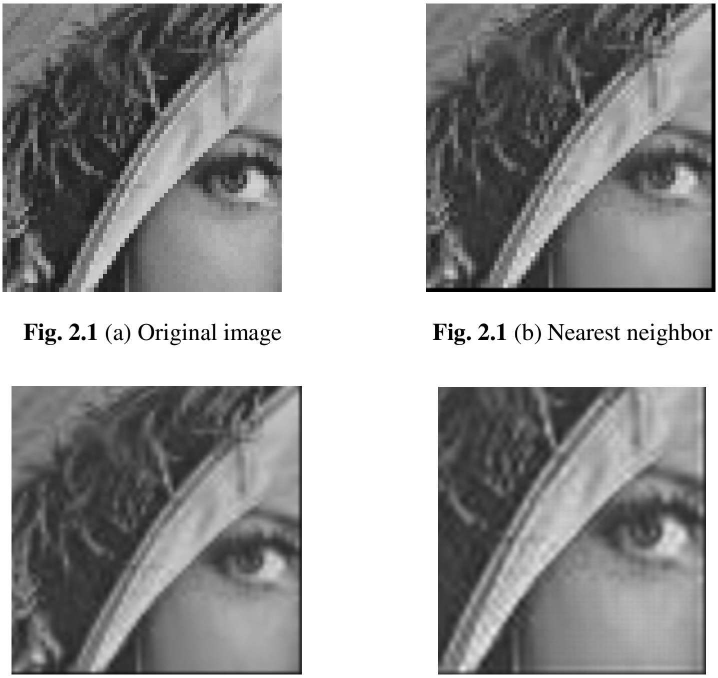


Fig. 2.1 (a) Original image

Fig. 2.1 (b) Nearest neighbor

Fig. 2.1 (c) Linear image

Fig. 2.1 (d) B-spline image

Fig 2.1 Interpolated image of Lena (64x64) using various types of interpolation methods.

3 Conclusion and Future Work

This paper is focused on interpolation methods using E-spline. The method is fast for calculation of E-spline (where the calculation of B-spline is a particular case). When complex parameters are used in exponential functions, trigonometric spline are obtained and here we are dealing with only real parts of the signal. Less band limited functions can be achieved by using the real par of this later functions and comparing them with the polynomial splines counterparts. Although more amount of computation is required, it is specially used for images containing high frequency components. As demonstrated by codes in this paper, the interpolation methods using E-spline is easy to implement. The E-spline is general case of B-spline polynomials and performs better interpolation for images containing high frequency parts.

Since we have a good result for the interpolation using E-spline techniques, the further study on interpolation should be carry on in the future. Interpolation of images is popular problem until yet but now we have to find kernel that enhances high frequency parts. If we get the required result then we go for L-spline that is general case of E-spline. Future work will address techniques to get better result in interpolation methods by finding the adaptive method for the appropriate image.

References

1. Unser, M.: Splines: A perfect fit for signal and image processing. IEEE Signal Processing Magazine 16(6), 22–38 (1999)
2. Lehmann, T.M., Gonner, C., Spitzer, K.: Survey: Interpolation Methods in Medical Image Processing. IEEE Transactions on Medical Imaging 18(11) (November 1999)
3. Asahi, T., Ichige, K., Ishlii, R.: Fast Computation of Exponential Splines. Proceedings of the IEEE
4. Unser, M., Aldroubi, A., Eden, M.: Fast B-spline transforms for continuous image representation and interpolation. IEEE Trans. On Pattern Anal. & Machine Intell. 13(3), 277–285 (1991)
5. Meijering, E.: A Chronology of Interpolation: From Ancient Astronomy to Modern Signal and Image Processing. Proceedings of the IEEE 90(3) (March 2002)
6. Unser, M., Blu, T.: Cardinal Exponential Splines: Part I- Theory and Filtering Algorithms. IEEE Transactions on Signal Processing 53(4) (April 2005)
7. Unser, M.: Cardinal Exponential Splines: Part II- Think Analog, Act Digital. IEEE Transactions on Signal Processing 53(4) (April 2005)
8. Mcartin, B.J.: Compuatation of Exponential Splines. SIAM J. Sci STAT. Comput. 11(2), 242–262 (1990)
9. Miklos, P.: Image Interpolation techniques

A Multi-purpose Density Based Clustering Framework

Navneet Goyal, Poonam Goyal, Mayank P. Mohta,
Aman Neelappa, and K. Venkatramaiah

Department of Computer Science & Information Systems, BITS, Pilani, India

Abstract. In this paper, we present a multi-purpose density-based clustering framework. The framework is based on a novel cluster merging algorithm which can efficiently merge two sets of DBSCAN clusters using the concept of intersection points. It is necessary and sufficient to process just the intersection points to merge clusters correctly. The framework allows for clustering data incrementally, parallelizing the DBSCAN algorithm for clustering large data sets and can be extended for clustering streaming data. The framework allows us to see the clustering patterns of the new data points separately. Results presented in the paper establish the efficiency of the proposed incremental clustering algorithm in comparison to IncrementalDBSCAN algorithm. Our incremental algorithm is capable of adding points in bulk, whereas IncrementalDBSCAN adds points, one at a time.

Keywords: Incremental clustering, DBSCAN, cluster merging algorithm.

1 Introduction

Density-based clustering is used for discovering clusters of arbitrary shapes. In this paper, we propose a multi-purpose framework for density-based clustering. At the core of the proposed framework is an algorithm for merging two sets of existing DBSCAN [1] clusters based on the concept of *intersection points*. R-tree [2] is the main data structure used and we have innovatively exploited its structure to find intersection points. The framework allows for incremental clustering [3,4] through which new data points are clustered separately and merged with existing set of clusters. We have shown that it is necessary and sufficient to process just the intersection points to merge clusters correctly. The framework can also be used for parallelizing the DBSCAN algorithm and for clustering streaming data using the sliding and landmark window models [5]. The framework provides the flexibility to cluster data over user-defined time windows. The methodology adopted in the paper clusters data belonging to different time intervals or sources, separately. The cluster merging algorithm can then be used to merge clusters based on user requirements.

2 The Proposed Framework

The steps involved in the framework are as follows. We first cluster separately the new data points to be added using the DBSCAN algorithm. We then find the overlap

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 538–540, 2011.

between the old clusters and the new clusters in terms of intersection points. An object *p*∈ *D'* (set of new data points) is an intersection point if ∃ at least one object ∈ *D* (set of old data points) in the ε-neighborhood of *p*. The intersection points are then processed to merge these clusters. The cluster merging algorithm has two constituent algorithms namely, the find intersection point algorithm and the process intersection point algorithm. After merging of clusters, the R-trees holding the old and new data points are also merged. The framework is then ready to receive the next set of new points.

3 Results and Discussions

The proposed incremental clustering algorithm produces the same clusters as IncrementalDBSCAN [3]. The efficiency of the proposed algorithm is compared with IncrementalDBSCAN, which adds new points, one at a time, whereas our algorithm adds a cluster at a time. The metric used for comparison is the number of R-tree nodes (both internal and external) visited during the entire process. Results are presented for 2D synthetic data which has 4500 data points having 3 clusters. Five new datasets are generated with 360 data points each having different percentage of intersection points varying from 0 to 83. In the figure, the old dataset is shown in light grey and the new dataset in dark grey. The relative improvement is plotted against percentage of intersection points. It can be seen that even for as high a percentage as 83, our algorithm performs better than IncrementalDBSCAN. With a decrease in intersection points, the improvement factor of our algorithm increases. The result for the data set with zero intersection point has not been shown as the relative improvement is very high.

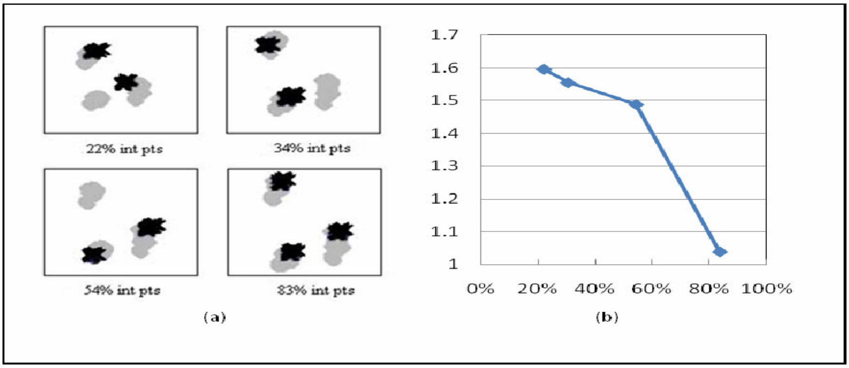


Fig. 1. Relative Improvement over IncrementalDBSCAN

References

1. Ester, M., Kriegel, H.-P., Sander, J., Xu, X.: A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In: Proc. of 2nd Int. Conf. on Knowledge Discovery and Data Mining, Portland, pp. 226–231. AAAI Press, Menlo Park (1996)
2. Guttman, A.: R-Trees: A Dynamic Index Structure for Spatial Searching. In: ACM SIGMOD, Boston, Massachusetts, pp. 47–57 (1984)
3. Ester, M., Kriegel, H.P., Sander, J., Wimmer, M., Xu, X.: Incremental Clustering for Mining in a Data Warehousing Environment. In: VLDB, New York, pp. 323–333 (1998)
4. Rehman, S., Khan, M.N.A.: An Incremental Density-Based Clustering Technique for Large Datasets. Advances in Soft Computing, vol. 85, pp. 3–11 (2010)
5. Yang, D., Rundensteiner, E.A., Ward, M.O.: Neighbor-Based Pattern Detection for Windows over Streaming Data. In: ACM EDBT 2009, Saint-Petersburg, pp. 529–540 (2009)

Pruning of Rule Base of a Neural Fuzzy Inference Network

Smarti Reel[1] and Ashok Kumar Goel[2]

[1] Electronics and Communication Engineering Department,
Thapar University, Patiala – 147001, India
kotwalsmarti@gmail.com
[2] M M Group of Institutions,
V.P.O Sadopur, Distt. Ambala – 134003, India
ashokkgoel@rediffmail.com

Abstract. In this work, Neural Fuzzy Inference Network (NFIN) controller is implemented that has a number of membership functions and parameters that are tuned using Genetic Algorithms. The number of rules used to define the Neuro-Fuzzy controller is then pruned. Pruning is utilized effectively to eliminate irrelevant rules in the rule base, thus keeping only the relevant rules. Pruning is performed at various threshold levels without affecting the system performance. This methodology is implemented for Water Bath System and analysis has been carried out to investigate the effect of pruning using a multi-step reference input signal. From the results, it is concluded that reasonably good performance of controller can be obtained with lesser number of rules, thus, reducing the computational complexity of the network.

Keywords: Neural fuzzy inference network, pruning, artificial intelligence, fuzzy logic, neural network.

1 Introduction

Fuzzy logic has become a common buzzword in machine control. It provides remarkably simple way to draw definite conclusions from vague, ambiguous or imprecise information [1]. The basic building block of a fuzzy logic control system is set of fuzzy IF-THEN rules that approximate a functional mapping. A neuro-fuzzy network is proposed where all the parameters of the network and the IF-THEN rule base are evolved using GA [2]. The Neuro-Fuzzy network [3] has been first implemented on Water Bath Temperature Control System. The total numbers of membership functions corresponding to each input are seven. Thus, the total numbers of membership functions considered in this network are 49. Such increase in the number of membership functions increases the resolution and gives and refined control.

2 Pruning of Rule Base

In the network proposed by [3], complete set of rules that can be formed using all the membership functions of each input are taken. Practically, all rules in the rule base do

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 541–542, 2011.

not fire. In fact some rules correspond to such combinations that may not be even practically viable. Such impractical rule combinations and the ones having low firing strength can be easily removed from the rule database without much effect on the performance of the network. The reduction in number of rules and the performance of the NFC for the Simulated Water Bath Temperature Control System have been analyzed.

3 Results and Discussions

Pruning has been performed at 16 pruning levels ranging from 0.01 to 0.60. As per the threshold level, all the rules in the rule base whose output is greater than the set level are maintained and all the rules whose firing strength is lesser then the defined pruning level are pruned and thus, the number of rules decreases. The results show that as the pruning level increases, the numbers of rules decrease and corresponding Mean Squared Error (MSE) increases. For each pruning level, the output of the Water Bath System is obtained for a multi-set point reference input. The performance of the output is analyzed in terms of MSE, Integral Sum Squared Error (ISSE), number of rules used, overshoots/undershoots and control signal.

4 Conclusions

From the results it is concluded that pruning is utilized effectively to eliminate irrelevant rules in the rule base. Pruning is implemented successfully and total number of rules is reduced from initially 49 to finally 9 without much affecting the system performance, a reduction of 82.0 %. It is further concluded that pruning also decreases the system complexity to a great extent.

References

1. Chen, J.-Q., Xi, Y.-G.: Non-linear System Modelling by Competitive Learning and Adaptive Fuzzy Inference System. IEEE Transactions on Systems, Man and Cybernetics, Part C Applications & Reviews 28(2), 231–238 (1998)
2. Farag, W.A., Quintana, V.H., Lambart, T.G.: A Genetic Based Neuro-Fuzzy Approach for Modelling and Control of Dynamical Systems. IEEE Transactions on Neural Networks 9(5), 756–757 (1998)
3. Goel, A.K., Saxena, S.C., Bhanot, S.: A Genetic based Neuro-fuzzy controller for Thermal Processes. Journal of Computer Science and Technology 5(1), 37–43 (2005)

Integrating Aspects and Reusable Components: An Archetype Driven Methodology

Rachit Mohan Garg and Deepak Dahiya

Dept. of Computer Science and Engineering, Jaypee University of Information Technology Waknaghat, Solan, H.P, India
rachit.mohan.garg@gmail.com, deepak.dahiya@juit.ac.in

Abstract. The proposed work focuses on developing a methodology that promotes software development by partitioning the whole system into different independent components and aspects. This facilitates component reuse along with the ease of modeling the components separately and emphasizing on the concerns that the widely used OOP paradigm has failed to address. Identification of reusable components is carried out using the hybrid methodology and aspects are identified by the domain experts. Along with the components the platform independent models and aspects developed are stored in separate repositories so as to be used in development of other software of similar requirements and basic structure.

Keywords: Model Driven Architecture; System design; Component Based Development; Aspect Oriented Development.

1 Introduction

To survive the cut throat world of competition organizations are trying to develop the software in a cost effective way so they are using more of the available reusable components. The work presented in this paper describes a design methodology which will help in creating highly reliable, adaptable software products in a timely fashion.

A brief overview of the proposed methodology uses the concept of model driven architecture (MDA) [1], components [2, 3], UML models [4, 5] and aspects [6, 7].

2 Related Work

2.1 Component Identification Techniques

Identification of reusable components in software is one the most important task of the component based software development. Many approaches for component identification are proposed [8, 9] but in the end it is the work of the experts to separate out the components manually as these approaches only provide the knowledge of the component without actually separating them out.

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 543–545, 2011.

2.2 Integration of Aspects with Archetype Driven Development

The separation of concerns from the core business logic is a standard practice that helps in the development of better software that is free from the problem of scattering and tangling [10, 11].

3 Proposed Methodology

The underlying proposed methodology provides a repository based architecture in which the modeled PIM is stored so that it can be used in future project with the similar type of requirements and specifications along with the basic structure [12].

The different phases of the methodology include investigation phase, modeling phase, component identification phase, artifact generation phase and integration phase [13].

4 Results and Significance

The significance of the proposed methodology lies in the form of numerous advantages that are gained over the other prevailing approaches. The advantages gained by the proposed methodology are:

- Aspect Modeling
- Developer Overhead Reduction
- Early Error Detection
- Reusability of the Models
- Reusability of the Components
- Reusability of the Aspects

5 Conclusion

Using Archetype driven methodology models are developed and later transformed to generate the code artifacts for the specific platform. Thus developer only lays emphasis on writing the code for the specific functionality and thereby reduces a considerable amount of burden. This methodology promotes reuse of the components, PIM and aspects in other products of similar domain and basic structure.

References

1. Object Management Group: MDA Guide Version 1.0.1.?, `http://www.omg.org/mda/` (last accessed: January 21, 2010)
2. Wu, Y., Offutt, J.: Maintaining Evolving Component-Based Software with UML. In: Proc. of the 7th IEEE European Conference on Software Maintenance and Reengineering (2003)
3. Cai, X., et al.: Component-Based Software Engineering: Technologies, Development Frameworks, and Quality Assurance Schemes. In: Proc. of the 7th IEEE Asia-Pacific Software Engineering Conference (2000)

4. Fuentes-Fernández, L., Vallecillo-Moreno, A.: An Introduction to UML Profiles. Journal of Informatics Professional 5, 6–13 (2004)
5. Dr. Marko Boger, Elizabeth Graham, Matthias Köster.: Poseidon for UML, `http://www.gentleware.com/fileadmin/media/pdfs/userguides/PoseidonUsersGuide.pdf` (last accessed: November 5, 2010)
6. Elrad, T., et al.: Special Issue on Aspect-Oriented Programming. Communications of the ACM 44 (2001)
7. Zhang, J., Chen, Y., Li, H., Liu, G.: Research on Aspect-Oriented Modeling in the Framework of MDA. In: Proc. of the 2nd IEEE International Conference on Computer Science and Information Technology, pp.108–111 (2009)
8. Rodrigues, N.F., Barbosa, L.S.: Componesnt Identification through Program Slicing. Electronic Notes in Theoretical Computer Science (2005)
9. Fan-Chao, M., Den-Chen, Z., Xiao-Fei, X.: Business Component Identification of Enterprise Information System: A hierarchical clustering method. In: Proc. of the 2005 IEEE Int. Conf. on e-Business Engineering, pp. 473–480 (2005)
10. Fuentes-Fernández, L., Vallecillo-Moreno, A.: An Introduction to UML Profiles. Journal of Informatics Professional 5(2), 6–13 (2004)
11. Simmonds, D.M., Reddy, Y.R., Song, E., Grant, E.: A Comparison of Aspect-Oriented Approaches to Model Driven Engineering. In: Proc. of Conference on Software Engineering Research and Practice, pp. 327–333 (2009)
12. Shahmohammadia, G.R., Jalilia, S., Hasheminejada, S.M.H.: Identification of System Software Components Using Clustering Approach. Journal of Object Technology (2010)
13. R.M. Garg, D. Dahiya: An Aspect Oriented Component Based Model Driven Development. Paper accepted for publication in the Springer Series in Communications in Computer and Information Science (CCIS) of the 2nd International Conference on Software Engineering and Computer Systems (ICSECS 2011). University Malaysia, Pahang (2011)

Resource Provisioning for Grid: A Policy Perspective

Rajni Aron and Inderveer Chana

Computer Science & Engineering Department,
Thapar University, Patiala 147004, India

Abstract. To enhance the efficiency of grid resource management systems, resource provisioning is required. For efficient resource provisioning, a policy based resource provisioning framework needs to developed. This paper presents Resource Provisioning framework and discusses the resource policy for better resource utilization and customer satisfaction.

Keywords: Resource Provisioning, Quality of Service, Grid Computing.

1 Introduction

Grid computing provides the facility of resource sharing in multi-institutional virtual organizations [1]. To manage the heterogenous and dynamic nature of the grid resources, resource provisioning should be done in an effective manner.

2 Motivation

Resource Provisioning is particularly useful in resource management systems as it allows the users and providers to access the specified resources according to availability of the resources in virtual organizations. Efficient resource provisioning can be done only if resource provisioning framework has been developed for grid that works on the laid down policy.

3 Resource Provisioning Policy

Figure 1 illustrates the resource provisioning framework. First of all, authenticated users will try to access the resource through grid portal for execution of application. Broker will collect the information about the resources and job status. GRAM will communicate with the Resource Provisioning Policy Manager (RPPM). RPPM will take the information about policy which are stored in the policy repository. Policy Decision Point (PDP) and Policy Enforcement Point(PEP) are logical entities which make and enforce policy decisions. Resource Manager (RM) checks for availability of the resources according to policy conditions and then provisions the resources to user's application. The task of scheduling will then be performed and the result will be again sent back to the user.

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 546–547, 2011.

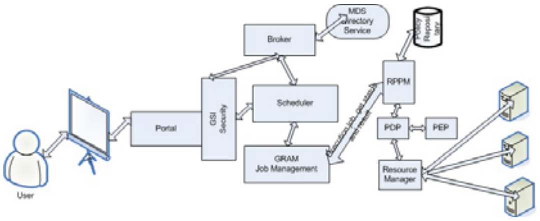


Fig. 1. Resource Provisioning framework

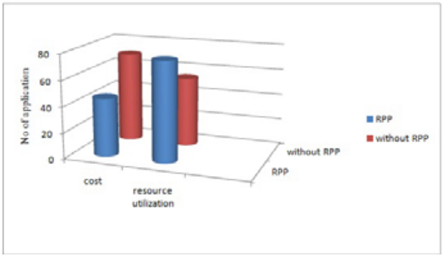


Fig. 2. cost and resource utilization of the resources for the execution of the application

4 Results and Discussion

The performance evaluation of resource provisioning framework and resource provisioning policy is performed through the simulation by using gridsim [2]. We have done the experiment by taking cost as quality of service parameter. Cost is defined as a unit of resources that are consumed by the user for execution of their application and should be considered at the time of resource provisioning. Figure 2, shows that the cost will decrease as well resource utilization will increase with the resource provisiong.

5 Conclusion

In this paper, we have proposed resource provisioning framework as a policy perspective for grid environment. Resource Provisioning policy can assist organizations in enhancing customer satisfaction and contribute directly to the company growth and institutional's progress.

References

1. Foster, I., Kesselman, C.: The Grid: Blueprint for a Future Computing Infrastructure. Morgan Kaufmann Publishers, USA (2004)
2. Buyya, R., Murshed, M.: GridSim; A toolkit for the modeling and simulation of distributed management and scheduling for grid computing (2002)

Updated Version of PHYTODB, the First Data Warehousing and Mining Web Server for Phytoplasma Research

R. Manimekalai, P. Anoop Raj, O.M. Roshna, Anil Paul, and George V. Thomas

Department of Biotechnology, Agribioinformatics center,
Central Plantation Crops Research Institute, Kasaragod – 67 1124, Kerala, India
rmanimekalai@rediffmail.com,
apposklr@gmail.com, agribioros@gmail.com,
paulanilpaul@gmail.com, georgevthomas@yahoo.com

1 Introduction

PHYTODB contains a repository of phytoplasma genes and proteins. It provides a unified gateway to store, search, retrieve, update information about phytoplasma and computational resources for the analysis of nucleotide and aminoacid sequence data of phytoplasma. Server facilitates to differentiate and classify new phytoplasma for taxonomic purposes. PHYTODB database was updated by dividing the whole resources into two domains: *DataBanks* and *Tools*. *DataBanks* serve as the storage device of all information. Functional characterization of genes and protein are done. Updated Groupidentifier tool by rearrangement of RFLP classification scheme of phytoplasma and possibilities 6 new groups based on the new tool. PhytoDB can be obtained through http://220.227.88.253/phytodb/.

2 Design of Database

PHYTODB has 3-tier organization where Web 2.0 technologies like AJAX provide high quality dynamic interfaces. Apache web server provides full range of Web server features to hand over the client requests. DHTML and JavaScript are used for developing user interfaces with the help of Web 2.0 technology. GroupIdentifier tool for taxonomic classification of phytoplasma was developed using Java Servlets, Java Server Pages and BioJava 1.5. Other tools are developed using PHP for querying and response management and Perl CGI scripts for result generation. MySQL is the database server used for data and sequence storage. Apache-Tomcat server is used for GroupIdentifier tool and Apache web server for the remaining tools. NCBI (National Centre for Biotechnology Information) stands for the primary data source from where the phytoplasma nucleotide and protein sequences were retrieved.

3 Architecture

The web server resources are categorized into two domains: *DataBanks* and *Tools.* The DataBanks functions as repository of all phytoplasma information assembled in distinct sections facilitating easy data retrieval. Tools domain contains Hlogs,

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 548–550, 2011.

MSalign, PhyloCass and GroupIdentifier. GroupIdentifier achieves phytoplasma group determination based on similarity calculation.

3.1 Databanks

PHYTODB has 5 different databanks that comprises of DBgene, ProtB, 16Sr Groups, G-Nome and electronic literature service providing important research works done worldwide on phytoplasma (E-Lite). The functional categorization of all phytoplasma genes and proteins have performed 1600 phytoplasma genes and 2188 proteins entries are stored in '*DBgene*' and '*ProtB*' databanks respectively. 16S rRNA gene region of 775 various phytoplasma species are organized in taxonomic groups according to classification scheme recognized by Phytoplasma Taxonomy Group of the International Research Program on Comparative Mycoplasmology [1]. in '16Sr Groups' cluster. *G-Nome* contains information regarding phytoplasma whole genome sequencing projects of various phytoplasma.

3.2 Tools

The domain has embedded with 3 sequence analysis tools (Hlogs, Msalign and PhyoClass) and a phytoplasma taxonomic group identification tool. *Hlogs* determine homologues, based on BLASTN algorithm [2]. The output of the similarity search has taxonomic group tag in sequence description line. *MSalign* produces biologically meaningful multiple sequence alignments of divergent sequences. It calculates the best match for the selected sequences. User can perform multiple sequence alignment with our 16Sr group database by entering a single 16S sequence as input. *PhyloClass* constructs phylogenetic tree from molecular sequence data. Algorithm behind the tool performs multiple sequence alignment and drawn a phylogenetic tree. The output tree file can be viewed through any tree viewer program. *GroupIdentifier* performs the group identification of phytoplasma. It represents an attempt of phylogenetic classification based on the most conserved 16S rRNA gene sequence of phytoplasma genome.

In our classification scheme 36 distinct phytoplasma groups are exist based on similarity calculation between sequences. The group system is developed based on similarity cutoff value >98.64%, which is verified through MSA and phylogeny analysis of nearly full length (~1246nts) 16S rDNA sequences of ~200 phytoplasma species. At this cutoff each phytoplasma groups can be differentiated clearly.

4 Conclusions

PHYTODB, publicly accessible resource of phytoplasma, developed by various computer languages and software such as Web 2.0 technologies, DHTML, JavaScript and Apache web servers. Various bioinformatics tools are incorporated using the Perl CGI scripts for sequence analysis. The database embraces fundamental information on phytoplasma. Tools embedded are useful for homologues search, multiple sequence alignment, phylogenetic analysis and 16S rRNA based group identification of new phytoplasma.

References

1. IRPCM Phytoplasma/ Spiroplasma Working Team – Phytoplasma Taxonomy Group, -‘Candidatus Phytoplasma’, a taxon for the wall-less, non-helical prokaryotes that colonize plant phloem and insects. Int. J. Syst. Evol. Microbiol. 54, 1243–1255 (2004)
2. Altschul, S.F., Gish, W., Miller, W., Myers, E.W., Lipman, D.J.: Basic local alignment search tool. J. Mol. Biol. 215, 403–410 (1990)

De-duplication in File Sharing Network

Divakar Yadav, Deepali Dani, and Preeti Kumari

Jaypee Institute of Information Technology,
A-10, Sector-62, Noida (India)
divakar.yadav@jiit.ac.in, deepali.achieve@gmail.com,
kpreeti257@gmail.com

Abstract. Redundant data transfer over a network is one of the important reasons of traffic congestion today. In this paper, we proposed an efficient and secure file sharing model using de-duplication technology to resolve it. A file sharing based de-duplication system reduce bandwidth and storage at both client and server machine. It does not download duplicate blocks that already have been downloaded. To achieve the security of client data, three-tier architecture is proposed in this work. For this purpose SHA-1 hash function is being used, in which 8KB block of data is converted into a 20 bytes digest. Thus the design presents a dramatic reduction in storage space requirement for various workloads and hence reduces time to perform backup in bandwidth constraint environment.

Keywords: De-duplication, three-tier architecture, Hash Algorithm, Bandwidth conservation, Storage reduction.

1 Introduction

Our idea is to exploit the de-duplication techniques in online or on-the-fly compression, as de-duplication was earlier being used only in back-up operations. Also existing file sharing systems like Bit-Torrent [1] does not facilitate the user to update the version of his own file. In our implementation while downloading a large file, our system exploited *Duplicate Transfer Detection* (DTD), i.e. client will only download non-redundant blocks from multiple sources [2].

2 Proposed File Sharing Network System

System works in Coordinator-peer architecture, where information about files resides on a central server and peers contain actual data that needs to be downloaded. Server stores data in database tables, which is directly accessible only from Application server and not from client. Server has optimizations for computing hashes and it has scalable architecture to handle multiple requests. Clients have capability of returning blocks of shared files when requested remotely. Client can make the file sharing private by making a peer group. Metadata is shared by server and files are

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 551–553, 2011.

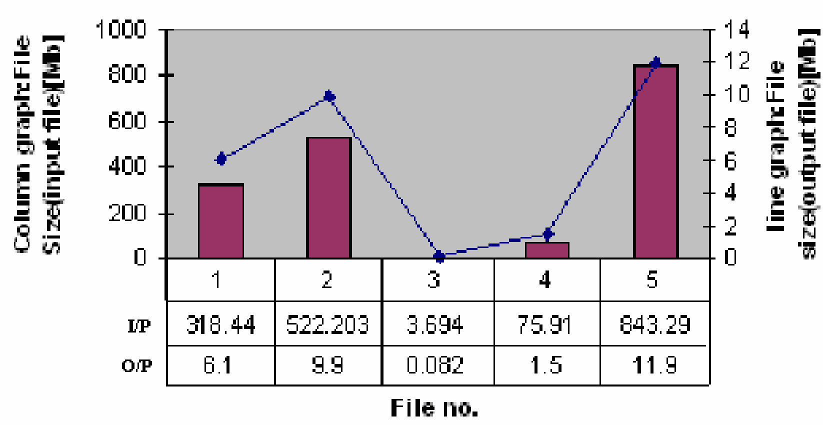
downloaded P2P. This ensures robustness during downloading and updating of file in the network. The proposed algorithm for the duplicate finder (at database server) is as follow:

```
Duplicate_Finder(file_name, file_hash_values)
Begin
updatedVersion ←findLatestVersion(file_name)
L = totalHashes(updatedVersion)
M= totalHashes(file_name)
TotalDe-dupeBlocks= 0
If L>M OR L==M
then
        For( i=1;i<=L;i++) // to check the de-duplicate blocks hash by hash
        Begin
        N= compareHash(file_name$_i$ , updatedVersion$_i$)
                If N==0   // de-duplicate block present
                then
                Increment TotalDe-dupeBlocks by1
                Endif
        End
        If TotalDe-dupeBlocks > 0
        UpdateFile(file_name)
        Return(updated_Version,BlockNumbers,ClientIPAddress)
        Else
        return NULL ; // No latest Version Present at the server
        Endif
Else            // L<<M
DownloadFile(file_name,version)
Return(BlockNumbers,ClientIPAddress)
Endif
End
```

Fig. 1. Algorithm for duplicate_finder

3 Implementation Results and Performance Analysis

Our experimental set comprises of PC's, each connected to LAN (for testing purpose we used JIIT LAN). PC1 acting as database server, PC2 as application server (tracking and handling the clients) and rest PC's as potential clients for our system. When client shares the file, rather storing entire file data to the De-duplication server, only metadata of the file gets stored. Hence, reduction ratio is original file size to size of metadata of the file shared by the server. When any client wants the updated version of the file, application server sends only the de-duplicated hashes to the client; these hashes are further used by the client side of the system to retrieve the de-duplicated blocks data from the client who shared the updated version of the file. So, in the whole process, data blocks of the file which are redundant are not being shared (or transferred), which consumes more than half portion of the bandwidth. Hence, the system is able to save considerable amount of bandwidth in the network. Here, size of each file is taken in Megabytes.



	1	2	3	4	5
I/P	318.44	522.203	3.694	75.91	843.29
O/P	6.1	9.9	0.082	1.5	11.9

Fig. 2. Graph shows reduction in data stored of shared file

In fig.2, vertical bars depicts original size of files being shared by different clients in terms of MB and the line graph depicts the size of metadata being stored at for the respective files in MB. The storage reduction can be viewed in terms of vertical scale at left and right axis of the graph. Database design avoids redundancy by use of special keys assigned by the server, to the data-elements. Version control is implemented using modified timestamp put by OS and version is identified by the timestamp. Redundancy in communication data is avoided by sending temporal maps to resolve the IDs at client side rather than returning redundant data.

4 Future Works

The approach can also be implemented as an application layer protocol where first message in any file transfer, returns a list of duplicate blocks. This can also be implemented in Network layer of TCP/IP stack where each router can de-duplicate the list of blocks.

References

1. Fan, B., Chiu, D.M., Lui, J.C.S.: The Delicate Tradeoffs in Bittorrent-Like File Sharing Protocol Design. In: Proceeding of 14th IEEE International Conference on Network Protocols (ICNP), pp. 239–248 (2006)
2. Mogul, J., Chan, Y., Kelly, T.: Design, Implementation and Evaluation of Duplicate and Transfer Detection in HTTP. In: Proceeding of First Symposium on Networked Systems Design Implementation, vol. 8(3), pp. 221–254 (2004)

Event Management Software Evolution Using FOP and AOP Integration Approach in Eclipse-Based Open Source Environment

Amita Sharma[1] and S.S. Sarangdevot[2]

[1] Tulsi Shree, B-16, Kanta Khaturia Colony, Bikaner-334003, India
amita214@rediffmail.com
[2] Director, Deptt. Of Computer Science & I.T., J.R.N. Rajasthan Vidyapeeth (Deemed) University, Udaipur-313001, India
drsssarangdevat@yahoo.com

Abstract. Integration of Feature-Oriented Programming (FOP) and Aspect-Oriented Programming (AOP) methodologies can overcome their individual limitations and thus superior software can be evolved. This approach has been investigated in evolving application software for a representative 'Event Management System' using Eclipse-based open source environment. The study concludes that this approach supports modular design and implementation of consistent, reusable, maintainable and cost effective 'Event Management Software System', tailored to the specific needs of the stakeholders.

Keywords: Feature-Oriented Programming, Aspect-Oriented Programming, Feature Model, Eclipse-FeatureIDE-AJDT, FeatureHouse, Event Management.

1 Introduction

Integration of FOP [1] and AOP [2] approach is reported to improve both heterogeneous and homogeneous crosscutting modularity [3, 4, 5]. This approach has been investigated in evolving a user-friendly and menu driven *'Event Management Software'* using Eclipse-FeatureIDE-AJDT open source environment and FeatureHouse tool chain. In summary, the following contributions are made: (1) development of 'Feature Model' for a new domain of 'Event Management', (2) identification of its crosscutting concerns, and (3) using this case study, testing the usefulness of FOP and AOP integration approach in business software evolution.

2 Design and Implementation

In the first phase, the domain was analyzed to identify the features and their relationships to develop the 'Feature Model' (Figure 1) using Eclipse-FeatureIDE. Next, features were implemented as feature modules. Based on feature selection FOP system was generated. Crosscutting concerns were then identified and represented as BillingCheck, EventManager, ModificationManager and Logging aspects, using AOP

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 554–555, 2011.

methodology through AspectJ Development Tools (AJDT). The final software product was tested for its correctness and quality.

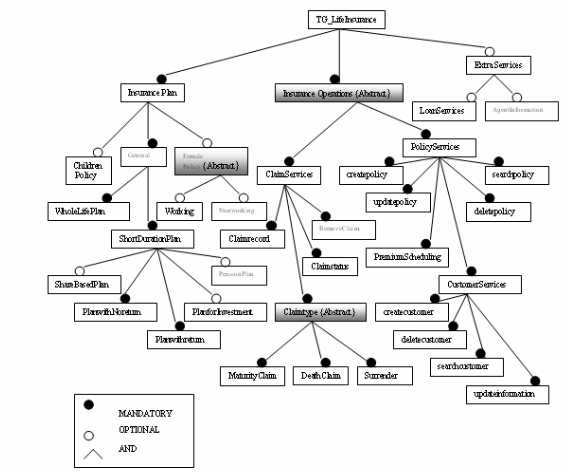


Fig. 1. Feature Model Diagram for Event Management System

3 Conclusion

It is observed that integration of FOP and AOP improves modularity, reduces complexity and software development and maintenance cost. Software is elegant, flexible, reusable and highly adaptive to changing requirements. Using aspect-enhanced FOP, the power of aspect is controlled. Successful implementation concludes that integration of FOP and AOP methodologies supports modular design and implementation of consistent, reusable, maintainable and cost effective 'Event Management Software System', tailored to the specific needs of the stakeholders.

References

1. Prehofer, C.: Feature-Oriented Programming: A Fresh Look at Objects. In: Aksit, M., Auletta, V. (eds.) ECOOP 1997. LNCS, vol. 1241, pp. 419–443. Springer, Heidelberg (1997)
2. Kiczales, G., et al.: Aspect-Oriented Programming. In: Aksit, M., Auletta, V. (eds.) ECOOP 1997. LNCS, vol. 1241, pp. 220–242. Springer, Heidelberg (1997)
3. Apel, S., Leich, T., Saake, G.: Aspectual Feature Modules. IEEE Trans. On Software Engineering 34(2), 162–180 (2008)
4. Kastner, C., et al.: FeatureIDE: A Tool Framework for Feature-Oriented Software Development. In: Proc. ICSE 2009, Vancouver, Canada, May 16-24 (2009)
5. Apel, S., Kastner, C., Lengauer, C.: FeatureHouse: Language-Independent, Automated Software Composition. In: Proc. ICSE 2009, pp. 221–231. IEEE Computer Society Press, Washington DC, USA (2009)

A Robust Bengali Continuous Speech Recognizer Using Triphone and Trigram Language Model

Sandipan Mandal, Biswajit Das, Pabitra Mitra, and Anupam Basu

Department of Computer Science and Engineering,
IIT Kharagpur, India
{mandal.sandipan,bdas,pabitra,anupambas}@gmail.com

Abstract. In this paper we introduce a robust Bengali Automatic Speech Recognition(ASR) system which covers most of the commonly spoken words. This ASR system converts standard Bengali continuous speech to Bengali Unicode with a decent accuracy rate. The existing reported Bengali ASR system is confined within small vocabulary. The system uses triphone clustering mechanism and trigram language model to increase accuracy. For execution of training we have created Bengali Speech Corpus, corresponding Bengali Text corpus, Pronunciation Dictionary.

1 Training Module

SphinxTrain[2] is the acoustic training module for sphinx3 decoder [2] and its a collection of programs and scripts to build acoustic models from training data.

1.1 Corpus Creation

To develop Bengali speech corpus transcript sentences were selected from Anadabazar Patrika, web-based blogs, common conversations and editorials articles and those sentences were recorded using sony v-220 microphone in different sessions. Transcript sentences were labeled corresponding to spoken sentences. Corpus specification is given in Table 1.

Table 1. Corpus Specification

Parameter	Value	Parameter	Value
Sampling Rate	16 kHz, 16 bits	Wave Format	Mono, Wav
Language	Bengali	Sentence	9,000
Unique Word	17,000	Speakers	30 Males + 10 Females

1.2 Phoneme Selection and Pronunciation Dictionary

Although Bengali alphabet contains total 39 consonants and 12 vowels, it has been observed that pronunciation of Bengali words in our speech corpus can be covered using 47 phonemes. Example of pronunciation dictionary and silence dictionary is given bellow.

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 556–558, 2011.

Words	Phoneme Representation
A.DAi	A R A i
A.DAiguNa	A R A i g u n

Dictionary Structure

<s>	SIL
</>	SIL
<sil>	SIL

Silnce Dictionary

1.3 Acoustic Features Computation and Triphone Acoustic Model

16-bit 16 KHz wave files are windowed in frames with duration of 25 ms with consecutive frame overlap by 10 ms. The basic feature vector is Mel Frequency Cepstal Coefficients(MFCC) [1]. MFCC are computed by taking Discrete Cosine Transform (DCT) of the log power spectrum from Mel spaced filter banks [5]. 13 Mel frequency cepstra are computed, x(0), x(1),... x(12), for each window of 25 ms. x(0) is represents for log mel spectrum energy, and it is used to derive other feature parameters. Rest of the 12 coefficients are basic feature vectors. For the temporal properties, 3 other derived vectors are constructed from the basic MFCC coefficients: a 40-ms and 80-ms differenced MFCCs (24 parameters), a 12-coefficient second order differenced MFCCs, and 3 dimensional vector representing the normalized power (log energy), differenced power, and second-order differenced power.

For state probability distribution we use continuous density of Gaussian Mixture distributions. All phonemes (unit of pronunciation) are modeled as a sequence of HMM state and likelihoods (emission probability) of a certain frame observation is produced by using traditional Gaussian Mixture Model(GMM)[3]. Training data includes thousands of sentences or utterances consisting of the spoken text and corresponding audio sample stream. Each utterance, the text is converted into a linear sequence of triphone HMMs using the pronunciation lexicon. This is usually called the sentence HMM. Best state sequence through the sentence HMM is selected for the corresponding feature vector sequence. Each feature frame is labelled with a senone ID[2]. The best state sequence is one with the smallest mismatch between the input feature vectors and the labelled senones underlying statistical models. Circularity problem in training is resolved by using the iterative Baum-Welch or forward-backward [2,3]training algorithm.

Triphone based modeling extracts the left right context information from the training corpus depending upon the observations to represent continuity and co-articulation effects in continuous speech [4]. Individual phones within words are modeled using their left and right context and the phonemes at the word boundaries are modeled as diphones.

1.4 Trigram Language Model

The probability of any word in a sequence of words depends only on the previous N words in the sequence. Thus, a trigram language model[2] would compute as

$$P(wd_1 wd_2 \ldots wd_n) = P(wd1)P(wd2|wd1)P(wd3|wd2,wd1)P(wd4|wd3,wd2)...$$

Word unigram is counted from text and trigram language model is created using

CMU Language Modeling toolkit. Later this model is converted into standard ARPA format.

2 Results

90 test sentences was created using dictionary words and recorded by 4 training speaker and 4 other speaker. Evaluation is performed using slite tool. From Table 2 and Fig.1 , best result is observed using 16 Gaussian mixture densities and 5 state HMM.

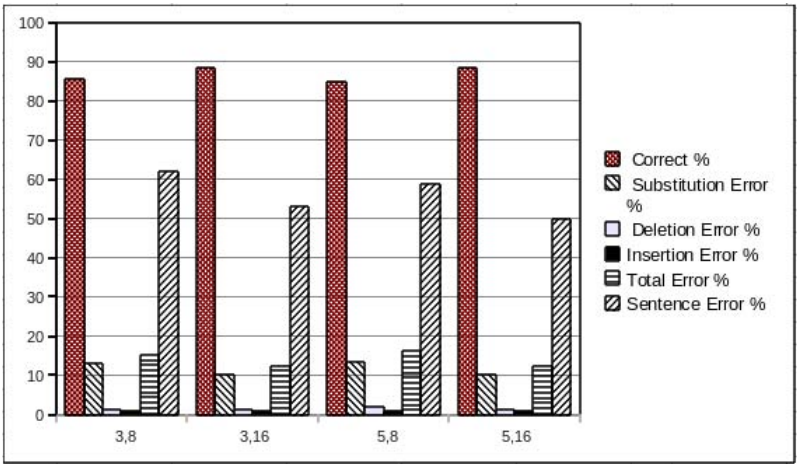


Fig. 1. Result Comparison

Table 2. Accuracy and Error Percentage Comparison

Gaussian,HMM	Correct	Substitution Err	Deletion Err	Insertion Err	Sentence Err
8,3	85.6	13.0	1.4	0.9	61.8
16,3	88.4	10.2	1.4	0.9	52.9
8,5	84.7	13.4	1.9	0.9	58.8
16,5	88.4	10.2	1.4	0.9	50.0

References

1. Rabiner, L., Juang, B.H.: Fundamentals of speech recognition, 3rd edn. Prentice-Hall, Addison-Wesley, Englewood Cliffs,Harlow (1993)
2. CMU Sphinx Group, http://www.speech.cs.cmu.edu/sphinx/
3. Kannan, A., Ostendorf, M., Itohlicek, J.R.: Maximum Likelihood Clustering of Gaussians for Speech Recognition (1994)
4. Young, S.: The HTK Book, Microsoft Corporation and Cambridge University Engineering Department(CUED), Ver 3.4. MIT Press, Cambridge (2006)

VHDL Implementation of PCI Bus Arbiter Using Arbitration Algorithms

Paramjot Saini[1], Mandeep Singh[2], and Balwinder Singh[2]

[1] K.C. College of Engg & Nawanshar (Punjab), India
[2] Centre for Development of Advanced Computing, Mohali, India

Abstract. System on Chip (SOC) is the integration of IP core like CPU'S, DSP's, Application Cores, memories etc. Communication between these IP cores is necessary for proper functionality of SOC. On chip communication arbiters plays an important role for the communication arbitration. In this paper, four arbitration algorithms i.e. Round Robin, Lottery Based Arbiter, FIFO (First in First out), TDMA (Time Division Multiple Access) are implemented in Hardware Description Languages. The results of four arbiters are compared the basis of area, power and delay.

Keywords: On chip communication, Arbitration, Round Robin, Processor, Lottery Based Arbiter.

1 Bus Arbitration

An Arbiter plays an important role in communicating devices and in SOC. It is used to decide the priority level of devices communicating on a single chip or connected through bus. Usually a number of devices in a system send a request to transfer data, at the same time these devices can't access the bus at the same time. To overcome this problem, arbiters are used. In the previous work [2] three arbitration algorithms were compared on the basis of power consumed. But area, delay was not considered and In [3] analysis of power consumption by different arbiters like Fixed Priority Arbitration with hold Control (FAWHA), Fixed Priority Arbitration, TDMA arbiters. In [4] the performance analysis of the Bus Architectures like Wishbone, PCI, AMBA AHB, Core connect done with three level dynamic scheduler, two level dynamic scheduler, Real time static priority, Real time Round Robin, but not with TDMA, FIFO And Lottery based algorithms. In this section four arbitration algorithms are discussed in briefly.

First in First out Arbiter: FIFO Arbiter is based on first in first out scheme

Round Robin arbiter: In Round Robin Arbiter, token number decides priority level in rounded fashion. Devices having a token number can access the bus. If first higher priority device doesn't want to access the bus then second higher priority device access the bus [5].

Time division multiple access (TDMA): TDMA is a Memory Arbiter which provides static Scheduling for memory Access. Memory Access time is independent

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 559–560, 2011.

of task running on another core. TDMA is two level arbitration algorithms, in first level access time is divided into number of time slots and these slots are assigned to master. Master with current time slot has a pending request, arbiter grants permission to transfer data then wheel is rotated or move to text time slot. When there is no pending request from master for current slot then second level of arbitration occurs. In second level of arbitration, arbiter grants permission of round robin fashion.

Lottery Based Arbiter: This arbitration method having a centralized lottery manager. Lottery based arbiter consist of master and manager. Manger decides a ticket number and then compared with a master's ticket number to provide output.

2 Results and Conclusion

The arbiters discussed above were modeled in VHDL in Xilinx 9.1 , simulated on Modelsim 5.8 and functionality of each arbiter is verified and number of LUT's, IOB's, Gate power and delay was calculated from the synthesis results is shown in table 1. In round robin and TDMA arbiter area overhead is more as compared to lottery and FIFO arbiter. Power utilized by lottery arbiter minimum and maximum in TDMA as compared to other arbiters. Delay of round robin arbiter is maximum compared to others.

Table 1. Comparison of Hardware Overhead ,power and delay of Arbiter Algorithms

Parameter	**Round Robin**	**Lottery**	**FIFO**	**TDMA**
LUT'S	25	8	11	23
Bonded IOB's	13	11	10	38
Gate Count	220	83	480	1,824
Power (µW)	41.25	0.04125	0.0412	0.04165
Delay (ns)	6.429	6.388	4.178	4.155

References

1. Zhang, Y.: Architecture and performance comparison of a statistic-based lottery arbiter for shared bus on chip. In: Proceedings of Asia and South Pacific Design Automation Conference, vol. 2, pp. 1313–1316 (2005)
2. Srinivasan, P., Olugbon, A., Ahmadinia, A., Erdogan, A.T., Aslan, T.: Power Analysis of Arbitration Techniques for AMBA AHB Based Reconfigurable System on Chip. In: 24th Norchip Conference, pp. 227–230 (2006)
3. Srinivasan, P., Ahmadinia, A., Erdogan, A.T., Aslan, T.: Integrated Heterogeneous modeling for power estimation of single pr ocessor based reconfigurable SOC Platform. In: IEEE SOC Conference, Taiwan, pp. 159–162 (2007)
4. Hema Chitra, S., Vanathi, P.T.: Design and Analysis of Dynamically Configurable Bus Arbiters for SoCs. ICGST-PDCS Journal 8(1), 227–230 (2008)
5. Shin, E.S., Mooney, V.J., Riley, G.F.: Round Robin Arbiter design and Generation. In: Proceedings of the 15thinternational Symposium on System Synthesis, pp. 243–248 (2002)

Erratum: Performance Analysis of Handover TCP Message in Mobile Wireless Networks

Ashutosh Kr Rai[1] and Rajnesh Singh[2]

[1] Shobhit University, Meerut, India
[2] Rajnesh Singh, Manav Bharti University, Solan, India
akrai.iimt@gmail.com, rajneshcdac.mtech@gmail.com

S. Aluru et al. (Eds.): IC3 2011, CCIS 168, pp. 254–261, 2011.

DOI 10.1007/978-3-642-22606-9_63

Due to a serious case of plagiarism this paper has been retracted.

The paper "Performance Analysis of Handover TCP Message in Mobile Wireless Networks" <http://www.springerlink.com/content/h12j112777784247/> appearing on pages 254-261 of this publication has been retracted due to a severe case of plagiarism.

The original online version for this chapter can be found at
http://dx.doi.org/10.1007/978-3-642-22606-9_27

Author Index