



Global Journal of Computer Science and Technology

discovering thoughts and inventing future





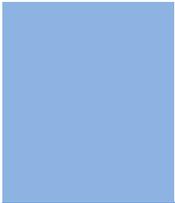
Global Journal of Computer Science and Technology



Global Journal of Computer Science and Technology

Volume 1.2

Global Academy of Research and Development



Copyright by Global Journal of Computer Science and Technology 2009. All rights reserved.

This is a special issue published in version 1.0 of “Global Journal of Computer Science and Technology.” All articles are open access articles distributed under the Global Journal of Computer Science and Technology Reading License, which permits restricted use. Entire contents are copyright by of “Global Journal of Computer Science and Technology.” unless otherwise noted on specific articles. No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopy, recording, or any information storage and retrieval system, without written permission.

The opinions and statements made in this book are those of the authors concerned. Ultraculture has not verified and neither confirms nor denies any of the foregoing and no warranty or fitness is implied.
Engage with the contents herein at your own risk.

Editorial Board Members

John A. Hamilton, "Drew" Jr.,

Ph.D., Professor, Management
Computer Science and Software Engineering
Director, Information Assurance Laboratory
Auburn University

Dr. Wenying Feng

Professor, Department of Computing &
Information Systems
Department of Mathematics
Trent University, Peterborough,
ON Canada K9J 7B8

Dr. Henry Hexmoor

IEEE senior member since 2004
Ph.D. Computer Science, University at
Buffalo
Department of Computer Science
Southern Illinois University at Carbondale

Dr. Thomas Wischgoll

Computer Science and Engineering,
Wright State University, Dayton, Ohio
B.S., M.S., Ph.D.
(University of Kaiserslautern)

Dr. Osman Balci, Professor

Department of Computer Science
Virginia Tech, Virginia University
Ph.D. and M.S. Syracuse University, Syracuse,
New York
M.S. and B.S. Bogazici University, Istanbul,
Turkey

Dr. Abdurrahman Arslanyilmaz

Computer Science & Information Systems
Department
Youngstown State University
Ph.D., Texas A&M University
University of Missouri, Columbia
Gazi University, Turkey

Yogita Bajpai

M.Sc. (Computer Science), FICCT
U.S.A.
Email: yogita@computerresearch.org

Chief Author

Dr. R.K. Dixit (HON.)

M.Sc., Ph.D., FICCT

Chief Author, India

Email: authorind@computerresearch.org

Dean & Editor-in-Chief (HON.)

Vivek Dubey(HON.)

MS (Industrial Engineering),

MS (Mechanical Engineering)

University of Wisconsin

FICCT

Editor-in-Chief, USA

editorusa@globaljournals.org

Sangita Dixit

M.Sc., FICCT

Dean and Publisher, India

deanind@globaljournals.org

Er. Suyog Dixit

BE (HONS. in Computer Science), FICCT

SAP Certified Consultant

Technical Dean, India

Website: www.suyogdixit.com

Email: suyog@suyogdixit.com,

dean@globaljournals.org

Er. Prachi Telang

BE (HONS. in Computer Science), FICCT

Project Manager, India

Email: prachicse@globaljournals.org

Contents of the Volume

- i. Copyright Notice
- ii. Editorial Board Members
- iii. Chief Author and Dean
- iv. Table of Contents
- v. From the Chief Editor's Desk
- vi. Research Papers
 1. Input Data Processing Techniques in Intrusion Detection Systems – Short Review, **2**
 2. Semantic Annotation of Stock Photography for CBIR using MPEG-7 standards, **7**
 3. An Experimental Study to Identify Qualitative Measures for Website Design, **12**
 4. Process modeling using ILOG JViews BPMN Modeler tool to Identify Exceptions, **18**
 5. A new approach to: Obstacle-Avoiding Rectilinear Steiner Tree Construction, **24**
 6. Algorithmic Approach for Creating and Exploiting Flexibility in Steiner Trees, **27**
 7. Initial Hybrid Method for Software Effort Estimation, Benchmarking and Risk Assessment Using Design of Software, **34**
 8. Diffie-Hellman Key Exchange: Extended to Multi-Party key Generation for Dynamic Groups, **41**
 9. A Framework for Systematic Database Denormalization, **44**
 10. Experiments with Self-Organizing Systems for Texture and Hardness Perception, **53**
 11. Diagnosing Parkinson by using Artificial Neural Networks and Support Vector Machines, **63**
 12. Secured Data Comparison in Bioinformatics using Homomorphic Encryption Scheme, **72**
 13. Performance Evaluation of Message Encryption Scheme Using Cheating Text, **77**
 14. Finding Error Correction of Bandwidth on Demand Strategy for GPRS Using Constant Modulus Algorithm, **81**

15. An Introduction to DNA Computing, **88**
 16. Distributed Diagnosis in Dynamic Fault Environments using HeartBeat Algorithm, **96**
 17. Temperature Variation on Rough Actor-Critic Algorithm, **103**
 18. Evaluation of Efficient Web caching and prefetching technique for improving the proxy server performance, **108**
 19. Wireless LAN Security System, **113**
 20. A Trust-Based Secured Routing Protocol for Mobile Ad hoc Networks, **121**
 21. Generation of Fractal Music with Mandelbrot set, **127**
 22. Performance Analysis & QoS Guarantee in ATM Networks, **131**
 23. Survey of Forest Fire Simulation, **137**
 24. Detecting Redundancy in Biological Databases – An Efficient Approach, **141**
 25. Semantic Search and Retrieval of Stock Photography based on MPEG-7 Descriptors, **146**
 26. Efficient use of MPEG-7 Color Layout and Edge Histogram Descriptors in CBIR Systems, **157**
 27. Computation of Merging Points in Skeleton Based Images, **164**
 28. Separating Words from Continuous Bangla Speech, **172**
 29. A Survey on User Interface Defect Detection in Object Oriented Design, **172**
 30. A Survey- Object Oriented Quality Metrics, **183**
 31. Modeling and Analysis of the Associative Based Routing (ABR) Protocol by Using Coloured Petri Nets, **187**
 32. A Framework of Distributed Dynamic Multi-radio Multi-channel Multi-path Routing Protocol in Wireless Mesh Networks, **193**
 33. A Security Analysis Framework for Dynamic Web Applications, **199**
 34. Analysis Of Knowledge Management Tools, **204**
- vii. Fellows
 - viii. Auxiliary Memberships
 - ix. Process of Submission of Research Paper
 - x. Preferred Author Guidelines
 - xi. Index

From the Chief Author's Desk

Research is the backbone of any stream of knowledge and philosophy. Computer Science is the frontier of all the streams, so research activity in computer science is the most effective process to make people and society stronger in all the aspects. Computer Science research is extremely fast growing field. So, as to cover up all the aspects, it is highly demanded to stable a research journal of real time.

Now-a-days, the field of Computer Science is spreading its essence in all the areas. Research is one of the most important factor of that. The main reason of success of computer research field is that, people relating to it, get highly involved in research and spend couple of days to put their ideas in the best way, so that readers can be enriched with more and more knowledge and philosophy.

The “GJCST” is matching all these requirements. The scope of GJCST covers near about all the fields of computer science & technology, internationally. GJCST is associating the groups of researchers all over the world. Computer science is the essential and important part of near about all the branches of research and its applications. So, GJCST is very important journal helping to develop the main structure of knowledge and reforming the field of research.

GJCST is the platform to facilitate people and providing way to express their researches to the communities in every part of world. People are giving their views over a topic, so that they can be helpful for future in developing new vision for the people directly or indirectly related with computer science stream. GJCST is the mediatory to represent the ideas of researchers.

“GJCST”, as the name reflects, globally exploring the ideas of people, to the people and for the people for vast development in the field of computer science.

Dr. R. K. Dixit

Input Data Processing Techniques in Intrusion Detection Systems – Short Review

Suhair H. Amer, and John A. Hamilton, Jr.

Abstract—In this paper intrusion detection systems (IDSs) are classified according to the techniques applied to processing input data. This process is complex because IDSs are highly coupled in actual implemented systems. Eleven input data processing techniques associated with intrusion detection systems are identified. They are then grouped into more abstract categories. Some approaches are artificially intelligent such as neural networks, expert systems, and agents. Others are computationally based such as Bayesian networks, and fuzzy logic. Finally, some are based on biological concepts such as immune systems and genetics. Characteristics of and systems employing each technique are also mentioned.

I. INTRODUCTION

When traditionally classifying intrusion detection systems (IDSs) as misuse, anomaly or hybrid, the systems are grouped according to the technique they utilize to detect intrusions. For example, misuse-based IDSs match already stored attack signatures against the audit data gathered while the monitored system is or was running. In anomaly based IDSs, detection utilize models of normal behavior where any deviation from such behavior is identified as an intrusion. Another type of traditional classification is categorizing an IDS according to its setup as network-based, host-based or hybrid. Network based systems monitor network activities whereas a host based system monitor the activities of a single system for intrusion traces [1]. In general, IDSs may apply many techniques to detect intrusions and improve detection such as neural networks, expert systems, agents, Bayesian networks, fuzzy logic, immune systems and genetics. Little attention has been given to classifying the processing techniques applied on the input data provided to the IDS. In this paper we classify input data processing techniques utilized with IDSs that may use and may not use the same processing technique to detect intrusions. In section 2, abstract classification of the different input data processing techniques utilized with IDSs will be presented. Eleven input data processing techniques associated with IDSs are identified. Then they are grouped into more abstract categories. In section 3, a general description as well as some advantages and disadvantages of each technique and examples of system employing these techniques will be presented.

Manuscript received July 31, 2009.

S. H. Amer was with Auburn University, Auburn, AL 36849 USA. She is now with the Department of Computer Science, Southeast Missouri State University, Cape Girardeau, MO 63701 USA (telephone: 573-651-2525, e-mail: samer@semo.edu).

J. A. Hamilton, Jr., is with the Department of Computer Science and Software Engineering, Auburn University, Auburn, AL 36849 USA (telephone: 334-844-6360, e-mail: hamilton@auburn.edu).

II. CLASSIFICATION OF INPUT DATA PROCESSING TECHNIQUES IN IDSs

In this paper we are concerned with the techniques used to process input data that is considered when designing and implementing IDSs. Classifying such techniques are not easy because in the actual implemented system, combination of techniques may be used. However, identifying them individually helps better understand the merits and limitations of each, and how to improve a techniques performance by using another. Eleven techniques are identified [shown at the lower level of diagram 1] that are widely and currently used for processing input data of IDSs. They are then grouped into more abstract categories that are identified at the upper levels of diagram 1. This is important because the characteristics of each technique are highly affected by the category(ies) that it belongs to. In the lower level of Fig. 1, techniques such as Agents and Data Mining belong to the Intelligent Data Analysis category. This is indicated by the dotted relation between Data Analysis and AI categories. The techniques: Expert systems and Fuzzy logic are intelligent model-based-rule-based systems shown by the dotted relation between Rule based and AI categories in Fig. 1. Next is an explanation of each item in Fig. 1, along with some identified characteristics.

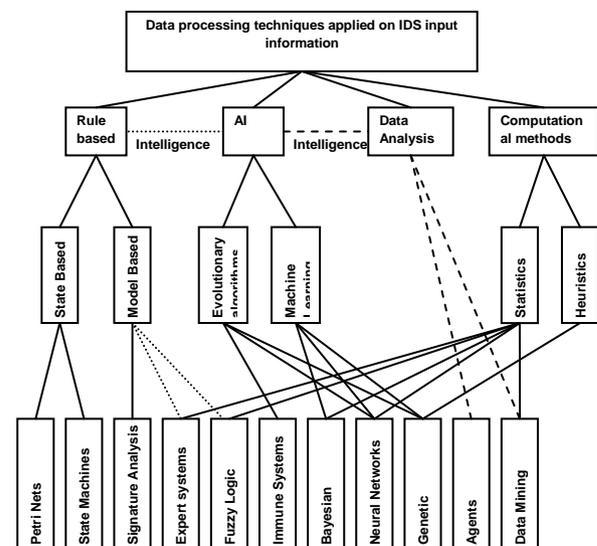


Fig.1. Data processing techniques applied on input data processed by Intrusion Detection Systems.

A. Rule Based

If a rule-based IDS is to use input data or audit data, such information will be in a codified rules format of known intrusions. The input data will represent identified intrusive behavior and categorizing intrusion attempts by sequences of user activities that lead to compromised system states. The IDS will take as input the predefined rules as well as the current audit data and check if a rule is fired. In general, using rule bases are affected by system hardware or software changes and require updates by system experts as the system is enhanced or maintained. Such input data technique is very useful in an environment where physical protection of the computer system is not always possible (e.g., a battlefield situation) but require strong protection [<http://www.sei.cmu.edu/str/descriptions/rbid.html>].

In general, rule based systems can be:

1. State-based: in the audit trails, intrusion attempts are defined as sequences of system states leading from an initial state to a final compromised state represented in a state transition diagram. The two inputs to the IDS will include the audit trail and the state transition diagrams of known penetrations that will be compared against each other using an analysis tool. One advantage of using state based representation of data is that it is independent of the audit trail record and is capable of detecting cooperative attacks and attacks that span across multiple user sessions. However, some attacks cannot be detected because they cannot be modeled with state transitions [<http://www.sei.cmu.edu/str/descriptions/rbid.html>].
2. Model-based: intrusion attempts in input data can be modeled as sequences of user behavior. This approach allows the processing of more data, provide more intuitive explanations of intrusion attempts and predict intruder's next action. More general representation of penetrations can be generated since intrusions are modeled at a higher level of abstraction. However, if an attack pattern does not occur in the appropriate behavior model it cannot be detected [<http://www.sei.cmu.edu/str/descriptions/rbid.html>].

B. Artificial Intelligence (AI)

AI improves algorithms by employing problem solving techniques used by human beings such as learning, training and reasoning. One of the challenges of using AI techniques is that it requires a large amount of audit data in order to compute the profile rule or pattern sets. From the audit trails, information about the system is extracted and patterns describing the system are generated. In general, AI can be employed in two ways: (1) Evolutionary methods (Biologically driven) are mechanisms inspired by biological evolution, such as reproduction, mutation and recombination. (2) Machine learning is concerned with the design and development of algorithms and techniques that allow the learning of computers. The major focus of

machine learning research is to extract information from data automatically [2].

C. Data Analysis

With data analysis, data is transformed in order to extract useful information and reach conclusions. It is usually used to approve or disapprove an existing model, or to extract parameters necessary to adapt a theoretical model to an experimental one. Intelligent data analysis indicates that the application is performing some analysis associated with user interaction and then provides some insights that are not obvious. One of the problems faced when applying such an approach is that most application logs (input information) do not conform to a specific standard. Analysis of logs should be performed to find commonalities and different types of logs should be grouped. Another problem is the existence of noise, missing values and inconsistent data in the actual log information. Attackers may take advantage of the fact that logs may not record all information and therefore exploit this point. Finally, real world data sets tend to be too large and multidimensional which requires data cleaning and data reduction [3].

D. Computational Methods

Computational intelligence research aims to use learning, adaptive, or evolutionary algorithms to create programs. These algorithms allow the systems to operate in real time and detect system faults quickly. However, there are costs associated with creating audit trails and maintaining input user profiles as well as some risks. For example, because user profiles are updated periodically, it is possible to accept a new user behavior pattern where an attack can be safely mounted. This is why it is difficult sometimes to define user profiles especially if they have inconsistent work habits. In general, there are two types of IDSs that utilize a computational method: (1) Statistics-based IDS are employed to identify audit data that may potentially indicate intrusive behavior. These systems analyze input audit trail data by comparing them to normal behavior to find security violations. (2) Heuristics-based IDS which can be a function that estimates the cost of the cheapest path from one node to another [<http://www.sei.cmu.edu/str/descriptions/sbid.html>].

III. CAPABILITIES AND EXAMPLES OF PROCESSING TECHNIQUES OF INPUT DATA USED BY IDSs

Because some IDS data processing techniques are closely interacting and similar, classifying them is complex. However, we believe that the identified eleven categories capture most of the well known types. For example, from Fig. 1, although expert systems and fuzzy logic belong to the categories AI and rule based they have distinguishing characteristics and usages. The output of the expert system is specific; the data that is used to build the system is complete, and the set of rules are well defined. As for fuzzy logic, it is usually used in systems where the output is not well defined and is continuous between 0 and 1.

A. Bayesian networks

Bayesian networks are used when we want to describe the conditional probability of a set of possible causes for a given observed event that are computed from the probability of each cause and the conditional probability of the outcome of each cause. They are suitable for extracting complex patterns from sizable amounts of input information that can also contain significant levels of noise. Several systems have been developed using Bayesian network concepts. In the following system, Scott's [4] IDS is based on stochastic models of user and intruder behavior combined using Bayes' theorem which mitigates the complexity of network transactions that have complicated distributions. Intrusion probabilities can be calculated and dynamic graphics are used to allow investigators to use the evidence to navigate around the system.

B. Neural networks

Training Neural networks enable them to modify a state of a system by discriminating between classes of inputs. They also learn about the relationship between input and output vectors and generalize them to extract new input and output relationships. They are suitable when identification and classification of network activities are based on incomplete and limited input data sources. They are able to process data from a number of sources, accept nonlinear signals as input and need a large sample size of input information. Finally, neural networks are not suitable when the information is imprecise or vague and it is unable to combine numeric data with linguistic or logical data. In the following system, Bivens et al. [5] employed the time-window method for detection and were able to recognize long multi-packet attacks. They were able to identify aggregate trends in the network traffic in the preprocessing step by looking only at three packet characteristics. Once the system is trained and by using the input data, the neural network was able to perform real-time detection.

C. Data mining

Data mining refers to a set of techniques that extracts previously unknown but potentially useful data from large stores system logs. One of the fundamental data mining techniques used in intrusion detection is associated with decision trees [6] that detect anomalies in large databases. Another technique uses segmentation where patterns of unknown attacks are extracted from a simple audit and then matched with previously warehoused unknown attacks [7]. Another data mining technique is associated with finding association rules by extracting previously unknown knowledge on new attacks and building normal behavior patterns [8]. Data mining techniques allows finding regularities and irregularities in large input data sets. However, they are memory intensive and require double storage: one for the normal IDS data and another for the data mining. The system of Lee, Solto and Mok's [7] was able to detect anomalies using predefined rules; however, it needed a supervisor to update the system with the appropriate rules of certain attacks. The rule generation methodology developed, first defines an association rule that

identifies the relation between rules and specifies the confidence for the rule.

D. Agents

Agents are self contained processes that can perceive their environment through sensors and act on the environment through effectors. Agents trace intruders and collect input information that is related only to the intrusion along the intrusion route and then decide if an intrusion has occurred from target systems across the network. One of the major disadvantages associated with agents is that it needs a highly secure agent execution environment while collecting and processing input information. It is difficult also to propagate agent execution environments onto large numbers of third-party servers. Several systems have been developed utilizing agents. Spafford and Zamboni [9] introduced Autonomous Agents for Intrusion Detection (AAFID) using autonomous agents for performing intrusion detection. Their prototype provides a useful framework for the research and testing of intrusion detection algorithms and mechanisms. Gowadia, Farkas and Valtorta [10] implemented a Probabilistic Agent-Based Intrusion Detection (PAID) system that has cooperative agent architecture. In their model agents are allowed to share their beliefs and perform updates. Agent graphs are used to represent intrusion scenarios. Each agent is associated with a set of input, output, and local variables.

E. Immune based

Immune based IDS are developed based on human immune system concepts and can perform tasks similar to innate and adaptive immunity. In general, audit data representing the appropriate behavior of services are collected and then a profile of normal behavior is generated. One challenge faced is to differentiate between self and non-self data which when trying to control causes scaling problems and the existence of holes in detector sets.

There have been several attempts to implement immunity-based systems. Some have experimented with innate immunity which is the first line of defense in the immune system and is able to detect known attacks. For example, Twycross and Aickelin [11] implemented *libtissue* that uses a client/server architecture acting as an interface for a problem using immune based techniques. Pagnoni and Visconti [12] implemented a native artificial immune system (NAIS) that protects computer networks. Their system was able to discriminate between normal and abnormal processes, detect and protect against new and unknown attacks and accordingly deny access of foreign processes to the server. For adaptive immunity two approaches have been studied: negative selection and danger theory concepts. Kim and Bentley [13] implemented a dynamic clonal selection algorithm that employs negative selection by comparing immature detectors to a given antigen set. Immature detectors that bind to an antigen are deleted and the remaining detectors are added to the accepted population. If a memory detector matches an antigen an alarm is raised. A recent approach to implement adaptive

immunity uses the danger theory concept [14]. Danger theory suggests that an immune response reacts to danger signals resulting from damage happening to the cell and not only for being foreign or non-self to the body.

F. Genetic algorithms

Genetic algorithms are a family of problem-solving techniques based on evolution and natural selection. Potential solutions to the problem to be solved are encoded as sequences of bits, characters or numbers. The unit of encoding is called a gene, and the encoded sequence is called a chromosome. The genetic algorithm begins with chromosomes population and an evaluation function that measures the fitness of each chromosome. Finally, the algorithm uses reproduction and mutation to create new solutions. In the system of Shon and Moon [15] the Enhanced Support Vector Machine (Enhanced SVM) provides unsupervised learning and low false alarm capabilities. Profile of normal packets is created without preexisting knowledge. After filtering the packets they use a genetic algorithm for extracting optimized information from raw internet packets. The flow of packets that is based on temporal relationships during data preprocessing is used in the SVM learning.

G. Fuzzy logic

Fuzzy logic is a system of logic that mimics human decision making and deals with the concept of partial truth and in which the rules can be expressed imprecisely. Several systems have been developed using fuzzy logic. Abraham et al. [16] modeled Distributed Soft Computing-based IDS (D-SCIDS) as a combination of different classifiers to model lightweight and heavy weight IDSs. Their empirical results show that a soft computing approach could play a major role for intrusion detection where the fuzzy classifier gave 100% accuracy for all attack types using all used attributes. Abadeh, Habibi and Lucas [17] describe a fuzzy genetics-based learning algorithm and discuss its usage to detect intrusion in a computer network. They suggested a new fitness function that is capable of producing more effective fuzzy rules that also increased the detection rate as well as false alarms. Finally, they suggested combining two different fitness function methods in a single classifier, to use the advantages of both fitness functions concurrently.

H. Expert systems

Expert systems-based IDSs build statistical profiles of entities such as users, workstations and application programs and use statically unusual behavior to detect intruders. They work on a previously defined set of rules that represent a sequence of actions describing an attack. With expert systems, all security related events that are incorporated in an audit trail are translated in terms of if-then-else rules. The expert system can also hold and maintain significant levels of information. However, the acquisition of rules from the input data is a tedious and is an error-prone process. The system of Ilgun, Kemmerer and Porras [18], is an approach to detect intrusions in real time based on state transition analysis. The model is represented

as a series of state changes that lead from an initial secure state to a target compromised state. The authors developed USTAT which is a UNIX specific prototype of a state transition analysis tool (STAT) which is a rule based expert system that is fed with the diagrams. In general, STAT extracts and compares the state transition information recorded within the target system audit trails to a rule based representation of known attacks that is specific to the system.

I. Signature analysis or Pattern Matching

In this approach the semantic description of an attack is transformed into the appropriate audit trail format representing an attack signature. An attack scenario can be described, for example, as a sequence of audit events that a given attack generates. Detection is accomplished by using text string matching mechanisms. Human expertise is required to identify and extract non conflicting elements or patterns from input data. The system of Kumar's [19] is based on the complexity of matching. Based on the desired accuracy of detection, he developed a classification to represent intrusion signatures and used different encodings of the same security vulnerability. His pattern specification incorporated several abstract requirements to represent the full range and generality of intrusion scenarios that are: context representation, follows semantics, specification of actions and representation of invariants.

J. State machines

State machines model behavior as a collection of states, transitions and actions. An attack is described with a set of goals and transitions that must be achieved by an intruder to compromise a system. Several systems have been developed using this technique. Sekar et al. [20] employ state-machine specifications of network protocols that are augmented with information about statistics that need to be maintained to detect anomalies. The protocol specifications simplified the manual feature selection process used in other anomaly detection approaches. The specification language made it easy to apply their approach to other layers such as HTTP and ARP protocols. Peng, Leckie and Ramamohanarao [20] proposed a framework for distributed detection systems. They improved the efficiency of their system by using a heuristic to initialize the broadcast threshold and hierarchical system architecture. They have presented a scheme to detect the abnormal packets caused by the reflector attack by analyzing the inherent features of the reflector attack.

K. Petri nets

The Colored Petri Nets are used to specify control flow in asynchronous concurrent systems. It graphically depicts the structure of a distributed system as a directed bipartite graph with annotations. It has place nodes, transition nodes and directed arcs connecting places with transitions. In the system of Srinivasan and Vaidehi [22] a general model based on timed colored Petri net is presented that is capable of handling patterns generated to model the attack behavior as sequence of events. This model also allows flagging an

attack, when the behavior of one or more processes matches the attack behavior. Their use of a graphical representation of a timed colored Petri net gives a straightforward view of relations between attacks.

IV. CONCLUSION

Choosing an IDS to be deployed in an environment would seem to be simple, however, with the different components, types and classifications such a decision is quite complex. There have been many attempts to classify IDSs as a mean to facilitate choosing better solutions. In this paper we classified IDSs according to the data processing techniques applied to input information. Careful design of an IDS may allow correct implementation of an IDS. However, the actual merits and limitations of each approach, which is also discussed in this paper, indicate that obtaining complete security and different desirable system characteristics can not be achieved by employing only one type of an implementation approach. The data processing techniques were grouped into general (abstract) categories and were then further expanded into eleven more specialized techniques.

We discussed and summarized the characteristics of each technique followed by examples of developed systems using each technique. Fig. 1, for example, helps us understand that we can use the state machine technique to build an IDS, and that we can add intelligence to it and use the expert system technique with added merits and costs. The merits are the ability to perform and provide intelligent actions and answers. Unrealistic actions or answers can be refuted or ignored. It also borrows from statistics the ability to detect intrusions without prior information about the security flaws of a system. Some of the incurred costs are the conflicting requirement of maintaining high volume of data which affects throughput and selecting the appropriate thresholds that lower false positive and negatives. To conclude, selecting the appropriate technique should be carried out carefully. Each organization should state prior to development the requirements of its agency and the acceptable costs. Accordingly, the selected system should be able to incorporate most of the requirements, as complete security can not be achieved.

REFERENCES

- H. Debar, M. Dacier, A. Wespi, "Towards A Taxonomy Of Intrusion-Detection Systems," *Computer Networks*, Vol. 31, pp. 805-822, 1999.
- S. Peddabachigari, A. Abraham, C. Grosan and J. Thomas, "Modeling Intrusion Detection System Using Hybrid Intelligent Systems," *Journal of Network and Computer Applications*, Vol. [MAT02], No. 1, pp. 114-132, 2007.
- Andre' Muscat, "A Log Analysis Based Intrusion Detection System for the Creation of a Specification Based Intrusion Prevention System", *CSAW 2003 Proceedings*, 2003.
- S. L. Scott, "A Bayesian Paradigm for Designing Intrusion Detection Systems," *Computational Statistics Data Analysis*, Vol. 45, No. 1, pp. 69-83, 2004.
- A. Bivens, M. Embrechts, C. Palagiri, R. Smith, and B. K. Szymanski, "Network based Intrusion Detection using Neural Networks," *Intelligent Engineering Systems through Artificial Neural Networks*, Vol. 12, 2002.
- W. Fan, M. Miller, S. Stolfo, W. Lee, and P. Chan, "Using Artificial Anomalies to Detect Unknown and Known Network Intrusions", *In Proceedings of the First IEEE International Conference on Data Mining*, San Jose, CA, 2001.
- W. Lee, S. J. Stolfo, and K. W. Mok, "Adaptive Intrusion Detection: A Data Mining Approach," *Artificial Intelligence Review*, Vol. 14, No. 6, pp. 533-567, 2000.
- T. Bass, "Intrusion Detection Systems Multi-sensor Data Fusion: Creating Cyberspace Situational Awareness," *Communication of the ACM*, Vol. 43, No. 1, pp. 99-105, 2000.
- E. H. Spafford and D. Zamboni, "Intrusion Detection Using Autonomous Agents," *in Computer Network*, Vol. 34, No. 4, pp. 547-570, 2000.
- V. Gowadia, C. Farkas, and M. Valtorta, "PAID: A Probabilistic Agent-Based Intrusion Detection System," *Computers & Security*, 2005.
- J. Twycross, and U. Aickelin, "Libtissue - Implementing Innate Immunity," *Proceedings of the IEEE Congress on Evolutionary Computation (CEC 2006)*, Vancouver, Canada, 2006.
- A. Pagnoni, and A. Visconti, "An Innate Immune System for the Protection of Computer Networks," *ACM International Conference Proceeding Series*, Vol. 92 archive Proceedings of the 4th international symposium on Information and communication technologies, 2005.
- J. Kim, and P. J. Bentley, "A Model of Gene Library Evolution in the Dynamic Clonal Selection Algorithm," *Proceedings of the First International Conference on Artificial Immune Systems (ICARIS) Canterbury*, pp.175-182, 2002.
- P. Matzinger, "The Danger Model: A Renewed Sense of Self," *Science*, Vol. 296, pp. [MAT02]1- [MAT02]5, 2002.
- T. Shon and J. Moon, "A Hybrid Machine Learning Approach to Network Anomaly Detection," *Information Sciences: an International Journal*, Vol. 177, No. 18, pp. 3799-3821, 2007.
- A. Abrahama, R. Jainb, J. Thomasc, and S. Y. Hana, "D-SCIDS: Distributed Soft Computing Intrusion Detection System," *Journal of Network and Computer Applications*, Vol. 30, pp. 81-98, 2007.
- M. S. Abadeh, J. Habibi and C. Lucas, "Intrusion Detection Using a Fuzzy Genetic Based Learning Algorithm," *Journal of Network and Computer Applications*, Vol. 30, No. 1, pp. 414-428, 2007.
- K. Ilgun, R. A. Kemmerer, and P. A. Porras, "State Transition Analysis: A Rule- Based Intrusion Detection Approach," *IEEE Transactions on Software Engineering*, pp. 181-199, 1995.
- S. Kumar. "Classification and Detection of Computer Intrusions," *Ph.D. dissertation*, Purdue University, 1995.
- R. Sekar, A. Gupta, J. Frullo, T. Hanbhag, A. Tiwari, H. Yang, and S. Zhou, "Specification-Based Anomaly Detection: a New Approach for Detecting," *International Journal of Network Security*, Vol. 1, No.2, pp. 84-102, 2005.
- T. Peng, C. Leckie and K. Ramamohanarao, "Information Sharing for Distributed Intrusion Detection Systems," *Journal of Network and Computer Applications*, Vol. [MAT02], No. 3, pp. 877-899, 2007.
- N. Srinivasan and V. Vaidehi. "Timed Coloured Petri Net Model for Misuse Intrusion Detection." *First International Conference on Industrial and Information Systems*, 8-11 Aug. 2006.

Semantic Annotation of Stock Photography for CBIR using MPEG-7 standards

Balasubramani R

Assistant Professor – IT
Sikkim Manipal University – DDE
I Floor, Syndicate House
Manipal - 576104.
Karnataka, India
Email: microtech_balu@yahoo.com

Dr.V.Kannan

Dean
Centre for Information
Bharath University
Chennai - 600073
Tamil Nadu, India
Email: drvkannan62@yahoo.com

Abstract-Nowadays research and development activities are accompanied by an increasing focus on future user needs in the field of multimedia retrieval. The fast growing of multimedia data repositories is an undeniable fact, so specialized tools allowing storage, indexing and retrieval of multimedia content have to be developed, and in addition easy-to-use content exchange is needed. The transition from text to photo retrieval raises the necessity of generating, storing and visualizing additional meta-information about the content to allow semantic retrieval. “NWCIBR”, a prototype allowing semantic annotation of digital photos based on MPEG-7 standards [4], is presented as a possible new way of handling semantics in descriptions of multimedia data.

Keywords: Semantic annotation, MPEG-7, NWCIBR.

I. INTRODUCTION

The evolution of digital information repositories produce more and more specialized requirements towards intelligent information retrieval. Numerous research and development teams are doing fundamental research concerning various unforeseen topics like managing more than 300 TV channels with a remote control without losing orientation. Base for interdisciplinary future developments are overall agreed standards and standardized methods. Using the following scenario we examined the possibilities current technologies like MPEG-7 [3], bring us in context of one real world problem.

Digital camera users produce a lot of images throughout the year and save them to personal computers. After some time the amount of photos exceeds the critical mass for being manageable without specialized tools. Most people create an intuitive structure for storing their personal image library. They create folders for images that are taken in the same context, for example “Photos from LC Convention Meet June 2008” or “Spiritual Tour Photos”. Nevertheless this does not enable the user to find a photo which shows a specific person, object or even expresses a specific idea or feeling when needed. Some file formats like *TIFF* and *JPEG* permit the user to enrich the visual information with structured textual descriptions, but they only offer limited retrieval capabilities. MPEG-7 offers a whole range of descriptors to annotate images with manually or automatically generated metadata

[2]. The picture can be described in many different ways regarding for example its quality, its technical attributes, its instances (thumbnails, high resolution, and so on) and its content from either a technical or a semantic point of view.

The prototype, NWCIBR system allows annotating digital photos manually and extracts content based on low level features from the image automatically.

II. EXISTING METADATA STANDARDS FOR DESCRIBING MULTIMEDIA

The standard being used to define the way of handling the metadata has to be a lot more powerful than *EXIF* or for instance Dublin Core [5]. *DC* only defines 15 core qualifiers, which can be understood as metadata tags, which can be filled by the user. A combination of Dublin Core and adapted Resource Description Framework structures, *RDF*, would at least permit a structured storage of graphs and a quality rating, although content based image retrieval would not be supported. An import of the *EXIF* information to a *RDF*-based structure is possible. The main proposition against *RDF* is that there exists, at this time, no standardized structure for saving all or most of the metadata defined in the requirements above. Although it would not prove impossible to create such a structure, to gain interoperability with other systems and implementations, agreeing on the same *RDF* based enhancements with all other developers or vendors is necessary. Based on these facts a much better choice is MPEG-7 [1].

III. REALIZATION OF NWCIBR ANNOTATION TOOL

As MPEG-7 is a complex *XML* based standard, it would be no good idea to confront the user with a *XML* editor and an instruction manual as tools for expressing the semantics of a photo. To deal with large description graphs, a visualization of this graph, besides a possibility to edit this graph interactively, is necessary. As a result NWCIBR System’s Annotation Tool was designed for supporting the user in the time consuming task of annotating photos.

The Annotation Tool was implemented using *Sun Java SDK 1.4*, while as runtime environment the versions *JRE 1.4* and higher are supported. For *XML* handling, the libraries *JDOM* and *JAXEN* are used since they provide high level functions for dealing with *XML* based contents, which

speeds up the development significantly. For reading the *EXIF* information stored in the images Drew Noakes' *exifExtractor* classes were used.

Since NWCBIR is a Java Swing application, the designing started with creating a user interface that divides the annotation methods from the image preview and file browsing mechanisms. The annotation methods were separated from each other in extending a *JPanel* GUI element for each method or logical group of methods. As shown in figure 1, there are panels for creating the *ColorLayout* and *ScalableColor* descriptor, which are extracted from the image on first loading. There is the so called "creation panel" which shows the *EXIF* tags and values and holds the creator of the image and there are the "metadata description panel" for defining version and author

of the metadata description. The "quality rating panel" is used for assigning a quality value and defining the person who rated the image quality, and the "text annotation panel" allows the input of a simple textual description of the image contents. Since a series of photos should be annotated in short time the file browsing tool is a specialized table, which allows the user to select the image in a fast and intuitive way. Obviously, a preview panel is also required to allow the user to examine the image, but also a full size preview has been implemented as well as the possibility to define an external image viewer, which can be called using a keyboard command to give the user the ability to use his favorite tools.

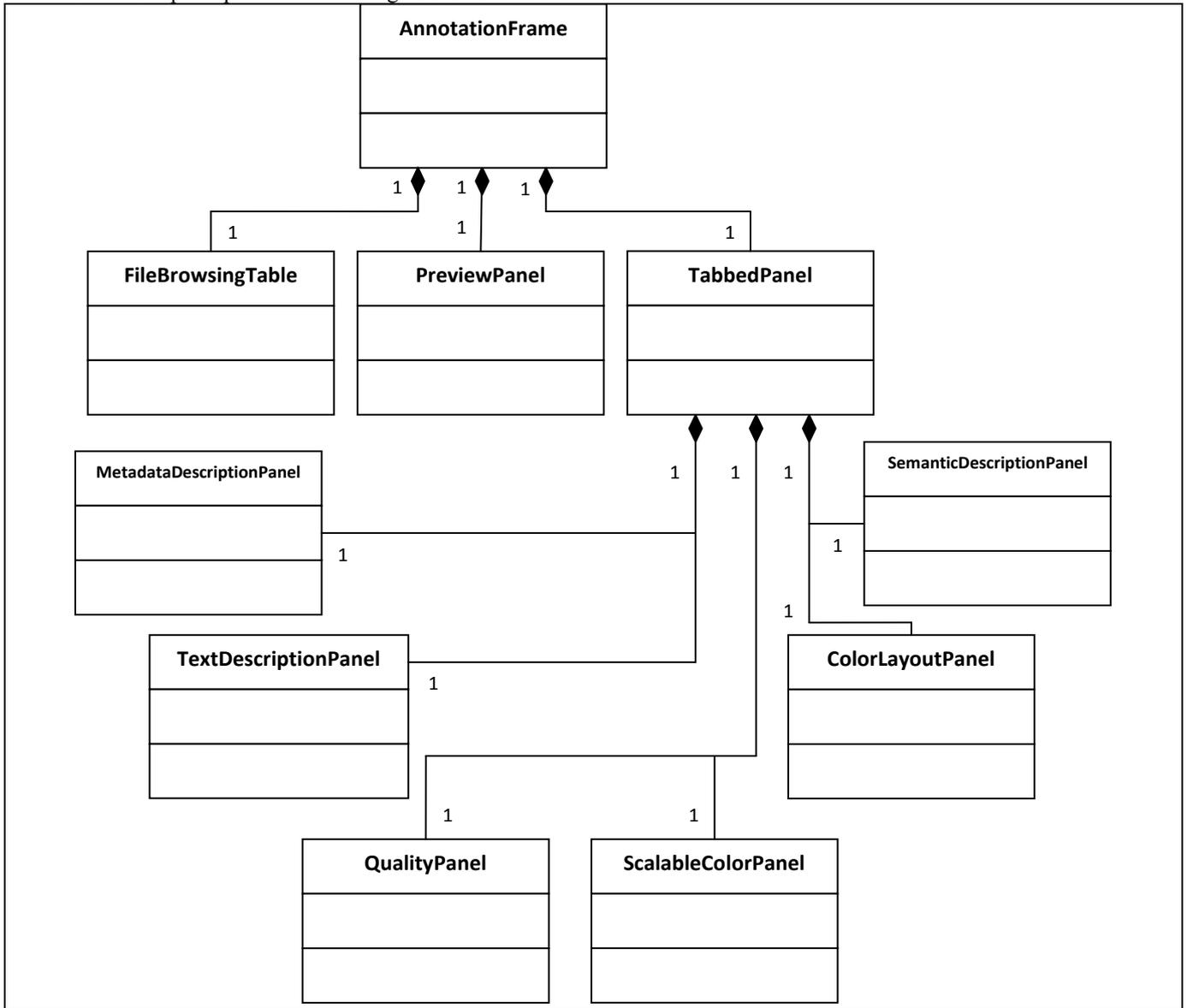


Fig. 1: Simplified UML diagram of NWCBIR System's Annotation Tool

Central part of Annotation Tool is the so called "semantic description panel". It allows the user to define semantic

objects like agents, places, events and times which are saved on exit for reusing them the next time starting NWCBIR.

These semantic objects can also be imported from an existing MPEG-7 file to allow exchange of objects between users and editing and creating those objects in a user preferred tool. Semantic objects can be used for creating the description by dragging and dropping them onto the blue panel with the mouse, shown in figure 2. While testing this model we experienced, that this model is sufficient for users who often take pictures of the same persons and objects, and

who take pictures in series, which is quite often the case with hobby and amateur photographers. As once the objects exist, they can be reused if some pictures or series have the same context. This is especially true for objects representing persons, animals and places like the relatives, colleagues, friends, favorite pets or places like “at home” or “at work”.

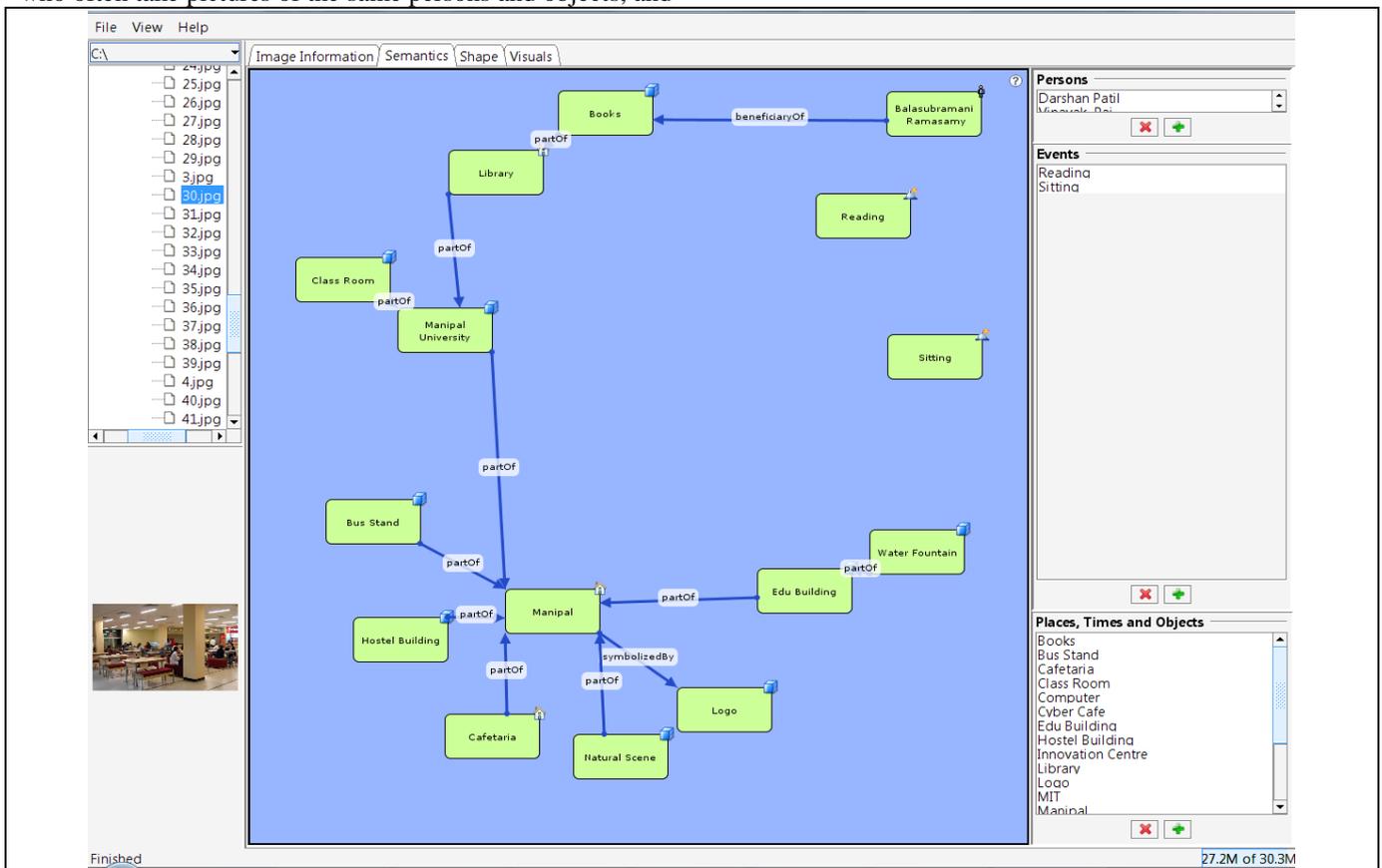


Fig. 2: Creating a semantic description using NWCIBR System by drawing a graph as abstraction of the semantics.

After dropping all the needed objects onto the blue panel the user can interconnect these objects by drawing relations between them using the middle mouse button. The graph, which is generated through these user interactions with NWCIBR, can be saved as part of an MPEG-7 description. In addition to the ability to create a new graph, NWCIBR is also a tool for importing, editing and deleting existing graphs or sub graphs.

Further a whole series can be pre-annotated for simplifying and speeding up the task of annotating multiple images. All images within the same context are placed in one file system folder and the user opens the first one using NWCIBR.

After defining a “base” description which is the same for all images of the series like the creator, a base textual description like “on our visit to Pudukkottai” and a base graph including the location where the photos were taken and time and motivation when they were taken. Finished with this minimal description the so called “autopilot” can be used, which opens all images in the defined folder sequentially, calculates the visual descriptors, which is a rather time consuming task depending on the size and resolution of the image, extracts the *EXIF* data and image specific parameters, creates a thumbnail instance of the image for later retrieval and saves the base description.

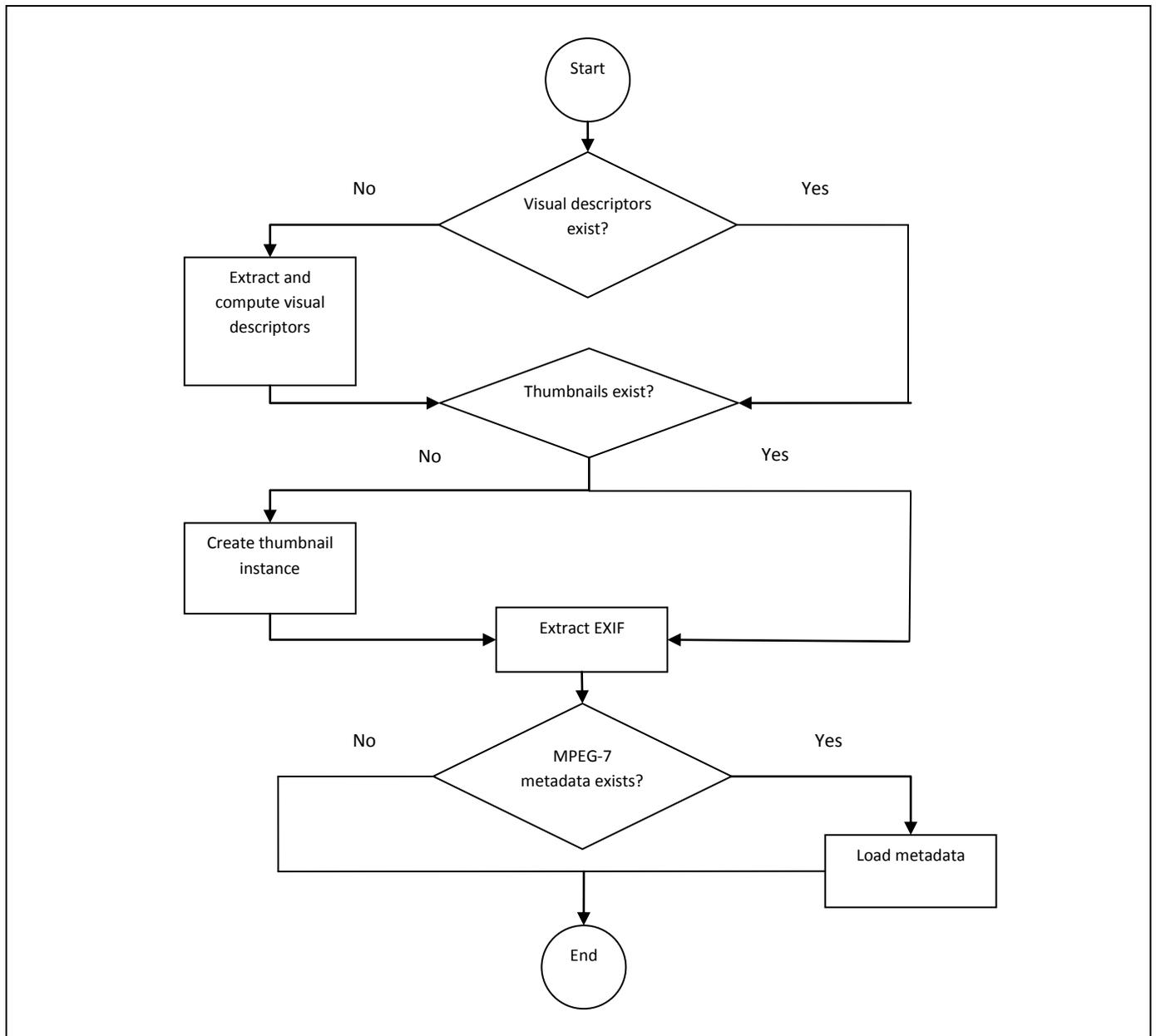


Fig. 3: Flow chart showing the Annotation Process in NWCBIR.

An obviously positive effect is that, when opening one of the pre-annotated photos afterwards, the thumbnail instance and the visual descriptors can be read from the existing metadata and do not have to be created, which saves time when opening a single image for editing. The entire annotation process is shown in the flow chart figure 3.

Inside an MPEG-7 document the *MediaProfile* descriptor is used to reference instances of the media, described by the metadata. As well as the original image, which is referenced in the master profile, a thumbnail instance, created by NWCIBR if not already present, is referenced in another *MediaProfile*.

IV. RESULTS

Annotating digital photos is a very time consuming task. A very common problem is the extraction of existing and computable metadata like *EXIF* information and the visual descriptors like *ColorLayout*, *ScalableColor* and *EdgeHistogram*. The time used for extracting a visual descriptor is proportional to the resolution of the image. This time can be easily reduced by using a faster computer or extracting the metadata on a server or parallel to the interactions of the user.

Another problem is the interactive creation of the graph by the user. The creation of the main objects takes a lot of time. In another project, where all data comes from a specific context, a pre-built ontology based catalogue of semantic objects is used, which includes at least 95 percent of the needed objects. Using a catalogue like this in the context of a personal digital photo library does make sense, but it has to be updated and extended successively. If the user only takes photos in small numbers and on rare occasions, like two on a birthday party, three on this holidays, and one on his car newly washed, administrating and enhancing a catalogue of semantic objects demands more effort than typing in a textual description for each photo. Besides, in this case ability for retrieving annotated images will not be needed, because the number of photos will not exceed the critical mass for overlooking all of the images.

The graphical user interface of the annotation is, since it is a prototype, more or less an abstraction of the MPEG-7 descriptors, which is not intuitive for a user, who does not

really know about MPEG-7, so there has to be done a lot of work “hiding” the MPEG-7 from the user.

Existing metadata should not be lost while annotating the photos, but included in the MPEG-7 document. There are various ways in storing additional information inside an image, the two most common are *EXIF*, which is used by most digital camera manufacturers to save technical data about the photo, and a standard created by the *IPTC*, which is used for instance by the popular application Adobe Photoshop. The first one is very common and Java libraries for reading this information exist, while for the second one no Java implementation exists. A very interesting effect is that *EXIF* obviously allows the creator of the metadata to store the same information in different ways, which complicates a camera independent implementation. We experienced that Sony decided to store three tags for defining the time when the Photo was taken by using the tags “DateTime”, “DateTimeDigitized” and “DateTimeOriginal”, while Kodak only used the third one.

V. CONCLUSION

MPEG-7 matches many of the current requirements for a metadata standard for usage in a personal digital photo library and it defines a lot more useful descriptors, which could be integrated as features in such libraries. In addition it is not only a standard for describing the content of images, but it also defines ways to annotate video and audio documents and it is prepared for general usage with multimedia data.

VI. REFERENCES

- [1] Semantics of Multimedia in MPEG-7, Benitez Ana B. Et al, 2002, International Conference on Image Processing 2002, Proceedings Volume: 1, Page(s): 137-140
- [2] Image Description and Retrieval Using MPEG-7 Shape Descriptors, Carla Zibreira, Fernando Pereira, ECDL 2000, Proceedings, pp. 332-335
- [3] Everything You Want to Know About MPEG-7, F. Nack, A. Lindsay, Part 1, IEEE Multimedia, 6(3), July-September 1999, 65-77
- [4] Multimedia Content Description Interface – Part 5: Multimedia Description Schemes, MPEG, 23.10.2001
- [5] Proposal for Integration of DublinCore and MPEG-7, Hunter, Jane, October 2000.

An Experimental Study to Identify Qualitative Measures for Website Design

G. Sreedhar

Senior Lecturer in Computer Science
Dept. of Computer Science
Rashtriya Sanskrit University
Tirupati
gsrid74@yahoo.com

Dr. A.A.Chari

Professor in OR & SQC
Dept. of OR & SQC
Sri Krishnadevaraya University P.G Centre
Kurnool
chari_anand@yahoo.com

Abstract-The primary goal of this paper is to analyze the quality of Website design of various Websites of universities in India and identifying the causes that affect the quality of Website design. The Website of each university is scanned using W3C guidelines which are the bases for any type of Web application. The parameters such as Web page size, down loading time, broken links, Web page errors etc., are considered in identifying the qualitative measures for Website design.

Different kinds of tools are used to examine the components of Websites of various Indian Universities. These tools include: W3C Link checker, W3C Markup Validation Service, Webpage Analyzer and Website Extractor. The W3C Link checker accepts URL address of Web page and parses each and every hyperlink to find broken links in the page. The W3C Markup Validation Service finds the errors regarding HTML tags' usage errors, properties of Web page and standards of the Web page mentioned by W3C Consortium. The Webpage Analyzer finds the number of objects used in each Web page, Web page size, downloading time etc., The Website Extractor extracts URL addresses of all Web pages of the Website.

Keywords: Website, Page Size, Download time, Web page errors, W3C Link checker, W3C Markup Validation Service, Webpage Analyzer and Website Extractor.

I. INTRODUCTION

A Website is a collection of Web pages containing text, Images, audio and video etc. Thus Web is a vast collection of completely uncontrolled documents. Today, Web is not only an information resource but also it is becoming an automated tool in various applications.

Due to the increasing popularity of WWW, one can be very cautious in designing the Website. If the Website is not designed properly, the user may face many difficulties in using the Website. For example, if a student wants to join a course in a university through online mode, the Website must provide maximum facilities to the candidate so that he do not get any difficulty in admission process. To design a Website with high quality, one has to follow certain guidelines for achieving the quality Web design. Despite of many recommendations, ideas and guidelines, designing a quality Website is still a burning problem. It [1] is suggested that always Web design is continuous process. The authors Flanders, Vincent and Michel Wills [2] said that always design should be improved into good by looking from bad design. As a part of this tedious work, here I am

trying to find out various qualitative measures from the existing design. This paper presents various aspects on analyzing the quality of Website design. The research work was done with a case study.

II. TOOLS USED IN ANALYZING PROCESS

A case study was conducted on Indian universities' Websites related to the structure, content and other functional aspects. The main modules of each university's Website are Departments, courses, administration, staff, library, admissions, examinations etc. Screen shots of some of the universities are shown in Figure 1.

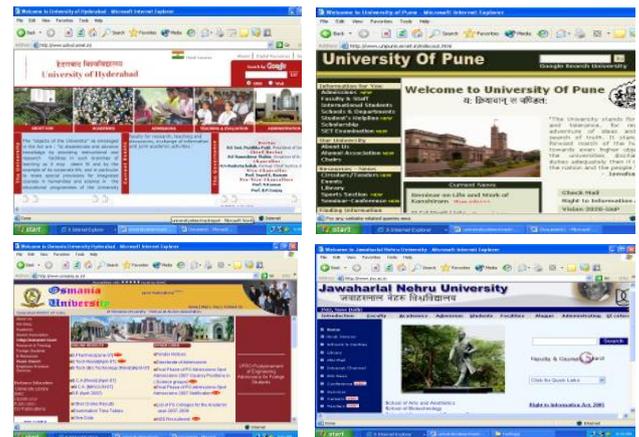


Fig. 1. Snapshots of some universities: An Example

Analysis was carried out using various tools. These include W3C link checker, W3C Markup Validation Service, Web Page Analyzer and Website Extractor.

A. W3C Link Checker:

The W3C Link checker [3] finds number of broken links in the Website. It accepts the URL address of Web page and parses each and every hyperlink in the page. It finds the status code of each link and by using the status code it identifies the broken links related to the page. A Screen shot of W3C Link checker was shown in the following Figure 2.

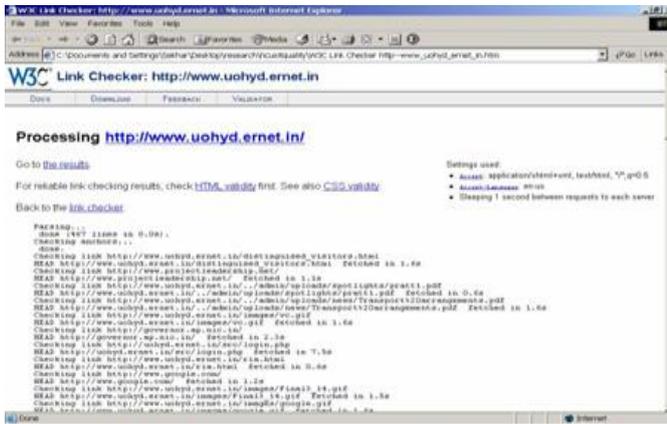


Fig. 2. Link Checker to find broken links

B. W3C Markup Validation Service:

The W3C Markup Validation Service [4] finds the errors related to the HTML pages. It validates the Web page regarding errors in HTML tags, properties of Web page and standards of the Web page mentioned by W3C organization. A screen shot of W3C Markup Validation Service is shown in Figure 3.

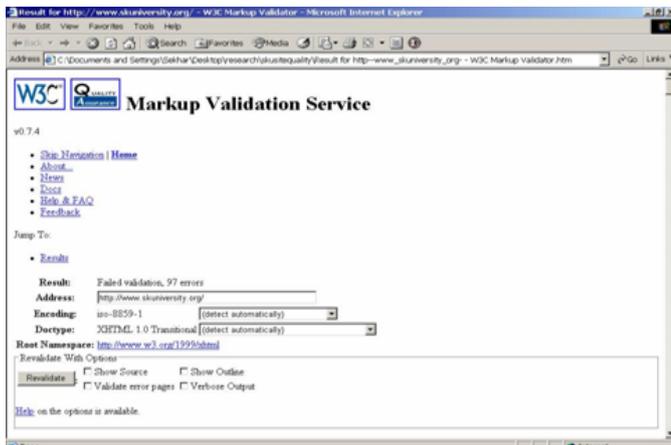


Fig. 3. HTML Validator to find markup errors of various Web pages of university Website

C. Web Page Analyzer:

The Web Page Analyzer [5] finds the number of objects used in each Web page, Web page size and downloading time of all objects. It accepts URL address of a Web page and generates a report containing details like number of image files, number of HTML files, number of script files, down load time etc., of the Web page. A screen shot of Webpage Analyzer is shown in Figure 4.

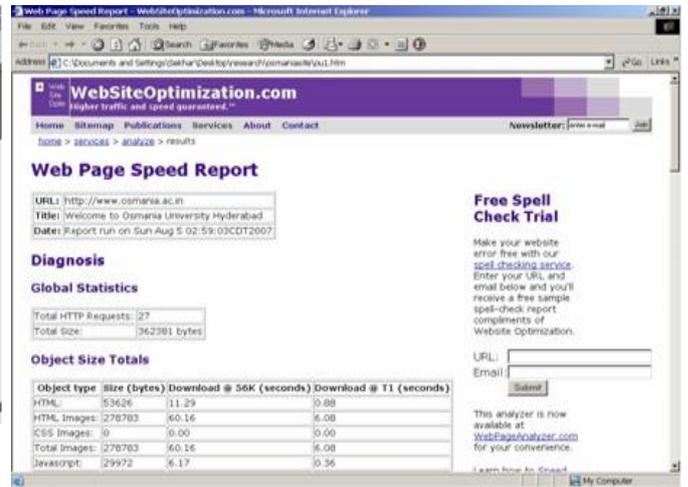


Fig. 4. Website Optimization using Web Page Analyzer

D. Website Extractor:

The Website Extractor [4] extracts the all the components of Website. It accepts Website address and produces URL addresses of all Web pages. The snapshot of Website extractor is shown in figure 5.



Fig. 5. Website Extractor to extract all Web pages of university Website

III. GUIDELINES FRAMED FROM W3C:

World Wide Web Consortium (W3C) [6], [7] defines a set of guidelines for quality Web design. All guidelines are summarized into 12 guidelines for simplicity. Every guideline provides a technique for accessing the content of Website. The guidelines are as follows.

Guideline 1: Provide a text equivalent for every non-text element. This includes images, graphical representations of text, image map regions, animations, applets and programmatic objects, frames, scripts, spaces, audio and video files.

Guideline 2: Do not rely on color scheme only. The content of Web page must match with foreground and background color. Also provide sufficient contrast to the content for visibility.

Guideline 3: Use markup and style sheets instead of images to convey information. Style sheets control the layout and

presentation of the Web page and decreases the download time of the Web page.

Guideline 4: Clearly mention the text information of Web page with natural language. Specify the expansion of each abbreviation or acronym in the document.

Guideline 5: Use tables properly in the Web document. For data tables, clearly specify row and column headers and number of rows and columns exactly.

Guideline 6: Ensure that Web pages featuring new technologies transform gracefully. When dynamic contents are updated, ensure that content is changed. Ensure that pages are available and meaningful when scripts, applets or other programmatic objects are not supported by the browsers. If this is not possible, provide equivalent information as alternative in the Web page.

Guideline 7: Ensure user control of time sensitive content changes. Until user agents provide the ability to stop the refresh, do not create periodically auto-refreshing pages.

Guideline 8: Ensure direct accessibility of embedded user interfaces. Make programmatic elements such as scripts and applets directly accessible or compatible with assistive technologies.

Guideline 9: Design for device-independence. Ensure that any element that has its own interface can be operated in a device-independent manner.

Guideline 10: Provide context orientation information. Title each frame to facilitate frame identification and navigation. Divide large blocks of information into more manageable groups wherever appropriate.

Guideline 11: Provide clear navigation mechanisms. Clearly identify the target of each link. Provide information about the general layout of a site such as site map or table of contents.

Guideline 12: Ensure that documents are clear and simple. Create a style of presentation that is consistent across pages.

IV. METHODOLOGY

The study was conducted on nearly 50 Indian Universities' Websites and considering approximately 5000 Web pages. A Web program was developed to study each university's Website. The Web program consists of four modules: Website Extractor, Link Checker, HTML Validator and Web Page Analyzer. The URL address of each Website is thoroughly scanned using Website Extractor to get all Web pages of Website and Web pages of each university are stored in separate files. A Website is verified with Link Checker module to get number of broken links in the Website. The components that include: text, images, forms, graphics, audio and video files etc., and download time of Web page are gathered using Web page analyzer and stored in separate file. The errors of Web page related HTML tags are traced using W3C HTML Validator and they are stored in files. The overall structure of Web program is shown in Figure 6.

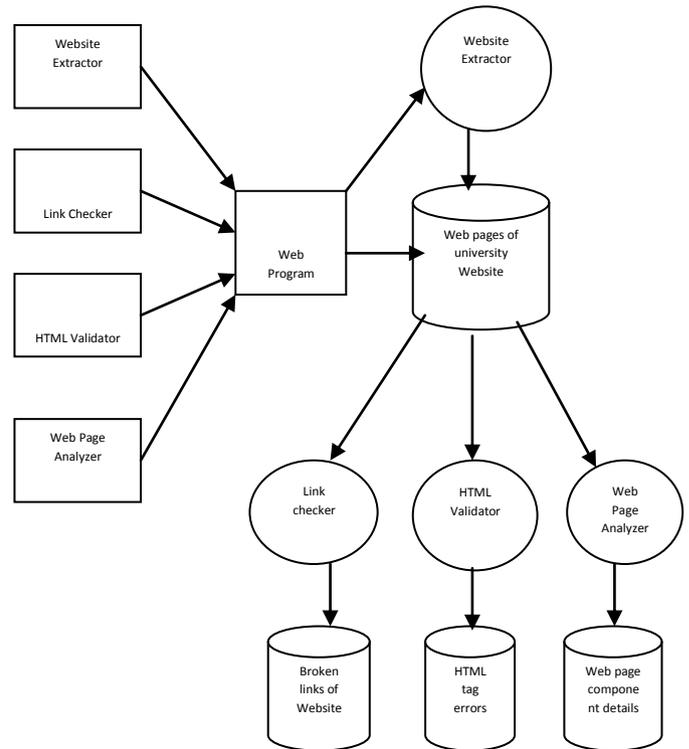


Fig. 6. Architecture of Web program

The components of some universities' Websites such as number of Web pages, size, number of errors, download time, broken links etc., are summarized in the table 1.

V. WEBSITE ERRORS

The Web page errors that are generated using Web program are considered to identify the measures for quality Website design. These errors are further divided into major and minor errors using statistical techniques [8].

A. Major errors:

The major errors directly affect the quality of Web site design and developers must concentrate on this category of errors and these should be eliminated. The major errors include: broken links, document type declaration errors, applet usage errors, server connectivity errors, image load errors, frames tag usage errors and title tag with no keyword errors. The major errors are proportional to the down load time of the Web pages. If major errors are minimized then down load time will be automatically reduced and hence it leads to the better quality. The major errors of some universities' Websites are shown in table 2. The figure 7 shows the graph that depicts different major errors and their effect on Website design.

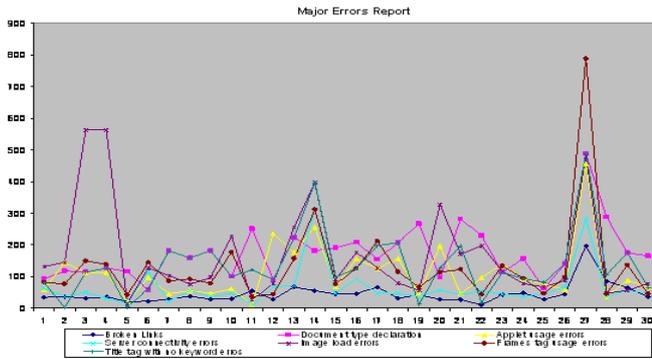


Fig. 7. Graph showing major errors of Websites of various universities

B. Minor errors

The minor errors are HTML tag errors and these may cause incorrect display of some components of Web pages. The minor errors include: table tag errors, body tag errors, image tag errors, head tag errors, font tag errors, script tag errors, style tag errors, form tag errors, link tag errors and other tag errors. The developers must be attentive so that Web pages can be properly designed with appropriate HTML tags. The minor errors of some universities' Websites are given in table 3. The graph in figure 8 shows various minor errors of various universities' Websites.

Sno	Measures to be evaluated	Errors considered	
		Minor errors	Major errors
1	Text formatting measures	BTE, FTE, HTE	
2	Link formatting measures	LTE	Broken links
3	Page formatting measures	TTE, FTE, StTE, FoTE	Frame tag usage errors, document type usage errors
4	Graphics element measures	ITE, BTE	Image load errors
5	Page performance measures	FmTE, STE,	Title tag with no keyword errors
6	Site architecture measures	STE	Applet usage errors, server connectivity errors, down load time of Website, broken links

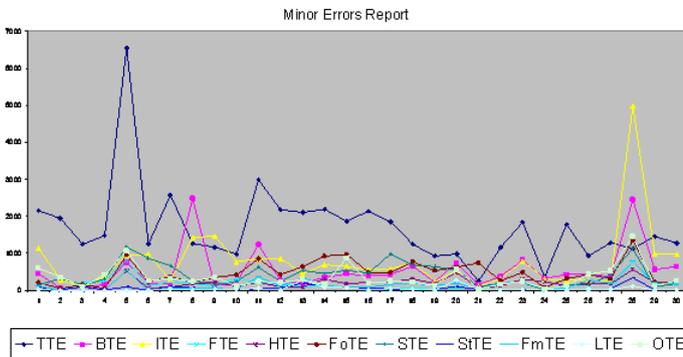


Fig. 8. Graph showing minor errors of Websites of various universities

Sno	University Name	Website address	No. of Web pages	Total Web pages size	Total no. of Web page errors in Website	Average no. of errors in each page	Download time at 28 Kbps (in secs)	No. of broken links
1	Anna University	www.annauniv.edu	117	6187357	2145	18.33333	1713	21
2	Bangalore University	www.bub.ernet.in	659	18297162	19555	29.67375	5065	117
3	Bharatiar University	www.b-u.ac.in	182	7571965	5568	30.59341	2096	29
4	Indira Gandhi National Open University	www.ignou.ac.in	252	9881675	4573	18.14683	2736	56
5	JNT University	www.jntu.ac.in	91	5663958	3885	42.69231	1568	28
6	Jawaharlal Nehru University	www.jnu.ac.in	224	8749265	8894	39.70536	2422	67

Table 1

Sno	University	Broken Links	Document type declaration errors	Applet usage errors	Server connectivity errors	Image load errors	Frames tag usage errors	Title tag with no keywords errors	Total Major Errors
1	Anna University	31	115	112	51	563	150	115	1137
2	Bangalore University	21	117	20	16	15	43	3	235
3	Bharathiar University	29	182	46	23	104	87	182	653
4	Indira Gandhi National Open University	29	182	47	37	97	79	182	653
5	Jawaharlal Nehru Technological University	31	101	62	48	228	178	101	749
6	Jawaharlal Nehru University	56	252	12	13	27	36	122	518

Table 2

Sno	University	TTE	BTE	ITE	FTE	THE	FoTE	STE	StTE	FmTE	LTE	OTE	TotME
1	Anna University	1245	75	97	0	0	154	143	98	0	0	98	1910
2	Bangalore University	6543	759	941	798	521	957	1168	97	534	0	1067	13385
3	Bharatiar University	2567	168	268	241	87	369	675	97	73	43	327	4915
4	IGNOU	1165	126	1467	114	219	347	148	26	46	32	365	4055
5	JNTU	976	214	765	320	118	421	241	23	26	32	112	3248
6	Jawaharlal Nehru University	2987	1236	869	225	224	859	621	46	356	45	248	7716

Table 3

TTE: Table Tag Errors

BTE: Body Tag Errors

ITE: Image Tag Errors

FTE: Frame Tag Errors

HTE: Head Tag Errors

FoTE: Font Tag Errors

STE: Script Tag Errors

SITE: Style Tag Errors

FmTE: Form Tag Errors

LTE: Link Tag Errors

OTE: Other Tags Errors

TotME: Total Minor Errors

VI. IDENTIFYING QUALITATIVE MEASURES FOR WEBSITE DESIGN

The errors that are found in Websites' of various universities lead to the necessity of qualitative measures for effective Website design. The head tag errors (HTE), font tag errors (FoTE) and body tag errors (BTE) identify the problems in the text elements of we page. Thus Text formatting measures are to be evaluated. The image tag error

(ITE), body tag errors (BTE) and image load errors related to image identifies the errors in display of images and hence Graphic element measures to be evaluated. The table tag errors (TTE), frame tag errors (FTE), style tag errors (StTE), font tag errors (FoTE), frame tag usage errors and document type declaration errors cause the invention of page formatting measures. Link Tag Errors (LTE) and broken links identify the need of link formatting measures. The form tag errors (FmTE), script tag errors (STE) and title tag with no keyword errors identify the need of page performance measure. The script tag errors (STE), applet usage errors, server connectivity errors, down load time of Website and broken link errors contribute the need of Website architecture measure. All these measures are shown in table 4.

Table 4

Sno	Measures to be evaluated	Errors considered	
		Minor errors	Major errors
1	Text formatting measures	BTE, FTE, HTE	
2	Link formatting measures	LTE	Broken links
3	Page formatting measures	TTE, FTE, STE, FoTE	Frame tag usage errors, document type usage errors
4	Graphics element measures	ITE, BTE	Image load errors
5	Page performance measures	FmTE, STE,	Title tag with no keyword errors
6	Site architecture measures	STE	Applet usage errors, server connectivity errors, down load time of Website, broken links

VII. CONCLUSION

This paper aims to investigate into various measures required for quality Website design. A focused approach has been made to identify all possible errors in developing Website so that before going to use any qualitative measure, it is necessary to verify whether all aspects of Website design are considered in quality assessment or not. This would enable to adjudge the quality status of Web design of the various universities and would indicate the necessity of improvement in the design of the Website. We can extend this work to develop a set of metrics specifically in higher educational institutions' Websites using the results of the study.

VIII. REFERENCES

- Yogesh Deshpande, San Murugesan Arthla Ginige, Steve Hanse, Daniel Schwabe, Martin Gaedke, Bebo White, "Web Engineering", Journal of Web Engineering, Vol.1. No.1 (2002) 003-017.
- Floders, Vincent and Michel Will, "Web Pages that suck: Learn Good Design by looking at Bad Design", San Fransico, CA, SYBEX.
- W3C Link Checker, <http://validator.w3.org/checklink>
- W3C Markup Validation Service, <http://validator.w3.org/>.
- Web Page Analyzer - 0.98 - from Website Optimization (Free Website Performance Tool and Web Page Speed Analysis) <http://www.Websiteoptimization.com/services/analyze/>
- Website Extractor 9.80, <http://wareseeker.com/screenshot/Website-extractor-9.80.exe/420871>
- Techniques for Web Content Accessibility Guidelines by W3C, <http://w3.org>.
- PaoloTonella and Filippo Ricca, "Dynamic Model Extraction and Statistical Analysis of Web Application", Proceedings of the IEEE Fourth International workshop on Website evolution, 2002.

Process modeling using ILOG JViews BPMN Modeler tool to Identify Exceptions

First A. Saravanan. M.S, Second B. Rama Sree. R.J

Abstract - Today all the Business analysts uses Business Process Modeling Notation (BPMN) to model business process diagrams. Business process modeling is the activity of representing processes of an enterprise, which allows the business analyst to focus on the proper sequence flow, of the business processes, without concerning himself / herself on the proper implementation of the process; e.g., be more concerned that a 'Sales or Purchase' process includes delivering or receiving the items and not how the items will be delivered or received. These strengths of BPMN allow businesses to increase efficiency by automating part of their business processes using Business Process Modeling Notation and by giving a clear representation and analysis of their business process, using business process diagrams (BPDs) to identify the unhandled exceptions. The Business process modeling tools provide business users with the ability to model their business processes, implement and execute those models. This paper presents a simple, yet instructive example of how an ILOG JViews BPMN Modeler tool can be used to identify and verify exceptions for a "Deliver Items" business process.

Keywords - Exception, Modeling, Notation, BPMN, BPD

I. INTRODUCTION

BPMN stands for business process modeling notation. It is a new standard for modeling business processes. BPMN has a diagram called the Business Process Diagram (BPD). Business process modeling is the activity of representing processes of an enterprise, which allows the business analyst to focus on the proper sequence flow, of the business processes [1]. A goal for the development of BPMN is that the notation be simple and adoptable by business analysts. Also, there is a potentially conflicting requirement that BPMN provide the power to depict complex business processes. [2]

The BPMN business process diagram has been designed to be easy to use and understand but also provides the ability to model complex business processes [3]. The execution of a business process often includes multiple entries. These entries are not under the control of the process. Because of the process details or complexity of the real world, their behavior

details cannot always be predicted. At final, a business process may have a single ideal execution path.

In practice many executions of process will encounter events, i.e., errors or missing deadlines that lead the process, off this path. The exception handling is not a favorite issue for programmers or analysts. They often focus on the likely or ideal business scenarios and end up ignoring the handling of diverse error conditions. Therefore, we used the BPMN tool to model the business process with unhandled exceptions.

The main goal of BPMN is to provide a notation that is readily understandable by all business users. This includes the business analysts that create the initial drafts to identify and verify the sequence of operations with minimal error possibilities likely called exceptions, occurs during the product implementation.

II. RELATED WORK

Exception handling is a one of the programming constructs [4], which occurs during the execution of a program that interrupts the normal flow of the program's instructions [5]. To achieve better quality product the Exception handling is acting as an interface between programmer and languages [6].

The Exceptions can be identified during the software modeling phase to avoid product failure during product implementation. The software development industry uses their own modeling technique without any standard, then after the introduction of BPMN; the entire software industry system analyst are started using "An Industry Standard for Process Modeling" [7].

The first commercial edition of BPMN 1.0 released in May'2004 [3], the BPMN 1.0 specification was released to the public, February'2006 and BPMN1.0 was adopted as an OMG standard [8]. Currently there are thirty-nine companies that have implementations of BPMN.

Prior to BPMN there were many of the process modeling tools and methodologies; i.e., all sorts of visual business process flow-chart formats were used. After the business analysts focus on BPMN's Business Process Diagram (BPD) [3], most of the software development system analysts were benefited more to produce quality product with good design and modeling. The problems with different representations of older system has created some problems, they are

- Business analysts are required to understand multiple representations of business processes.
- Business participants that don't share the same graphical notations might not understand one another.

F.A. Author is doing Research in Research and Development Centre, Bharathiar University, Coimbatore – 641 046. HOD and Associate Professor in Department of Computer Applications, VRN College of Computer Science and Management, Affiliated to S.V. University, Tirupati – 517 501, Andhra Pradesh, INDIA; (e-mail: saranenadu@yahoo.co.in).

S.B. Author is working as a Reader in Department of Computer Science, Rashtriya Sanskrit University, Tirupati – 517 501, Andhra Pradesh, INDIA; (e-mail: rjramasree@yahoo.com).

- A technical gap between the format of the business process initial design and the format of the languages that will execute these business processes.
 - (1) BPMN provides BPD - to be used by people who model and manage business processes [9].
 - (2) BPMN provides formal mapping to an Execution language of the BPM system to be used by System Analysts who design the Software process execution [9].

III. THE ROLE OF EXCEPTION HANDLING IN BUSINESS PROCESS MODELING

The Exception handling could be the critical focus of process modeling and analysis, in most of the cases you could be the wrong. So, Exceptions are playing major role during the time of process modeling phase and why model business process and when should we use the Exceptions? And can business analyst model the Exception handling? We will discuss these questions in the following sections.

A. Why Model Business Processes?

Companies are finding many reasons to capture their business processes. Companies who have merged want to examine processes across their lines of business to discover which one is the best of breed. Other companies are looking to improve their existing processes, or even to automate them. In some countries, government regulations require that business processes be properly documented. For example, some companies in the United States regulates that certain processes must be well documented. These are among the many factors in the business world today, that are making companies take a closer look at their business processes. [10]

B. When should we Use Exceptions?

The simple answer is: “whenever the semantic and performance characteristics of exceptions are appropriate”. An oft-cited guideline is to ask our self the question “Is this an exceptional or unexpected situation?” This guideline has an attractive ring to it, but is usually a mistake. The problem is that one person’s “exceptional” is another’s “expected”: when you really look at the terms carefully, the distinction evaporates and you are left with no guideline. After all, if you check for an error condition, then in some sense you expect it to happen, or the check is wasted code.

A more appropriate question to ask is: “do we want stack unwinding here?” Because actually handling an exception is likely to be significantly slower than executing mainline code, you should also ask: “Can I afford stack unwinding here?” For example, a desktop application performing a long computation might periodically check to see whether the user had pressed a cancel button. Throwing an exception could allow the operation to be cancelled gracefully. On the other hand, it would probably be inappropriate to throw and *handle* exceptions in the inner loop of this computation because that could have a significant performance impact. The guideline mentioned above has a grain of truth in it: in

time critical code, throwing an exception should *be* the exception, not the rule. [11]

C. Can Business Analyst Model Exception Handling?

In conventional wisdom in business that 80 percentage of the problems are caused by 20 percentage of the work. Certainly the designers of the Business Process Modeling Notation standard had Exception handling in mind from the start. BPMN introduces to process modeling the notion of intermediate events. It's a god awful name but an absolutely essential concept for making exception handling visible in the process diagram and understandable to business people. In fact, events are the key difference between BPMN and traditional flowcharting.

In BPMN, intermediate events are drawn as circles with a double border, with a symbol inside denoting the type of event: receipt of an external signal message, a timeout, a system fault, etc. When drawn attached to the border of a process activity or sub process, the semantics of BPMN say that the activity or sub process is interrupted immediately, and the process continues along the flow path leading out from the intermediate event. That path is called "exception flow." If the event never occurs, the activity or sub process completes normally and processing continues along the flow path leading out from it. That path is called "normal flow." Is that hard to understand? No, I didn't think so. You'd be surprised then to know that a number of modeling tools offered by Business Process Management System (BPMS) vendors that advertise themselves as BPMN compliant don't support intermediate events. What those vendors usually say is that the concept is too technical for business analysts to understand. What they really mean, in most cases that their process engine executes the model to automate and monitor the process flow can't handle them. The modeling tool's simulation engine has no idea what to do with them, either. So they're just left out of the tool. Shame on Object Management Group (OMG) for allowing this pseudo-BPMN to reproduce as it has.

The traditional alternative to modeling exception handling explicitly in the process diagram is to do it in code, toss it over the wall to IT. Once in a while this might be necessary, but as a general principle it's just plain wrong. By removing exception handling from the process model and burying it in implementation code, you've not only lost visibility into what's going on, you've lost the ability to use it in simulation analysis, you've lost agility, reuse, shared best practices, etc. All of those assume exception handling is in the process model. [12].

IV. BPMN TOOLS HISTORY

The current working routines of the business analyst models a diagram on a piece of paper, but you won't get the assistance that only quality software has to offer. We reviewed several business process modeling applications, ILOG JViews BPMN Modeler tool by ilog.com [13], BizAgi process modeler by bizagi.com [14], eBPMN Designer by soyatec.com, Business Process Visual

ARCHITECT by visual-paradigm.com, Business Process Modeler for Business Analysts by eClarus.com, Tibco Business studio by Tibco.com, and Intalio Designer by intalio.com.

These modeling tools follow the BPMN specification. Some may add extensions that fit their users' needs. They employ static verification to force the user to keep within the BPMN constraints; much like a word processor employs spell checking to warn against mistakes. Since BPMN specifications provide that exceptions must be handled and force its user to handle exceptions particularly ILOG JViews BPMN Modeler tool [13].

A. A First Look at ILOG JViews BPMN Modeler Tool

The ILOG JViews BPMN Modeler introduced different versions of modeling tools.

The Latest version of ILOG JViews BPMN Modeler 1.1.2 tool was introduced in the beginning of year 2009. It is very easy to use; in a matter of minutes you will be able to begin defining your processes and collaborate with other people in your organization.

Generally, BPMN specified a single business process diagram, called the Business Process Diagram (BPD) [3]. This diagram was designed to do two things well.

First, it is easy to use and understand. You can use it too quickly and easily model business processes, and it is easily understandable by non-technical users, usually management [15].

Second, it offers the expressiveness to model very complex business processes, and can be naturally mapped to business execution languages [15]. To model a business process flow, you simply model the events that occur to start a process, the processes that get performed, and the end result of the process flow. Business decisions and branching of flows is modeled using gateways. A gateway is similar to a decision symbol in a flowchart.

Furthermore, a process in the flow can contain sub-processes, which can be graphically shown by another business process diagram connected via a hyperlink to a process symbol. If sub-processes do not decompose processes, it is considered a task the lowest level process. A '+' mark in the process symbol denotes that the process is decomposed; if it doesn't have a '+' mark, it is a task.

As you drive further into business analysis, you can specify 'who does what' by placing the events and processes into shaded areas called pools that denote who is performing a process. You can further partition a pool into lanes. A pool typically represents an organization and a lane typically represents a department within that organization, although you may make them represent other things such as functions, applications, and systems.

B. ILOG JViews BPMN Modeler tool Events and Notations

The ILOG JViews BPMN Modeler tool follows the general modeling notations supported by business process modeling. During business process modeling, you model the events that happen in the business, and show how they affect

process flows. An event either kicks off a process flow [16], or happens during a process flow, or ends a process flow. BPMN provides a distinct notation for each of these types of events, shown in the Table I, below.

TABLE I.
BASIC EVENT TYPES IN BPMN AND THEIR NOTATIONS.

Start Event		Intermediate Event		End Event	
Starts a process flow.		Happens during the course of a process flow.		Ends a process flow.	

When you model more complex process flows, such as B2B web services, you need to model more complex business events, such as messages [16], timers [16], business rules [16], and error conditions. BPMN enables you to specify the trigger type of the event, and denote it with a representative icon, as specified in Table II. Specifying a trigger type to an event puts certain constraints on the process flow that you are modeling, which are explained in the table. For example, a timer cannot end a process flow. You can only draw message flows from and to message events. These types of modeling rules, which are actually kinds of business rules, should be enforced automatically by the modeling tool providing support for BPMN. Oftentimes an event happens while a particular process is being performed, causing an interrupt to the process, and triggering a new process to be performed. The process will complete, causing an event to start, and a new process to be performed. You can model these intermediate events by placing an event symbol directly on the process. The different events were available in the event toolbar and gives access to several types of event that can occur within a BPMN process, For example, message, timer, exception, cancel, compensation, rule, link, multiple, signal and terminate etc., are available in the ILOG JViews BPMN Modeler events toolbar.

TABLE II.
ADVANCED EVENT TRIGGER TYPES IN BPMN AND THEIR NOTATIONS.

Start Events	Intermediate Events	End Events	Description
Start Message 	Message 	End Message 	A start message arrives from a participant and triggers the start of the process, or continues the process in the case of an intermediate event. An end message denotes a message generated at the end of a process.
Start Timer 	Timer 	A Timer cannot be an End Event.	A specific time or cycle, for example every Monday at 9am can be set to trigger the start of the process, or continue the process in the case of an intermediate event.
Start Rule 	Rule 	A Rule cannot be an End Event.	Triggers when the conditions for a rule become true, such as "Stock price changes by more than 10% since opening."
Start Link 	Link 	End Link 	A link is a mechanism for connecting the end event of one process flow to the start event of another process flow.
Start Multiple 	Multiple 	End Multiple 	For a start multiple event, there are multiple ways of triggering the process, or continuing the process in the case of the intermediate event. Only one of them is required. The attributes of the event define which of the other types of triggers apply. For end multiple, there are multiple consequences of ending the process, all of which will occur, for example, multiple messages sent.
An Exception cannot be a Start event	Exception 	End Exception 	An end exception event informs the process engine that a named error should be generated. This error will be caught by an intermediate exception event.
A Compensation event cannot be a Start event	Compensation 	End Compensation 	An end compensation event informs the process engine that compensation is necessary. This compensation identifier is used by an intermediate event when the process is rolling back.
An End event cannot be a Start event	An End event cannot be an Intermediate event	End Cancel 	An end event means that the user has decided to cancel the process. The process is ended with normal event handling.
An End Kill event cannot	An End Kill event cannot	End Kill 	An end kill event means that there is a fatal error and that all activities in the process should be immediately

be a Start event	be a Intermediate event		ended. The process is ended without compensation or event handling.
------------------	-------------------------	--	---

C. ILOG JViews BPMN Modeler tool activity classification legend

To implement ILOG JViews BPMN Modeler tool to any type of business process, the following activity classification legends are used.

1. Query data (Example: Find Orders)
 2. Enter data (Example: Log received items)
 3. Update data (Example: Log received items)
 4. Produce data (Example: Perform Regression Test)
 5. Send notification (Example: Notify customer RMA number is invalid)
 6. Receive notification (Example: Receive Report State of Accounts)
 7. Send and Receive data (Example: Notify customer)
 8. Analyze data (Example: Allocate Defects)
 9. Perform action (Example: Negotiate return)
- s - Sub-Process
g - Gateway
x - Complex activity

D. ILOG JViews BPMN Modeler tool Exceptions legend

To implement ILOG JViews BPMN Modeler tool to any business process; the following exception legends are used.

- QF - Query failed
- UF - Update failed
- UR - Update rejected
- SF - Send failed
- SR - Send rejected
- RR - Receive rejected (data rejected)
- RR - Receive rejected (authorization)
- RR - Receive rejected (authentication)
- RR - Receive rejected (security)
- NR - Notification rejected
- VR - Analysis/Verification rejected
- AF - Action failed
- TO - Timeout
- Org - The original business process diagram, modeled using a regular modeling tool
- Enh - A user of ILOG JViews BPMN Modeler's modeling tool with suggestions
- Exp - An Industry expert
- #Sug - Number of suggestions for exceptions, given by the enhanced tool, according to classification
- #Sel - Number of exception the user of the enhanced tool selected from the suggestions.

V. EXCEPTION HANDLING EXPERIMENT RESULTS USING ILOG JViews BPMN MODELER TOOL

To prove the importance of ILOG JViews BPMN Modeler tool to model the business process, let us take one of the business process “**Deliver Items**” from a business company. This ILOG JViews BPMN Modeler tool is also used particularly to identify and verify the exceptions of any business process without any difficulties or overhead.

The results show the ILOG JViews BPMN Modeler tool usage and importance of modeling any business process to produce a quality model to do further phases of product development. The ILOG JViews BPMN Modeler tool will organize the processes with the help of business process diagrams.

A. Activity classification with possible exceptions

The “Deliver Items” process has the following table of Activity classification with probable exceptions.

TABLE III. ACTIVITY CLASSIFICATION WITH POSSIBLE EXCEPTIONS.

Business Activity	Classification	Probable Exceptions
Process: Deliver Items		
Delivery with invoice	9	AF
Update invoice	2	UR
Terminate delivery	9	
(pay on delivery?)	g	
Delivery	9	
Receive payment	9;2	
Follow up	9	

B. Business process diagram of “Deliver Items” process

The Business Process Diagram developed for “Deliver Items” process using the ILOG JViews BPMN Modeler tool with different notations.

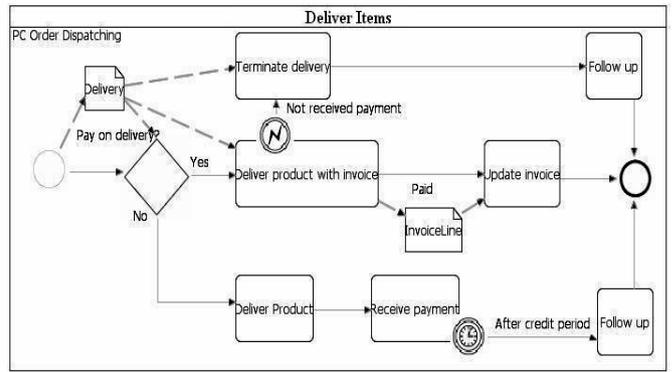


Fig.1. Experiment input: “Deliver Items”

C. Result of “Deliver Items” process with exception

The following table shows the deliver items process with identified and verified exceptions in various stages of the process. It is the out come of deliver items process result, conducted process modeling by the ILOG JViews BPMN Modeler tool and it agree with the possibility or occurrence of exception or failure of process with an expert or system analyst opinion of different software development companies all around the world.

TABLE IV. RESULT OF “DELIVER ITEMS” PROCESS WITH EXCEPTION

Business Activity	Cls	#sug	#sel	Org	Enh	Exp
Delivery with invoice	9	1	1	AF	AF	AF
Update invoice	2	1	1		UR	
Terminate delivery	9	1	0			
(pay on delivery?)	g					
Delivery	9	1	0			
Receive payment	9;2	2	0			TO
Follow up	9	2	0			

In the above “Deliver Items” process the possible or identified exceptions (Enh) by the ILOG JViews BPMN Modeler tool are mostly correlated with the expert (Exp) opinion and the regular software development companies modeling process (Org), for example the “Delivery with invoice” process produced or advised the “Action failed” exception for all these three cases. So, there is no denial that using the ILOG JViews BPMN Modeler tool produces more accurate business processes, according to an expert’s opinion. That is, most of the exceptions that the expert considered probable are handled in the resulting business process.

VI. CONCLUSION

Based on the observation that in a given “Deliver Items” business process produced accurate result, in this paper, we first looked about process modeling then importance of exception with the role of BPMN tool in the business analyst. Our experiment with an ILOG JViews BPMN Modeler tool is very useful to identify and verify the exceptions at the time of modeling the product. So we can produce good product with performance and quality. It adds an additional benefit that to reduce the program failure and ease to construct code and test the product at the time of Quality of service. So, this ILOG JViews BPMN Modeler tool assists in producing robust business processes during the product modeling. That is, most of the exceptions that the expert considered probable are handled in the resulting business process, unlike the case of the regular tool.

REFERENCES

- [1] “Business Process Modeling” from Wikipedia.
- [2] “Business Process Modeling Notation Specification”, OMG Final Adopted Specification, February 2006 dtc/06-02-01. pp. 15.
- [3] Martin Owen, Jog Raj, “BPMN and Business Process Management Popkin Software, 2003.
- [4] Peter Abrahams, Practice Leader - Accessibility and usability, Bloor Research Resolution Accelerator – Exception Handling for SOA.
- [5] Barbara G. Ryder, “Influences on the Design of Exception Handling an ACM SIGSOFT project on the Impact of software Engineering
- [6] Ilco of Mayday Software productions. “C++ Exception Handling”, 1983.
- [7] Jan Recker, “BPMN Modeling – Who, Where, How and Why”, 2008.
- [8] Stephen A. White, “Using BPMN to Model a BPEL Process”, IBM Corporation, United States.
- [9] White, S. A, “Introduction to bpmn”, IBM Software group Tutorial, Available: <http://www.bpmn.org/Documents/OMG%20Bpmn%20Tutorial.pdf>, 2005.
- [10] Marc Fasbinder, “Why model business process”, Consulting I/T Specialist, WebSphere Software Technical Sales, IBM, 30 May 2007, Available: http://www.ibm.com/developerworks/web/sphere/library/techarticles/0705_fasbinder/0705_fasbinder.html.
- [11] D. Abrahams, “Exception”, originally published in M. Jazayeri, R. Loos, D. Musser (eds.): Generic Programming, Proc. of a Dagstuhl Seminar, Lecture Notes on Computer Science. Volume. 1766, June 2009.
- [12] Bruce Silver, “BPMS Watch: Can Business Analysts Model Exception Handling”, Principal, Bruce Silver Associates Friday September 15, 2006.
- [13] ILOGS JViews BPMN Modeler 1.1.2, Overview, Available: http://www.ilog.com/products/jviews/diagrammer/bpmnmodeler/usermanual/Content/Visualization/Documentation/JViews/JViews_Diagrammer/usrfreebpmn/_pubskel/index.html, 2009.
- [14] BizAgi Process Modeler Available: <http://www.bizagi.com/eng/products/ba-modeler/modeler.html>, 2009.
- [15] Silver, B, “Can business analysts model Exception handling? BPMS Watch blogs”, Available: <http://www.brsilver.com/wordpress/2006/09/08/can-business-analysts-model-exception-handling/>, 2006.
- [16] “Ov-6c business process diagrams”. U.S. Department of Defense, Business Transformation Agency. Available: http://www.defenselink.mil/dbt/products/2008_EAETP/bea/iwp/bealist_ov-6cbusinessprocessdiagrams_na.html, 2008.

First A. Saravanan.

M.S received B.Sc degree in computer science from Madras University in 1996, the MCA degree from Bharathidasan University in 2001, the M.Phil degree from Madurai Kamaraj University in 2004, M.Tech degree from IASE University in 2005 and now pursuing PhD degree in Bharathiar University. His current research interest include modeling business processes, BPMN tools, exception handling. He is a Associate professor and Head in the Department of MCA in VRN College of Computer Science and Management, affiliated to S.V. University, Tirupati, Andhra Pradesh, India. He is totally having 11 years of teaching experience in various institutions in India.

Second B. Rama Sree.

R.J received M.S degree in computer science from BITS Pilani University in 1996 and PhD degree in S.P. Mahila University, Tirupati. She is a Reader in Department of Computer Science in Rashtriya Sanskrit University, Tirupati. Dr. Rama Sree has published three books and three international publications and ten national publications, having 17 years of teaching experience.

A new approach to: Obstacle-Avoiding Rectilinear Steiner Tree Construction

Animesh Pant^α
NIT Raipur

Abstract:-Given a set of pins and a set of obstacles on a plane, an obstacle-avoiding rectilinear Steiner tree(OARST) connects these points, possibly through some additional points(called Steiner points), and avoids running through any obstacle to construct a tree with a minimal total wire length. The OARST problem has received dramatically increasing attention recently. Nevertheless, considering obstacles significantly increases the problem complexity. Based on Obstacle-avoiding Spanning Graph (OASG), and edge based heuristic method has been applied to find the rectilinear Steiner tree with minimum wire length.

I. INTRODUCTION TO THE PROBLEM:

The problem of creation of Obstacle-Avoiding Rectilinear Steiner Trees can be stated by the following steps:

Step1: User defined inputs are taken from input files which contains set of nodes and set of obstacles.

Step2: Using a spanning graph generation algorithm an Obstacle Avoiding Rectilinear Spanning Graph (OARSG) is generated.

Step3: Prim's algorithm is applied to find the Minimum Spanning Tree (MST) from the OARSG.

Step4: Finally an Edge based heuristic is used on the minimum spanning tree to create the Obstacle Avoiding Rectilinear Steiner Tree.

II. ASSUMPTIONS:

1. Obstacles are of rectangular shape only.
2. No node can be inside the obstacle. Nodes can be either outside the obstacle boundaries or on the boundary of the obstacle.
3. Obstacles should not overlap each other.

III. ALGORITHMS

1) Constructing the Hanan Grid

A non-uniform two-dimensional routing grid is constructed by drawing horizontal and vertical lines on every node and the boundaries of the obstacles as shown in Fig. 1b. These lines are extended from terminals and corners of all obstacles in both horizontal and vertical directions until blocked by any obstacle or boundary of the design.



Fig. 1(a)

Fig. 1(a) A design instance with terminals and obstacles.

Fig. 1(b)

Fig. 1(b) A non-uniform routing grid (Hanan Grid)

2) Constructing the Obstacle-Avoiding Rectilinear Spanning Graph

A spanning graph is constructed considering the obstacles. We connect each pair of nodes in the input set through the Manhattan path taking one node as the source and the other as the destination and check if the Manhattan path is obstructed by any obstacle. If the path is obstructed by any obstacle then we find out the path to the nearest node of the opposite edge of the obstacle (opposite to the edge making the obstruction) and take that path and it is assigned as the source node. Now we will find the Manhattan path again for the changed source to the destination point. If this path is also obstructed by any obstacle then the same method is applied.



Fig (a) Manhattan paths being obstructed, Fig (b) The path being followed in case of an obstruction.

IV. PSEUDO CODE FOR GENERATION OF OBSTACLE-AVOIDING RECTILINEAR SPANNING GRAPH

INPUT: Co-ordinate points of set of nodes and obstacles

OUTPUT: Set of edges forming the Obstacle-Avoiding Rectilinear

Spanning Graph

For every pair of nodes:

Source=node1, destination=node2;

Spanning_graph(source,destination)

{ Find the manhattan paths between the source node and the destination node

For(each manhattan path)

{

if(edge is obstructed by an obstacle)

Algorithmic Approach for Creating and Exploiting Flexibility in Steiner Trees

Piyush Singh^α

Institute of Technology – BHU, Varanasi.

Animesh Pant^α

NIT Raipur.

Abstract-Routing is an important task in VLSI design and the rectilinear Steiner minimal tree (RSMT) construction is a fundamental research issue in the context of routing. Given a set of terminals, the RSMT problem is to find a rectilinear minimum spanning tree (RMST) that connects all the terminals, possibly through some additional points (called Steiner points) with minimal length. In practice, rectilinear Steiner trees are used to route signal nets by global and detail routers. Steiner tree problem is not just only routing problems in Computer networks it can also be used in designing proper road, airway routes. The concept of minimization of Steiner trees have practical applications in field of VLSI Routing, Wire length estimation, as all required minimization of intersections. Minimization of intersection can be achieved by creating and Exploiting flexibility in Steiner trees. But producing Flexibility in RST can produce such set with minimum number of intersections. The new, flexible tree is guaranteed to have the same total length. Any existing Steiner tree algorithm can be used for the initial construction of the Steiner tree. While solving for the flexibility in Steiner tree, problems like dealing with the overlaps have to be tackled and maximizing the flexibility has to done.

I. INTRODUCTION TO THE PROBLEM

The problem of creation and exploiting flexibility in Steiner Trees can be stated by the following steps:

Step1: User defined inputs are taken from input files which contains edge sets and Steiner nodes. The input files are generated from some pre-defined module for generation of Steiner tree.

Step2: Using unstable to stable rectilinear Steiner tree generation algorithm a stable Steiner tree is generated.

Step3: Algorithm to obtain parallel edge, flexible edge, and movable edge are applied to obtain the required.

Step4: The type of overlap is found, whether it is type 1 or type 2 overlap.

Step5: Finally Generate Steiner tree algorithm is used to create flexibility in Steiner tree.

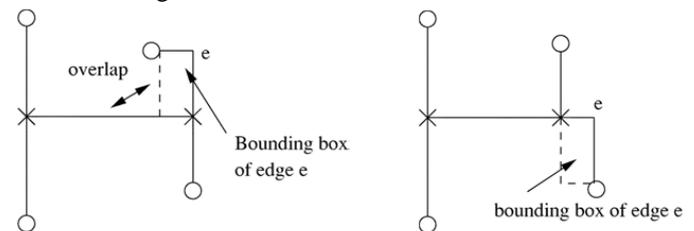
II. ALGORITHMS

1) Unstable Steiner tree to stable Steiner tree generation.

The major condition which is required for applying conditions of flexibility is that the rectilinear Steiner tree has to be stable. A RST is said to be stable under rerouting, if there is no pair of degenerate or non-degenerate L-shaped

segments, whose enclosing boxes intersect or overlap, except touching at a common end point (if any) of the two segments. No matter how we reroute a L-shaped segment, in a stable RST, within its enclosing box, no overlaps or crossings will occur, and the RST will essentially remain unchanged. A stable RST corresponds to a local minimum under the rerouting operation.

An RST is stable if there is no pair of edges such that their bounding boxes intersect or overlap except at a common endpoint (if any) of the two edges. Equivalently, a stable RST will not have overlaps when the edges are routed with minimum length.



Unstable RST

Stable RST

Pseudocode for generation of Stable Steiner tree:

Only when there are no degenerate segments (not edges)

Such that there bounding boxes overlap:

For (each segment (not edge))

{

//let the segment be n1n2

//where n1 and n2 denote the nodes of the segment considered

If (the segment is neither vertical nor horizontal)..... (1)

{

Declare two temporary nodes N1' and N2' such that

$N1'.x=n1.x, N1'.y=n2.y$

$N2'.x=n2.x, N2'.y=n1.y$

//make the set of all segments as A which are adjacent to segment under consideration

//let the segments belonging to the set be denoted by n3n4

// (the nodes n3 and n4 are variables)

If (((n3.x<n1.x<n4.x)&&(n3.y<n1.y<n4.y))||((similarly for n2))||((N1'))||((N2'))

{

//There exist overlapping

If (segment n3n4 is L-shaped)

{

Flip n3n4

}

```

If (segment n3n4 is not L-shaped and segment n1n2 is L shaped)
{
Flip n1n2
}
}
}
If (segment is either vertical or horizontal)
{
//The above process remains the same
//Only the number of comparisons reduces to 2
If (n3n4 is vertical or horizontal)
{
No overlapping
}
If (n3n4 is L-shaped)
{
Then perform the steps in (1), but the number of comparisons
would reduce to 2
}
}
}

```

2) *Algorithm for getting movable, parallel and flexible edge.*

Flexible edge: Flexible edges can be generated by moving the movable edges in RST.

Movable edge: These are special edges with following properties:

- Steiner-to-Steiner edge.
- Edge degree of each Steiner point is 3.
- Parallel edges exist at both ends.
- Flexible candidate exists at least at one end.

To get the movable edges we try to locate edge that is in between two Steiner points, if such edge is found than we try getting information whether it contains parallel edges, if it does and also contains either one or two flexible edge than that edge is considered to be a movable edge. To get a flexible edge we try to locate a movable edge, if movable edge is horizontal, we try to find the adjacent horizontal edge. If such an edge exist than that edge is considered to be a flexible edge.

To get a set of parallel edges we locate edges that are perpendicular to the movable edges and also pass through their pair of Steiner points.

Pseudocode for finding parallel, movable and flexible edges:

```

Checking for Flexibility and Movability
For (each segment)
{
CHECK FOR _MOVABILITY
{
If (the segment is between two Steiner points)
{
If (the Steiner edge is horizontal)
{
Check for s1;
{
If (there exist only one edge node n1 such that n1.y=s1.y)
{
One of the parallel edges is n1s1;
}
}
If (there exist two edge nodes n1 and n2 such that n1.y=n2.y=s2.y)
{

```

```

One of the parallel edge is n1n2 ;
}
}
If (there is no single edge node with any above two)
{
There is no parallel edge, hence continue; //go to 1
//if a parallel edge is found let it be S1
}
}
Check for s2;
{
Similarly check for single or double nodes for s2
//if a parallel edge through s2 is found let it be S2
//let the point of the parallel edge with lower y co-ordinate be
denoted by S.l
//and the one with higher y coordinate by S.h
}
If (parallel edge is found through both s1 and s2 )
{
If ((S1.l != S2.h) || (S1.h != S2.l))
{
The edge set is movable
So the movable set can be created;
}
}
}
If (the Steiner edge is vertical)
{
The process remains the same
But here the x co-ordinate has to be compared
And for verifying the parallel edges use the follow;
//let the point of the parallel edges with lower x coordinate be S.l
//the one with the higher x coordinate be S.h
}
}
CHECK FOR FLEXIBILITY
{
If both the Steiner points of the considered movable edge
Are not T-points simultaneously, then the considered movable edge
is
Movable and flexible
Hence the above edge and its adjacent edges can be entered into
The set of movable and flexible edges
}
}
3) Algorithm for generating flexible Steiner tree.
Once we have obtained a stable minimum rectilinear Steiner
tree, along with parallel, movable, and flexible edges than
we can apply Algorithm Generate flexible tree pseudo code,
to generate flexibility in Steiner tree. For exploiting the
Flexibility of Steiner tree we need flexible edges, movable
edges, and parallel edges.
Problem Formulation:

- Given a stable Rectilinear Steiner Tree
- Maximize the flexibility of the RST
- Subject to



à Topology remains unchanged (and thus if we do min-length edge connection, total length remains unchanged)



à No initial flexible edge is degraded in flexibility



Pseudocode for generating of flexible Steiner tree :



```

For Each edge e
{
If e and its adjacent edges are a movable set
{
Create Movable Set

```

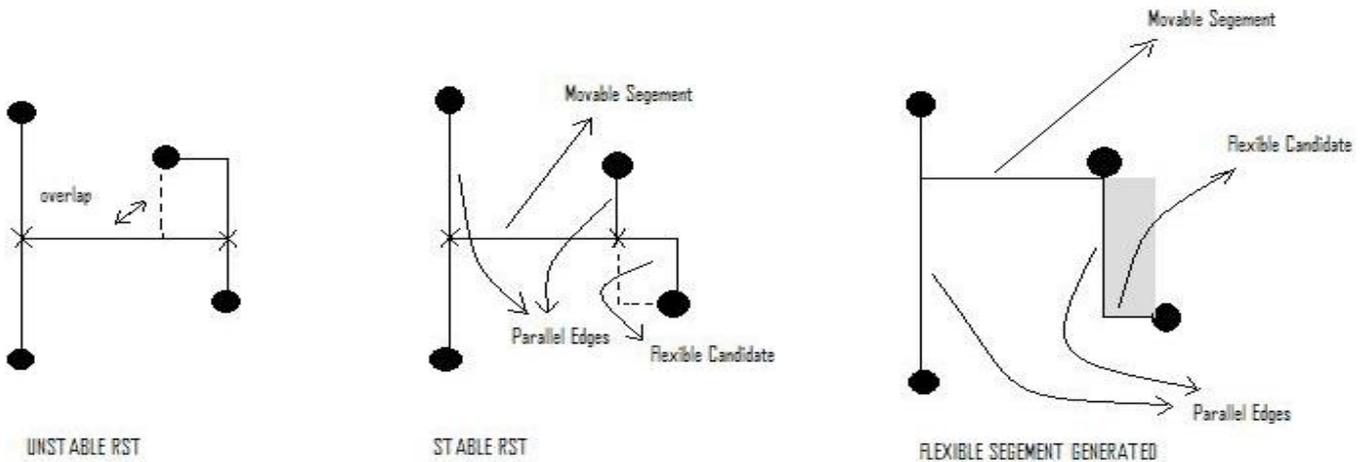

```

```

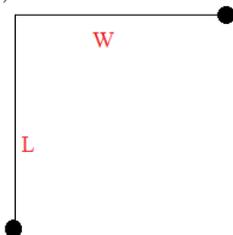
Check Overlap
}
For each movable set M
{
If M has no overlap
{
Move edge M
}
Move Overlapped edges
}
    
```

Steiner tree. The following figure shows example for generation of flexible Steiner tree when input is unstable Steiner tree.

Suppose we start from unstable Steiner tree, we apply algorithm to convert it into stable Steiner tree. Then we apply algorithm to get movable edge, flexible edge and parallel edge. One we are finished with these parts we apply Generate flexible tree algorithm to produce flexibility in



Flexibility function: Suppose we have a flexible edge as shown in the figure, w and l shows its dimensions than,



There are two general functions that can be used to compute the flexibility.

1. $f_1 = w + L$
2. $f_2 = w.L$

As we see in case 1 that $f_1 = w+L$ this shows the wire length, (f_1) will give us that change in the wire length. Similarly $f_2 = w.L$ gives us the area bounded by the rectangle formed by W and L. So (f_2) will give the change in area.

4) *Algorithm of finding out type of overlap that exists .*
The two types of overlap that exist are:

- **Overlap Type 1** It occurs when a parallel edge of two movable sets is the same and inequalities mentioned in intersection one hold.
- **Overlap Type 2** occurs when the flexible edge of a movable segment is a parallel edge of the other moving set.

Pseudocode for finding Type 1 overlap:

```

For each movable edges m1
{ //p1 and p2 be parallel edges
for each movable edge m2 !=m1
{
if (m1.p1==m2.p1 or m1.p2==m2.p2)
return "overlap Type1"
}
}
    
```

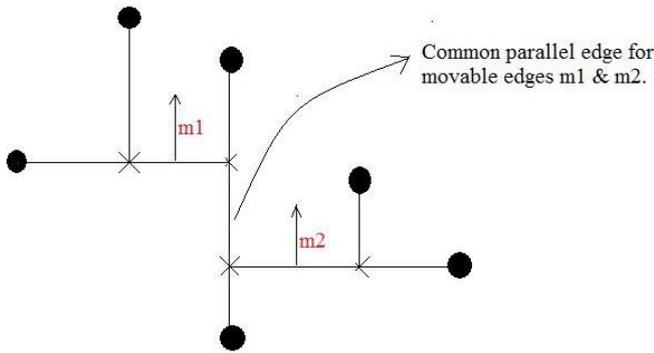


Fig: Showing Type 1 overlap

Pseudocode for finding Type2 overlap:

```

For each movable edges m1
{ //p1 and p2 be parallel edges; // f1 and f2 are flexible edges
for each movable edge m2 !=m1
{
if (m1.f1==m2.p1 or m1.f2=m2.p2)
return "overlap Type2"
}
}
    
```

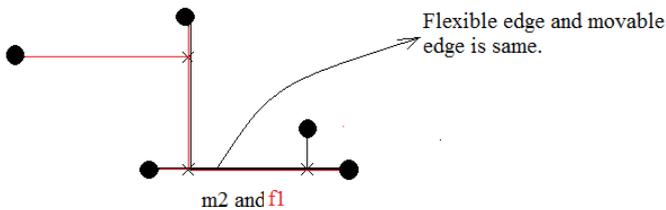


Fig: Flexible edge of one set is same as movable edge of other

5) *Algorithm of handling of overlap:*

Generating flexibility in a given stable RST is the final part of the algorithm. The need is to maximize the flexibility function. As there are only two types of overlaps that can exist in a RST, hence increment in flexibility is achieved only by solving the overlaps. When there is no overlapping, then it becomes a simple case of moving the movable edge to the maximum. In both types of overlap, we need to measure the flexibility in order to decide on moving the movable segments. The behavior of the overlap will depend on the flexibility function. If the movable sets overlap with each other, the maximum of flexibility function can not be obtained by maximizing the flexibility function for each movable set. Therefore, dealing with overlaps depends on the definition of flexibility function. Here we discuss mathematical formulation of flexibility function for overlaps of type I and II. If we take the flexibility functions as follows:

$$g(x,y)=X*Y$$

then the overall flexibility $G(x,y)$ is expressed in terms of flexibility function of each of the movable sets. Hence, the following expression gives the overall flexibility of a RST.

$$G(x,y)=g(x,y)$$

Here, we derive equations for solving the overlaps of type I and II.

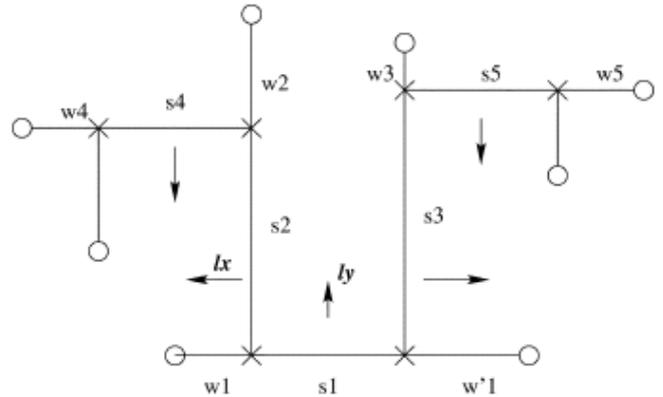
In the Fig. 5.1, we have a pair of movable edges exhibiting overlap of I kind. y_1 and y_2 are the distance moved by edges s_1 and s_2 respectively so as to maximize the flexibility.

$$G(x,y)=w_1y_1 + w_2y_2$$

Maximizing the above function results in values of y_1 and y_2 for which the $G(x,y)$ is maximum subject to certain constraints. In the Fig. 5, there are two movable edges in a single set. The two edges do not have the same direction of motion, so mathematical formulation becomes tedious as the number of movable edges increase above two in a given set. The flexibility of the RST in the Fig. 5.2 can be given by:

$$G(x,y)=(X_1+ X_2)y + Y_3x - xy$$

As the function is dependent on two variables, the solution for maxima yields the maximum flexibility. When the number of overlaps with a single movable edge becomes more than three, it is called a chain. As shown in the figure below, the formulation of the mathematical equation requires involvement of more than three variables, derivation of which is beyond the scope of this project.



III. EXPERIMENTAL RESULTS

We executed our C program in Dev C++ 4.9.9.2 on a PC-based machine with 3GHz Pentium processor and 2GB RAM under Windows Vista Operating system. The two inputs - edges of Steiner tree and Steiner points are taken from files edge.txt and spoint.txt. These two files contain input taken from some other module. For each set of input, we have taken two sets of output. The results are shown in the following table:

Table 1: Experimental Results

S.NO	Input Edges.	Input Steiner Nodes.	Type 1 Overlap.	Type 2 Overlap.	Time of run (Sec)
1.	10,10-20,10 20,10-30,10 20,10-20,30 30,10-30,30 30,10-50,10	20,10 30,10	0	0	0.07 sec
2.	10,10-20,10 20,10-30,10 20,10-20,30 30,10-30,30 30,10-50,10	20,10 30,10	0	0	0.066 sec
3.	60,50-60,60 60,60-60,80 60,60-80,60 80,60-80,70 80,70-76,70 80,60-90,60 60,50-60,60 60,60-60,80 60,60-80,60 80,60-80,70 80,70-76,70 80,60-90,60 90,40-90,60	60,60 80,60	0	0	0.088 sec
4.	60,50-60,60 60,60-60,80 60,60-80,60 80,60-80,70 80,70-76,70 80,60-90,60 90,40-90,60	60,60 80,60	0	0	0.086 sec
5.	10,50-60,50 60,50-60,160 60,160-60,190 60,160-110,160 110,160-110,101 10,160-110,230	60,160 110,160	0	0	0.07 sec
6.	10,50-60,50 60,50-60,160 60,160-60,190 60,160-110,160 110,160-110,101 10,160-110,230 10,150-40,150 40,150-40,110 40,150-110,150 110,150-110,200 110,150-110,80 110,80-110,50 110,80-210,80 210,80-210,130 210,80-250,80	60,160 110,160	0	0	0.069 sec
7.	10,50-60,50 60,50-60,160 60,160-60,190 60,160-110,160 110,160-110,101 10,160-110,230 10,150-40,150 40,150-40,110 40,150-110,150 110,150-110,200 110,150-110,80 110,80-110,50 110,80-210,80 210,80-210,130 210,80-250,80	40,150 110,150 110,80 210,80	Between 40,150 -110,150 & 110,80-210,80	0	0.107 sec

8.	10,150-40,150 40,150-40,110 40,150-110,150 110,150-110,200 110,150-110,80 110,80-110,50 110,80-210,80 210,80-210,130 210,80-250,80	40,150 110,150 110,80 210,80	Between 40,150 -110,150 & 110,80-210,80	0	0.109 sec
9.	0,0-1,0 1,0-3,0 3,0-4,0 3,0-3,1 1,0-1,1 1,1-2,1 1,1-1,2	1,1 1,0 3,0	0	Between 1,0-3,0 & 1,1-1,0	0.84 sec
10.	0,0-1,0 1,0-3,0 3,0-4,0 3,0-3,1 1,0-1,1 1,1-2,1 1,1-1,2	1,1 1,0 3,0	0	Between 1,0-3,0 & 1,1-1,0	0.99 sec
11.	1,0 - 2,0 2,0 - 2,2 2,2 - 2,0 2,2 - 2,3 2,0 - 3,0 3,0 - 3,1 3,0 - 4,0	2,2 2,0 3,0	0	Between 2,2-2,0 & 2,0-3,0	0.125 sec
12.	1,0 - 2,0 2,0 - 2,2 2,2 - 2,0 2,2 - 2,3 2,0 - 3,0 3,0 - 3,1 3,0 - 4,0	2,2 2,0 3,0	0	Between 2,2-2,0 & 2,0-3,0	0.098 sec
13.	0,2 - 1,2 1,2 - 1,1 1,2 - 3,2 3,2 - 3,3 3,2 - 3,0 2,0 - 3,0 3,0 - 4,0 4,0 - 4,1 4,0 - 5,0	1,2 3,2 3,0 4,0	Between 1,2-3,2 & 3,0-4,0	Between 1,2-3,2 & 3,2-3,0 3,2-3,0& 3,0-4,0	0.164 sec

14.	0,2 – 1,2	1,2	Between	Between	
	1,2 – 1,1	3,2	1,2-3,2 &	1,2-3,2 &	
	1,2 – 3,2	3,0	3,0-4,0	3,2-3-0	
	3,2 – 3,3	4,0			
	3,2 – 3,0			3,2-3,0&	0.164 sec
	2,0 – 3,0			3,0-4,0	
	3,0 – 4,0				
	4,0 – 4,1				
	4,0 – 5,0				

IV. ADVANTAGES OF PROJECT

1. This project can be used to make a Steiner tree flexible, so that it can reduce number of intersections if there are more than one Steiner trees are used in a grid.
2. If number of intersections can be reduced by using concept of flexibility in Steiner tree, than number of layers may also get decreased.
3. As the effective topology remain same and number of intersections decreases leading to decrease in the layering so the effective cost of VLSI chip may also decrease.

V. CRITICISM

If more than two overlapping exist in the project than there might be a change that software may not work.

VI. CONCLUSIONS AND FUTURE WORK:

This implementation is very simple to understand and easy to use, since it uses conventional data structures like struct, arrays, procedural functions and procedural code. Experimental results show our method can work well for the

defined problem. In this project, finding a solution to the problems was more important than the running time. We believe that the implementation and methods in our work can help the design of routing tools in the future. The code can be used to reduce number of intersections in multilayer environment.

The running time of the implemented algorithm is $O(n^3)$, in the worst case, and further improvement on the running time is possible by applying better programming techniques.

VII. REFERENCES

1. Elaheh Bozorgzadeh, Ryan Kastner, and Majid Sarrafzadeh : "Creating and Exploiting Flexibility in Rectilinear Steiner Trees".
2. Jan-ming Ho, Gopalakrishnan Vijayan and C. K. Wong: "A New Approach to the Rectilinear Steiner Tree Problem".
3. Manjit Borah, Robert Michael Owens, and Mary Jane Irwin: "An Edge-Based Heuristic for Steiner Routing".

Initial Hybrid Method for Software Effort Estimation, Benchmarking and Risk Assessment Using Design of Software

J. FRANK VIJAY

Department of CSE, SRM Valliammai Engineering College, SRM nagar, Kattankulathur, Chennai, Tamil Nadu – 603 203. India.
jeirus.jeff@gmail.com

PROF. DR. C. MANOHARAN,

Director/Principal, VSA Group of Educational Institutions, School of Engineering & School of Management,, LNH – 47Main Road, Uthamasolapuram (PO), Salem, Tamil Nadu -636 010, India.
c_m_66@yahoo.co.in

Abstract- Estimation models in software engineering are used to predict some important attributes of future entities such as development effort, software reliability and programmers productivity. Among these models, those estimating software effort have motivated considerable research in recent years [COSMIC, (2000)]. In this paper we have discussed an available work on the effort estimation methods and also proposed a hybrid method for effort estimation process [Briand et.al, (1998)]. As an initial approach to hybrid technology, we have developed a simple approach to SEE based on Use Case Models: The “Use Case Points Method.” [Briand et.al, (1998)]. This method is not new, but has not become popular although it is easy to understand and implement. We have therefore investigated this promising method, which is inspired by Function Points Analysis [Albrecht, (1994)]. Reliable estimates can be calculated by using our method in a short time with the aid of a spreadsheet but we are planning to extend its applicability to estimate risk and benchmarking measures [Briand et.al, (1998)][Sentas et.al, (2005)].

Keywords: Effort Estimation; Cost Refinement; Function Points; Use Case Points; Risk Assessment; Hybrid Method; Benchmarking.

I. INTRODUCTION

The planning, monitoring and control of software development projects require that effort and costs be adequately estimated. However, some forty years after the term “software engineering” was coined [Jorgenson and Shepperd,(2007)], effort estimation still remains a challenge for practitioners and researchers alike. There is a large body of literature on software effort estimation models and techniques in which a discussion on the relationship between software size and effort as a primary predictor has been included [Albrecht,(1994)] [Albrecht and Gaffney,(1983)] [Abts and Chulani,(2000)] [Boehm,(1981)] [Anda et.al,(2001)] [Arnold and Pedross,(1998)] [Basili and Freburger ,(1998)]. They conclude that the models, which are being used by different groups and in different domains, have still not gained universal acceptance [Guruschke and

Jorgensen ,(2005)]. As the role of software in the society becomes larger and more important, it becomes necessary to develop a package which is used to estimate effort within a short period. In order to achieve this goal, the entire software development processes should be managed by an effective model. So, our proposed model will be focusing on three basic parameters. 1. Software effort estimation 2. Benchmarking 3. Risk Assessment. So far, several models and techniques have been proposed and developed [Boehm and Royce,(1992)] [Anda et.al,(2001)] [Symons,(1991)] and most of them include “Software Size” as an important parameter. The below graph shows the application of software engineering principles and standards in medium sized organizations.

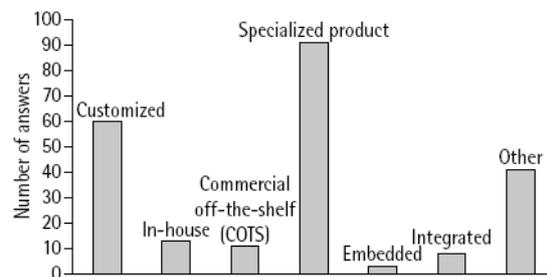


Fig: Reference: The Application of software engineering standards in very small enterprises, Vol3, issue 4

The Use Case Model can be used to predict the size of the future software system at an early development stage to estimate the effort in the early phase of software development;

Use case point method has been proposed [Smith,(1991)]. Use Case Point Method is influenced by the Function Points Methods and is based on analogous Use Case Point [Smith, (1991)].

We have been involved in the activity of developing a hybrid model to estimate the effort in the early phase of software engineering development [Briand et.al,(1998)]. This paper describes the method of introducing Use Case Points method to software projects for estimating effort. The paper also describes the automatic classification of actors and use cases in the UCP model rather than doing it manually. The result of this paper will be taken as a base for developing a hybrid method which will be used for benchmarking and risk assessment [Sentas et.al,(2005)].

II. PROBLEM FRAMEWORK

Our understanding of the effort-estimation problem arises from the idea that any software project is the result of a set of business goals that emerge from a desire to exploit a niche in the marketplace with a new software product. Take, for example, the development of an application server that caters to on-demand software. The business goals of having a robust, high-performance, secure server lead to a set of architectural decisions whose goal is to realize specific quality-attribute requirements of the system (e.g., using trimodular redundancy to satisfy the availability requirements, a dynamic load-balancing mechanism to meet the performance requirements, and a 256-bit encryption scheme to satisfy the security requirements). Each architecture A that results from a set $\{A_i\}$ of architectural decisions has a different set of costs $C\{A_i\}$ (Fig. 2). The choice of a particular set of architectural decisions maps to system qualities that can be described in terms of a particular set of stimulus/response characteristics of the system $\{Q_i\}$, i.e., $A_i \rightarrow Q_i$. (For example, the choice of using concurrent pipelines for servicing requests in this system leads to a predicted worst-case latency of 500 ms, given a specific rate of server requests.) The “value” of any particular stimulus/response characteristic chosen is the revenue that could be earned by the product in the marketplace owing to that characteristic. We believe that the software architect should attempt to maximize the difference between the value generated by the product and its cost.

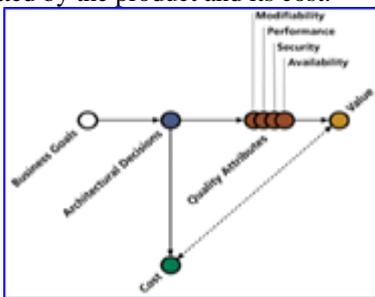


Fig: Business goals drive the architectural decisions $\{A_i\}$, which determine the quality attributes $\{Q_i\}$. Value (V_a) depends on Q_i and Cost (C) depends on A_i .

III. RELATED WORK

Until today, several researches [Boehm et.al,(2001)] [Boehm et.al,(1995)] and case studies have been reported about the Use Case point and effort estimation based on Use Case Model [COSMIC,(2000)]. Smith proposed a method to estimate Line of Code from Use Case Diagram [Smith,(1999)] [Aggarwal et.al,(2005)]. Arnold and Pedross reported the Use Case Method can be used to estimate the size of the software [Arnold and Pedross,(1998)]. They also suggested that Use Case Point Method should be used with other estimation method to get the optimum result.

IV. LIMITATIONS OF FUNCTION POINTS

Function Point is a measure of software size that logically measures the functional terms and the measured size stays constant irrespective of the programming language and environments used [IFPUG,(2002)]. In Function Point, it is very much essential to use the detailed information about the software. Such detailed information will be available in software design specification. Function Point metric evaluation is difficult to estimate for software which has short development time [Hajri et.al,(2005)]. So, in reality estimation of software at the earlier phase of the development life cycle process will certainly reduces risk. To estimate the effort accurately in the earlier phase of the development life cycle process, Use Case Point Method has been proposed [Smith ,(1999)].

V. USE CASE POINT METHOD

This section briefly explains the procedure how Use case point has been implemented in our model [Smith,(1999)].

A. Use case point method

The first and the foremost step is to calculate Use Case Point (UCP) from Use Case Model [Smith,(1999)]. The Use Case Model mainly consists of two documents, system or sub system documents & use case documents contains the following description of items: system name, risk factors, system – level use case diagram,, architecture diagram, subsystem descriptions, use case name, brief description, context diagram, preconditions, flow of events, post conditions, subordinate use case diagrams, subordinate use cases, activity diagram, view of participating classes, sequence diagrams, user interface, business rules, special requirements & other artifacts [Schneider and Winters,(2001)].

From the above specified information we are going to focus mainly on two parameters system – level use case diagram and flow of events. System – level use case diagram includes one or more use case diagrams showing all the use cases and actors in the system [Schneider and Winters,(2001)]. Flow of events includes a section for the normal path and each alternative path in each use case.

Figure 2 shows a part of flow of events of the use case “SESSION” in Figure 1.

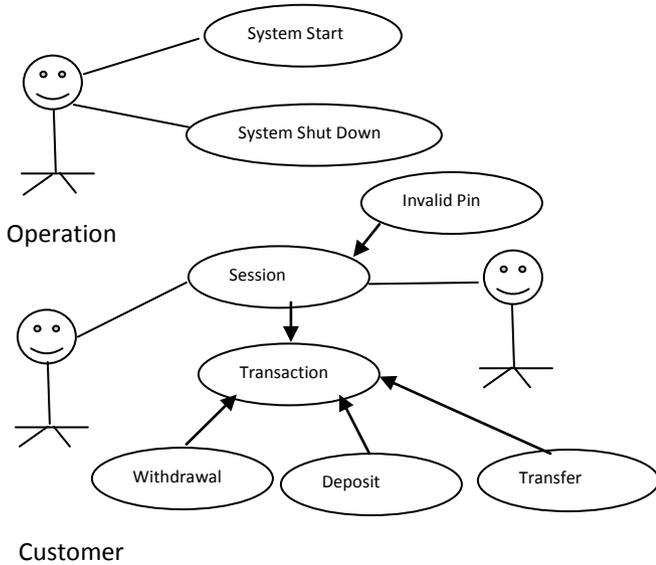


Fig: An example of System Level use case diagram for ATM System

A session is started when a customer inserts an ATM card into the card reader slot of the machine.

The ATM pulls the card into the machine and reads it.

If the reader cannot read the card due to improper insertion or damaged stripe, the card is ejected, an error screen is displayed, and the screen is aborted.

The customer is asked to enter his/her PIN, and is then allowed to perform one or more transactions, choosing from a menu of possible types of transaction in each case.

Fig: Flow of Events (Session Use Case)

B. Counting use case point:

Intuitively, UCP is measured by counting the number of actors and transactions included in the flow of events with some weight. A transaction is an event that occurs between an actor and the target system, the event being performed entirely or not at all. But, in our method the effort estimation is calculated by applying the following procedure

a) Counting actor’s weight

The actors in the use case are categorized as simple, average or complex. A simple actor represents another system with a defined API. An average actor is either another system that interacts through a protocol such as TCP/IP or it is a person interacting through a text based interface. A complex actor is a person interacting through a GUI interface.

Type	Description	Factor
Simple	Program Interface	1

Average	Interactive, Protocol Driver	or	2
Complex	Graphical Interface	User	3

Table – 1

The number of each actor type that the target software includes is calculated and then each number is multiplied by a weighting factor shown in TABLE – 1. Finally, actor’s weight is calculated by adding those values together.

b) Counting use case weights

Each Use case should be categorized into simple, average or complex based on the number of transactions including the alternative paths. A simple use case has 3 or fewer transactions, an average use case has 4 to 7 transactions and a complex use case has more than 7 transactions.

Then, the number of each use case type is counted in the target software and then each number is multiplied by a weighting factor shown in Table – 2.

Type	Description	Factor
Simple	3 or fewer transactions	5
Average	4 to 7 transactions	10
Complex	More than 7 transactions	15

Table- 2. Transaction Based Weighting Factors

Finally, use case weight is calculated by adding these values together.

c) Calculating unadjusted use case points

It is calculated by adding the total weight for actors to the total for use cases.

Factor	Description	Weight
T ₁	Distributed System	3
T ₂	Response or Throughput Performance Objectives	4
T ₃	End – User Efficiency (online)	5
T ₄	Complex Internal Processing	2
T ₅	Code must be readable	3
T ₆	Easy to install	5
T ₇	Easy to use	5

T ₈	Portable	2
T ₉	Easy to Change	5
T ₁₀	Concurrent	1
T ₁₁	Includes special security features	4
T ₁₂	Provides direct access for third parties	2
T ₁₃	User training facilities required	2

Table – 3

d) Weighting technical and environmental factors

The UUCP are adjusted based on the values assigned to a number of technical and environmental factors shown in Tables 3 & 4.

Factor	Description	Weight
F ₁	Familiar with the Rational Unified Process	4
F ₂	Application Experience	3
F ₃	Object – Oriented Experience	2
F ₄	Lead Analyst Capability	3
F ₅	Motivation	5
F ₆	Stable Requirements	4
F ₇	Part – Time Workers	3
F ₈	Difficult Programming Language	3

Table – 4

Method:

Each factor is assigned a value between 0 and 5 depending on its assumed influence on the project. A rating of 0 means the factor is irrelevant for this project and 5 means it is essential.

Calculation of TCF:

It is calculated by multiplying the value of each factor (T₁ – T₁₃) in Table 3 by its weight and then adding all these

numbers to get the sum called the T Factor. Finally, the following formula is applied:

$$TCF = 0.6 + (0.01 * T \text{ Factor})$$

Calculation of environmental factor:

It is calculated accordingly by multiplying the value of each factor (F₁ – F₈) in TABLE – 4 by its weight and adding all the products to get the sum called the E Factor. Finally, the following formula is applied:

$$EF = 1.4 * (-0.03 * E \text{ Factor})$$

Calculating UCP

Use Case Point (Adjusted) is calculated by

$$UCP = UUCP * TCF * EF$$

(3)

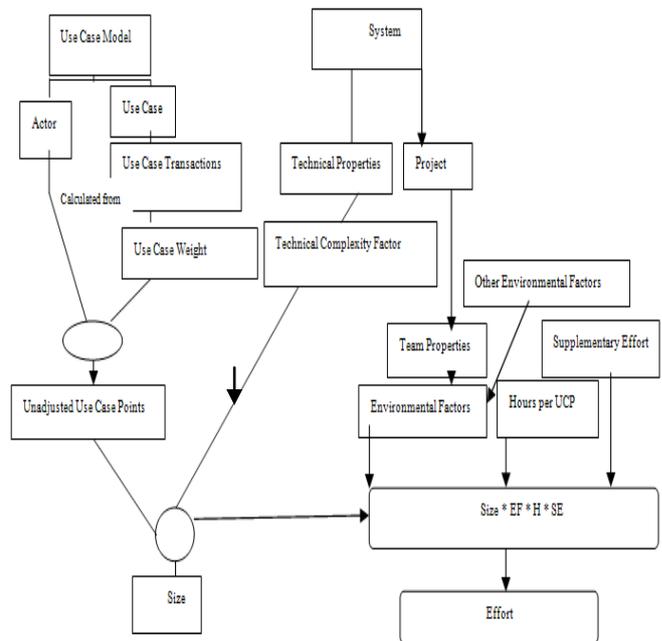


Figure 3: Calculating Use Case Point

Estimating effort:

By multiplying the specific value (man – hours) by the UCP, the effort can be easily calculated. In [Smith,(1999)], a

factor of 20 man – hours per UCP for a project is suggested. The entire procedure is diagrammatically shown above.

Research Method

Based on the proposed method, we have planned to develop a framework [Alistair,(2000)] as an automated tool under the name [Hybrid Tool]. The input is a XMI File. The tool is implemented in JAVA and Xerces 2 Java Parser is used to analyze the model file [OMG,(2005)].

VI. AN AUTOMATED TOOL FOR ESTIMATING USE CASE POINT

(1) Overview

In order to effectively introduce Use Case Point Method to the software development, we have decided to create a Use Case Point measurement tool [Smith, (1999)]. There were several existing tools available which is based on Use Case Model but in all these existing models, it is necessary to judge the complexity of actors and Use cases by manually. The judgment is the most important part in software cost estimation so we have decided to create an automated tool. So, in order to obtain the entire procedure described in section 5 automatically, it is mandatory to describe a set of rules to classify the weight for actor and use case in section 5.2.

Also, it is necessary to write the Use – Case Model in machine – readable format. So, we assume that the use case model is written in XMI [XML Metadata Interchange] [OMG,(2005)]. The reason for choosing this type of file format is because most case tools for writing UML diagrams support to export them as XMI files [OMG,(2005)].

(2) Rules for weighting actors

As described in section 5.2, weight for each action is determined by the interface between actor and the target software. But, the interface information will not be available in the actor description. Only the name of the actor will be available. So, it is very much essential to create a protocol which determines the complexity of actor.

Classification based on actor's name

At the initial stage of the classification we are going to determine whether the actor is a person or an external system based on the name of the actor. That is, beforehand, we prepare the list of keywords which can be included in the name of the software system. FOR EXAMPLE the keywords “system” and “server” are used in the system's name.

Keywords for step 1 (KL_a) : System, Server, Application, Tool.

We are planning to initially start the automated tool with a minimal set of keywords. As on later stages, the new keywords will be updated automatically and can be used for later projects.

Classification based on keywords included in use case

Here, we are going to classify based upon on the flow of events to which the actor is relevant. As an initial stage, we are planning to develop a three set of keywords to each complexity factor of actor and then, we will try to extract all words included in the flow of events and then match them with each keyword in the lists. Finally, the actor's weight is assigned as the complexity for the keyword list that is most fitted to the words in the flow of events.

Keywords for simple actor (KL_{sa}) : Request, Send, Inform.

Keywords for Complex actor (KL_{ca}) : Press, Push, Select, Show, GUI, Window

Keywords for Average Actor (Person) (KL_{aap}) : Command,Text, I/P, CUI

Classification based on experience data:

Suppose, if we are unable to determine the actor's weight at step2, we determine it based on the experience data. The experience data includes the information about the Use Case Model and the Use case Point developed in the past software projects.

(3) Rules For Weighting Use Cases

As described in section 5.2, the complexity of use case is determined by the number of transactions. So, we have decided to focus on the flow of events in the Use Case Model. The simplest way to count the transaction is to count the number of events. There are no standard procedures or protocols to write the flow of events and it is also quite possible that several transactions are described in one event. So, because of this limitation several guidelines to write events in use case model have been proposed [Schneider and Winters,(2001)]. There are ten guidelines to write a successful scenario. Among them, we focus on the following two guidelines.

(G₁) → Use a Simple Grammar

(G₂) → Include a reasonable set of actions.

Jacobson suggests the following four pieces of compound interactions should be described. (4)

The primary actor sends request and data to the system,

The system validates the request and the data,
 The system alters its internal state and
 The system responds to the actor with the result.
 So, based on the above said guidelines, we propose the way to analyze the events using the morphological analysis and syntactic analysis. Through these analyses, we can get the information of morpheme from the statement and dependency relation between words in the statement. We conduct the morphological analysis for all statements and get the information of the subject word and predicate word for each statement.

Then, we apply the following rules:

Rule U – 1:

We regard each set of the subject and predicate word as a candidate of a transaction

Rule U – 2:

Among the candidates, we identify the one that related to actor's operation and system response as a transaction

For each use case, we have to apply the above said rules and based on these rules, we get the number of transactions. Then, based on the number of transactions we determine the complexity of each use – case.

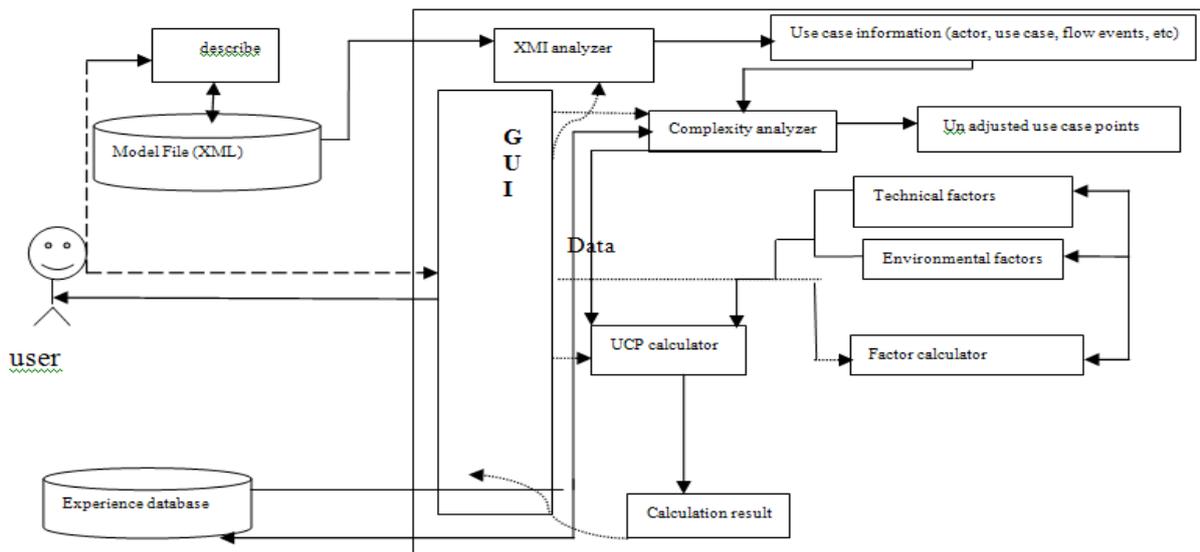


Figure 4: Automated Tool

VII. CONCLUSION & FUTURE WORK

This paper has proposed an automated Hybrid tool which calculates Use Case Points from Use Case Models in XMI files [OMG,(2005)]. We will use the effort estimation based on this Hybrid Tool in the hybrid technology proposed for Risk assessment and benchmarking. We will also extend this technique for developing an automated tool for assessing risk and effort.

REFERENCES

- Abts. C. ; Chulani . A. W. (2000): *Software Cost Estimation with COCOMO II*, Prentice Hall, New Jersey.
- Aggarwal.K.K., ; ChandraP., ; Singh. Y, ; Puri.M. (2005) “An Expert Committee Model to Estimate Lines of Code”ACM ,vol 30,pp 1-4.
- Albrecht. A. J. (1994) “Function Point Analysis”, Encyclopedia of Software Engineering, Vol. 1, pp. 518 – 524.
- Albrecht. A. J.; Gaffney. J. E. (1983) "Software function, source lines of codes, and development effort prediction: a software science validation," IEEE Trans Software Eng. Vol.SE-9, pp.639-648.
- Al-Hajri.M.A.,; Ghani. A.A.A.,; SulaimanM.S.,; Selamat.M.H.,(2005)” Modification of standard function point complexity weights system,” Journal of Systems and Software vol.74 ,195–206.
- listair. C. (2000) Writing effective use cases (Agile Software Development Series), Addison – Wesley.
- Anda. B, ; Dreiem. H,; Sjoberg. D.I.K,; Jorgensen. M(2001) : “Estimating Software Development

- Effort based on Use Cases – Experiences From Industry”, Proceedings of Fourth International Conference on the UML, pp. 487 – 504.
- A mold.M. ; Pedross.P (1998) : “Software Size Measurement and Productivity rating in a Large Scale Software Development Department”, Proceedings of the 20th ICSE, pp. 490 – 493.
- Asundi. J. (2005) “ The Need for Effort Estimation Models for Open Source,”,ACMvol30,pp1-3.
- Basili.V.R.,; Freburger.K. (1998) : “ Programming Measurement and estimation in the Software Engineering Laboratory”, Journal of Systems & Software, 2, pp. 490 – 493.
- B Boehm, B.W.(1981) *Software Engineering Economics*, Prentice-Hall.
- B Boehm, B.; Chen. Y. (2004) An Empirical Study of eServices Product UML Sizing Metrics. The 2004 International Symposium on Empirical Software Engineering (ISESE), Redondo Beach, California, pp. 199-206.
- Boehm. B.; Royce. W (1992) “Ada COCOMO and the Ada Process Model”, s.l, DTIC.
- Boehm.B. ; Clark. B ; Horowitz.E ; Westland,C (1995) ”Cost models for future software life cycle processes: COCOMO 2.0”, Springer Netherlands, Annals of Software Engineering, vol-1, pp. 57-94.
- Common Software Measurement International Consortium,S. COSMIC – FFP Version 2.0 (2000). [http:// www. Cosmicon.com/](http://www.Cosmicon.com/)
- Dekkers.T. (2007),”Benchmarking is an essential control mechanism for management,”, RPM-AEMES, vol. 4,pp 99-103.
- International Function Point Users Group [IFPUG], “Function Point Counting Practices Manual, Release 4.1.1”, (2002).
- Jeffery.R ; Ruhe.M.,; Wiczorek. M, (2000).: A Comparative Study of Two Software Development Cost Modeling Techniques using Multi-organizational and Company-specific Data. Information and Software Technology, vol. 42,1009-1016.
- Jørgensen, M.; Molokken-Ostfold, K. (2002): *Reasons for Software Effort Estimation Error*
- Jørgensen, M; Gruschke, T.M. (2005) Industrial use of Formal Software Cost Estimation Models: Expert Estimation in Disguise? Proc. of Em Assessment in Software Engineering (EASE), Keele, United Kingdom, April 11-13.
- Jorgensen,M.; Shepperd,(2007)”A Systematic Review of Software Development Cost Estimation Studies,”, Software Engineering, IEEE Transactions on, vol. 33. pp 33-53.
- Kitchenham, B.; Mendes, E.; (2004) Software Productivity Measurement Using Multiple Size Measures. IEEE Transactions on Software Engineering, Vol. 30, No. 12, 1023-1035.
- Lionel C. Briand,; Khaled El Emam,; Frank Bomarius (1998)“COBRA: A Hybrid Method fpr Software Cost Estimation, Benchmarking and Risk Assessment”, Proceedings of the 20th International Conference on Software Engineering, IEEE CS.
- Mittas.N.; Angelis.L. (2008), “Comparing Cost Prediction Models by Resampling Techniques”, *Journal of Systems and Software*, Vol. 81, Iss. 5.
- Naur P., Randell B. (Eds.), *Software Engineering*, Conference Report, NATO Science Committee, Garmisch (Germany), 7-11 October 1968
- Object Management Group (OMG), “XML Metadata Interchange (XMI) Specification Version 2.0”, 2005.
- Schneider.G. ; Winters. J. P.(2004) : “Applying Use Cases, Second Edition”, Addison Wesley (2001).Impact of Respondent Role, Information Collection Approach, and Data Analysis Method. IEEE Transactions on Software Engineering, Vol. 30, No. 12), 993- 1007.
- E. Pentas,; Angelis.L.; Stamelos. I. ; Bleris.G (2005) “Software Productivity and Effort Prediction with Ordinal Regression”, *Information and Software Technology* Trans. 47, pp. 17-29.
- Smith J(1999) : “The estimation of effort based on Use Cases”, Rational Software White Paper.
- C (1991) : *Software Sizing & Estimating*, John Wiley & Sons.
- alston. C.E. ; Felix. C. P.(1977) “A Method of program measurement and estimation”, IBM Systems Journal, 16(1), 54 – 73.
- Wei Xiaa, Luiz Fernando Capretz,” A new calibration for Function Point complexity weights,” Journal of systems and software. , Vol 50, pp.670 – 683, 200

Diffie-Hellman Key Exchange: Extended to Multi-Party key Generation for Dynamic Groups

B.SRINIVASA RAO
buragasrinivasarao@gmail.com
PVPSIT, Vijayawada, India

SWAPNA.D
swapna.donepudi@gmail.com

Abstract- As a result of increased popularity of group-oriented applications and protocols, group communication occurs in many different settings. This paper considers the two-party-Diffie-Hellman (DH) key exchanging technique is extended to generate an efficient contributory multi-party key exchanging technique for a dynamic group. In this technique, a member who acts as a group controller forms two-party groups with other group members and generates a DH-style shared key per group. It then combines these keys into a single multi-party key and acts as a normal group member. The technique has been compared with other techniques and satisfactory results are obtained.

Keywords

DH-Diffie-Hellman, KEO-Key Exchange Overhead

I. INTRODUCTION

Because of the rapid increase of popularity of remote communication through unsecured channels such as Internet, the use of cryptographic protocols to secure them increases. The reliability of protocol depends on keys being used, because the messages are easily interpreted when opponents know the secret values. So, we need an efficient cryptographic protocol.

This paper proposes an extension of basic DH to generate a multi-party key for a dynamic group. Its main advantage is to implement various groupware applications such as conference calls, distributed computations, distributed databases and so on in a secured way. Since key distribution is main factor of the secure group communication, it received a lot of attention.

The concept of multi-party key generation is that the two-party DH key exchange algorithm has been extended to work with three or more parties. Basically all group-key generating techniques can be divided into two classes. In one class, a single participant generates the key and distributes it to all parties. It is efficient if it is implemented correctly. However it requires a trusted key generator. A group key in other class is a contributory key, which is generated by exchange of private values of individual group member and by some additional Key Exchange Operations (KEO). The group key generating technique proposed in this paper belongs to this class.

In this paper by considering the multi-party key generation technique we focus on initial key agreement, member addition, member deletion, mass add, split, merge and key refresh operations.

The paper is organized as follows. The basics of original two-party DH algorithm. In the next section Multi-Party Key Generation, member addition, member deletion, mass join, split, merge and key refresh operations are discussed.

II. PRELIMINARIES OF TWO-PARTY DH TECHNIQUE

Diffie and Hellman introduced the concept of two-party key-exchanging technique [3] that allows two participants to exchange two public keys through unsecured channel and generate a shared secured key between them. Each party then combines the others public key along with its own private key to generate a shared key. The steps as follows.

Both Alice and Bob agree on two large positive integers, n and g such that n is a prime number and g is a group generator

Alice randomly chooses a positive integer, x , which is smaller than n and serves as Alice private key. Similarly Bob chooses his private key, y .

Both Alice and Bob compute their public keys using $X=g^x \bmod n$ and $Y=g^y \bmod n$, respectively.

They exchange their public keys through a public communication channel

On receiving, both Alice and Bob compute their shared key K , using $K=Y^x \bmod n = g^{xy} \bmod n$ and $K=X^y \bmod n = g^{xy} \bmod n$.

III. GENERATION OF MULTI-PARTY KEY

We proposed a contributory group key agreement protocol to exchange a multi-party key among group members. In Group-DH an arbitrary group member acts as a group-controller and forms a two-party group with the remaining group members. Each group individually generates a DH-style key using DH technique.

The group controller generates $(n-1)$ public keys by raising the exponent of g with product of $(n-2)$ shared keys at a time and sends to the corresponding group member's. On receiving each group member raises the exponent with its own shared key and generates the group key.

Let the group controller be P_i where $1 \leq i \leq n$ for n-party group. Initially it itself forms a two-party group with each of the remaining group members, and produces (n-1) two party groups. The P_i then generates a DH style key per group, and produces (n-1) shared keys for (n-1) two party groups. In order to accomplish it, P_i generates a public key and broadcasts to the remaining group members.

The public key $X_i = g^{x_i} \bmod n$ where x_i is the private key of P_i

Each group member, P_j where $j \neq i$ also assumes a private key and generates a public key as

$X_j = g^{x_j} \bmod n$ where X_j is private key of P_j and $1 \leq j \leq n$, $j \neq i$.

Each P_j then transmits X_j to the group controller P_i . After exchanging the public keys, each member similar to the basic DH generates a unique shared key, $K_j = (X_i)^{x_j} \bmod n = g^{x_i x_j} \bmod n$. It actually generates (n-1) shared keys for (n-1) groups. The group controller combines these shared keys to produce a single group key. The P_i computes the public key X_k as given below and sends to P_j

$$X_k = g^{\prod_{k \neq j} x_k} \bmod n$$

Where $1 \leq k \leq n$, $k \neq j$

Each party in the group then generates the group key k as follows

P_1 generates $K = (X_k)^{K_1} \bmod n = g^{k_1 K_2 \dots k_{n-1} K_n} \bmod n$

P_2 generates $K = (X_k)^{K_2} \bmod n = g^{k_1 K_2 \dots k_{n-1} K_n} \bmod n$

P_3 generates $K = (X_k)^{K_3} \bmod n = g^{k_1 K_2 \dots k_{n-1} K_n} \bmod n$

 P_{n-1} generates $K = (X_k)^{K_{n-1}} \bmod n = g^{k_1 K_2 \dots k_{n-1} K_n} \bmod n$

P_n generates $K = (X_k)^{K_n} \bmod n = g^{k_1 K_2 \dots k_{n-1} K_n} \bmod n$

Since the group controller knows all the two-party shared keys, it also generates the group key using

$$K = g^{k_1 K_2 \dots k_{n-1} K_n} \bmod n$$

IV. MEMBER ADDITION

The member addition operation occurs when a new joining member sends a join request to the group. The procedure is as follows

Suppose that group has n numbers $\{M_1 \dots M_n\}$. The new member M_{n+1} broadcasts a message to the group controller by sending its public key which is computed by assuming a private key.

Suppose P_i is group controller new member M_{n+1} sends his public key X_j where $X_j = g^{x_j} \bmod n$ where x_j is private key of M_{n+1} . After receiving the public key from M_{n+1} the group controller P_i sends his public key to M_{n+1} and both of them compute the shared key

M_{n+1} generates a shared key

$K_j = (X_i)^{x_j} \bmod n = g^{x_i x_j} \bmod n$ which is same as P_i shared key.

V. MEMBER DELETION

The delete operation occurs when a group member sends a leave request to the group. The group controller announces the leave event to remaining group members

On receiving this as he is leaving there are chances that he may leak the group key, so the group performs the multi-key generation technique and generates a new group key.

VI. MASS JOIN

The mass join operation occurs when two or more people send joining requests to group controller. The group controller then announces mass join event to current group and joining members

The current group and the new members receive notification and perform the multi-party key generation technique and generate a new group key.

VII. SPLIT

The split operation occurs when current group will split into two or more subgroups. The group controller announces split event to every member and each subgroup performs the multi-party key generation to generate their own group key.

VIII. MERGE

Merge operation occurs when two or more subgroups wish to merge into one group. The group controller announces merge event for subgroups. After forming a group it performs the multi-party key generation to form group key.

IX. KEY REFRESH

To avoid exposure we should refresh the group key periodically.

X. SECURITY ASPECT

The group key derived in the application is indistinguishable in polynomial time from random numbers.

XI. COMPARISON OF PROPOSED TECHNIQUE WITH OTHER TECHNIQUES

Protocols	Rounds	Messages	Unicast	Broadcast	Message size	Sequential Exponentiations
CCEGK	H	$2n-2$	n	$n-2$	$2n-2$	$2h-2$
EGK	H	$2n-2$	0	$2n-2$	$2n-2$	$2h-2$
TGDH	STR	$2n-2$	0	$2n-2$	$2n-2$	$2h-2$
STR	$n-1$	$2n-2$	0	$2n-2$	$2n-2$	$2(n-1)$
GDH 3.0	$n+1$	$2n-1$	$2n-3$	2	$3n-6$	$5n-6$
GROUP-DH	$n+1$	$n+1$	$n-1$	2	$2n-1$	$2n$

XII. CONCLUSION

In this paper we introduced multi-party key generation for dynamic and large group. We described implementation of group operations. The multi-party key generation technique needs comparatively less communication and computational costs, and therefore it is useful for practical applications.

REFERENCES

- [1] Diffie W., Hellman M.: 'New directions in cryptography', IEEE Trans .Info Theory, 1976, 22,(6)pp 644-654
- [2]STEINER M., TSUDIK WADINER **Key** Agreement in Dynamic Peer Groups IEEE Trans.Parallel Distrib.Syst. 2000, 11,(8) pp 769-780
- [3] Stallings 'Cryptography and Network Security' (Pearson Education)
- [4] Becker 'Communication complexity of Group Key Distribution' 5th conference Computer and Communication Security, 1998 pp 1-6
- [5] G.P.Biswas 'Diffie-Hellman Key Exchange Extended to Multi-party Key and Multiple Two party Keys 'IEEE Trans 2006
- [6] Steiner M., Diffie-Hellman Key Distribution Extended To groups 3rd ACM Conference Computer And Communication Security
- [7] Zheng.S Manz.D 'S communication computation efficient group key algorithm for large and dynamic groups Comput, Netw., 2007,51,pp 69-9

A Framework for Systematic Database Denormalization

YMA PINTO

Goa University , India
ymapinto@gmail.com

Abstract- It is currently the norm that relational database designs should be based on a normalized logical data model. The primary objective of this design technique is data integrity and database extensibility. The Third Normal Form is regarded by academicians and practitioners alike to be point at which the database design is most efficient. Unfortunately, even this lower level normalization form has a major drawback with regards to query evaluation. Information retrievals from the database can result in large number of joins which degrades query performance. So you need to sometimes break theoretical rules for real world performance gains. Most existing Conceptual Level RDBMS data models provide a set of constructs that only describes “what data is used” and does not capture “how the data is being used”. The question of “how data is used” gets embedded in the implementation level details. As a result, every application built on the existing database extracts the same or similar data in different ways. If the functional use of the data is also captured, common query evaluation techniques can be formulated and optimized at the design phase, without affecting the normalized database structure constructed at the Conceptual Design phase. This paper looks at denormalization as an effort to improve the performance in data retrievals made from the database without compromising data integrity. A study on a hierarchical database table shows the performance gain - with respect to response time – using a denormalization technique.

Keywords: denormalization, database design, performance tuning, materialized views, query evaluation

I. INTRODUCTION

Most of the applications existing today have been built, or are still being built using RDBMS or ORDBMS technologies. The RDBMS is thus not dead, as stated by Arnon-Roten [Roten_Gal, 2009]. Van Couver, a software engineer with vast experience in databases at Sun Microsystems, emphasizes the fact that RDBMSs are here to stay but do require improvements in scalability and performance bottlenecks [Couver, 2009].

Normalization is the process of putting one fact and nothing more than one fact in exactly one appropriate place. Related facts about a single entity are stored together, and every attribute of each entity is non-transitively associated to the Primary Key of that entity. This design technique results in enhanced data integrity and removes insert, update and

delete anomalies that would have otherwise been present in a non-normalized database. Another goal of normalization is to minimize redesign of the database structure. Admittedly, it is impossible to predict every need that your database design will have to fulfill and every issue that is likely to arise, but it is important to mitigate against potential problems as much as possible by a careful planning.

Arguably, normalizing your data is essential to good performance, and ease of development, but the question always comes up: "How normalized is normalized enough?" Many books on normalization, mention that 3NF is essential, and many times BCNF, but 4NF and 5NF are really useful and well worth the time required to implement them [Davidson, 2007]. This optimization, however, results in performance degradation in data retrievals from the database as a large number of joins need to be done to solve queries [Date, 1997] [Inmon, 1987] [Schkolnick and Sorenson, 1980].

"Third normal form seems to be regarded by many as the points where your database will be most efficient ... If your database is overnormalized you run the risk of excessive table joins. So you denormalize and break theoretical rules for real world performance gains." [Sql Forums, 2009].

There is thus a wide gap between the academicians and the database application practitioners which needs to be addressed. Normalization promotes an optimal design from a logical perspective. Denormalization is a design level that needs to be mitigated one step up from normalization. With respect to performance of retrieval, denormalization is not necessarily a bad decision if implemented following a systematic approach to large scale databases where dozens of relational tables are used.

Denormalization is an effort that seeks to optimize performance while maintaining data integrity. A denormalized database is thus not equivalent to a database that has not been normalized. Instead, you only seek to denormalize a data model that has already been normalized. This distinction is important to understand, because you go from normalized to denormalized, not from nothing to denormalized. The mistake that some software developers do is to directly build a denormalized database considering only the performance aspect. This only optimizes one part of the equation, which is database reads. Denormalization is a design level that is one step up from normalization and should not be treated naively. Framing denormalization against normalization purely in the context of performance

is unserious and can result in major application problems [Thought Clusters, 2009]. We need to understand how and when to use denormalization

This paper is organized as follows: Section 1 introduces the concept and current need for denormalization. Section 2 provides us a background of the related work in this area from the academic and the practitioners' point of view. Section 3 makes a strong case for denormalization while Section 4 presents the framework for a systematic denormalization. Section 5 elucidates some denormalization techniques that can be followed during the database design life cycle and shows the performance gain of this technique over a Hierarchical Normalized Relation.

II. BACKGROUND AND RELATED WORK

Relational Databases can be roughly categorized into Transaction Processing (OLTP) and Data Warehouse (OLAP). As a general rule, OLTP databases use normalized schema and ACID transactions to maintain database integrity as the data needs to be continuously updated when transactions occur. As a general rule, OLAP databases use unnormalized schema (the "star schema" is the paradigmatic OLAP schema) and are accessed without transactions because each table row is written exactly one time and then never deleted nor updated. Often, new data is added to OLAP databases in an overnight batch, with only queries occurring during normal business hours [Lurie M., IBM, 2009] [Microsoft SQL Server guide] [Wiseth, Oracle].

Software developers and practitioners mention that database design principles besides normalization, include building of indices on the data and denormalization of some tables for performance. Performance tuning methods like indices and clustering data of multiple tables exist, but these methods tend to optimize a subset of queries at the expense of the others. Indices consume extra storage and are effective only when they work on a single attribute or an entire key value. The evaluation plans sometimes skip the secondary indexes that are created by users if these indices are nonclustering [Khaldtiance, 2008].

Materialized Views can also be used as a technique for improving performance [Vincent et al, 97] but these consume vast amount of storage and their maintenance results in additional runtime overheads. Blind application of Materialized Views can actually result in worse query evaluation plans and should be used carefully [Chaudhuri et al, 1995]. View update techniques have been researched and a relatively new method of updating using additional views has been proposed [Ross et al, 1996].

In the real world, denormalization is sometimes necessary. There have been two major trends in the approach to demoralization. The first approach uses a "non normalized ERD" where the entities in the ERD are collapsed to decrease the joins. In the second approach, denormalization is done at the physical level by consolidating relations, adding synthetic attributes and creating materialized views to improve performance. The disadvantage of this approach

is the overheads required in view consistency maintenance. Denormalization is not necessarily a bad decision if implemented wisely [Mullins, 2009].

Some denormalization techniques have been researched and implemented in many strategic applications to improve query response times. These strategies are followed in the creation of data warehouses and data marts [Shin and Sanders, 2006] [Barquin and Edelstein] and are not directly applicable to an OLTP system. Restructuring a monolithic Web application composed of Web pages that address queries to a single database into a group of independent Web services querying each other also requires denormalization for improved performance [Wei Z et al, 2008].

Several researches have developed a list of normalization and denormalization types, and have subsequently mentioned that denormalization should be carefully deployed according to how the data will be used [Hauns, 1994] [Rodgers, 1989]. The primary methods that have been identified are: combining tables, introducing redundant data, storing derivable data, allowing repeating groups, partitioning tables, creating report tables, mirroring tables. These "denormalization patterns" have been classified as Collapsing Relations, Partitioning Relations, Adding Redundant Attributes and Adding Derived Attributes [Sanders and Shin, 2001]

III. A CASE FOR DENORMALIZATION

Four main arguments that have guided experienced practitioners in database design have been listed here [26]

The Convenience Argument

The presence of calculated values in tables' aids the evaluation of adhoc queries and report generation. Programmers do not need to know anything about the API to do the calculation.

The Stability Argument

As systems evolve, new functionality must be provided to the users while retaining the original. History data may still need to be retained in the database.

The Simple Queries Argument

Queries that involve join jungles are difficult to debug and dangerous to change. Eliminating joins makes queries simpler to write, debug and change

The Performance Argument

Denormalized databases require fewer joins in comparison to normalized relations. Computing joins are expensive and time consuming. Fewer joins directly translates to improved performance.

Denormalization of Databases, ie, a systematic creation of a database structure whose goal is performance improvement, is thus needed for today's business processing requirements. This should be an intermediate step in the DataBase Design Life Cycle integrated between the Logical DataBase Design Phase and the Physical DataBase Design Phase. Retrieval performance needs dictate very quick retrieval capability for

data stored in relational databases, especially since more accesses to databases are being done through Internet. Users are concerned with more prompt responses than an optimum design of databases. To create a Denormalization Schema the functional usage of the operational data must be analyzed for optimal Information Retrieval.

Some of the benefits of denormalization can be listed:

- (a) Performance improvement by
- Precomputing derived data
 - Minimizing joins
 - Reducing Foreign Keys
 - Reducing indices and saving storage
 - Smaller search sets of data for partitioned tables
 - Caching the Denormalized structures at the Client for ease of access thereby reducing query/data shipping cost.

(b) Since the Denormalized structures are primarily designed keeping in mind the functional usage of the application, users can directly access these structures rather than the base tables for report generation. This also reduces bottlenecks at the server.

A framework for denormalization needs to address the following issues:

- (i) Identify the stage in the DataBase Design Life Cycle where Denormalization structures need to be created.
- (ii) Identify situations and the corresponding candidate base tables that cause performance degradation.
- (iii) Provide strategies for boosting query response times.
- (iv) Provide a method for performing the cost-benefit analysis.
- (v) Identify and strategize security and authorization constraints on the denormalized structures.

Although (iv) and (v) above are important issues in denormalization, they will not be considered in this paper and will be researched on later.

IV. A DENORMALIZATION FRAMEWORK

The framework presented in this paper differs from the papers surveyed above in the following respects:

It does not create denormalized tables with all contributing attributes from the relevant entities, but instead creates a set of Denormalized Structures over a set of Normalized tables. This is an important and pertinent criteria as these structures can be built over existing applications with no “side effects of denormalization” over the existing data.

The entire sets of attributes from the contributing entities are not stored in the Denormalized structure. This greatly reduces the storage requirements and redundancies.

The Insert, Update and Delete operations (IUDs) are not done to the denormalized structures directly and thus do not

violate data integrity. The IUDs to data are done on the Base Tables and the denormalized structures are kept in synch by triggers on the base tables.

Since the denormalized structures are used for information retrieval, they need to consider the authorization access that users have over the base tables.

The construction of the “Denormalization View” is not an intermediate step between the Logical and the Physical Design phases, but needs to be consolidated by considering all 3 views of the SPARC ANSI architectural specifications.

Most existing Conceptual Level RDBMS data models provide a set of constructs that describes the structure of the database [Elmashree and Navathe]. This higher level of conceptual modeling only informs the end user “what data is used” and does not capture “how the data is being used”. The question of “how data is used” gets embedded in the implementation level details. As a result, every application built on the existing database extracts the same or similar data in different ways. If the functional use of the data is also captured, common query evaluation techniques can be formulated and optimized at the design phase, without affecting the normalized database structure constructed at the Conceptual Design phase. Business rules are descriptive integrity constraints or functional (derivative or active) and ensure a well functioning of the system. Common models used during the modeling process of information systems do not allow the high level specification of business rules except a subset of ICs taken into account by the data model [Amghar and Mezaine, 1997].

The ANSI 3 level architecture stipulates 3 levels – The External Level and the Conceptual Level, which captures data at rest, and the Physical Level which describes how the data is stored and depends on the DBMS used. External Schemas or subschemas relate to the user views. The Conceptual Schema describes all the types of data that appear in the database and the relationships between data items. Integrity constraints are also specified in the conceptual schema. The Internal Schema provides definitions for stored records, methods of representation, data fields, indexes, and hashing schemes. Although this architecture provides the application development environment with logical and physical data independence, it does not provide an optimal query evaluation platform. The DBA has to balance conflicting user requirements before creating indices and consolidating the Physical schema.

The reason denormalization is at all possible in relational databases is because, courtesy of the relational model, which creates lossless decompositions of the original relation, no Information is lost in the process. The Denormalized structure can be reengineered and populated from the existing Normalized database and vice-versa. In a distributed application development environment the Denormalization Views can be cached on the client resulting in a major performance boost by saving run time shipping

costs. It would require only the Denormalization View Manager to be installed on the Client. A High Level Architecture that this framework considers is defined as follows:

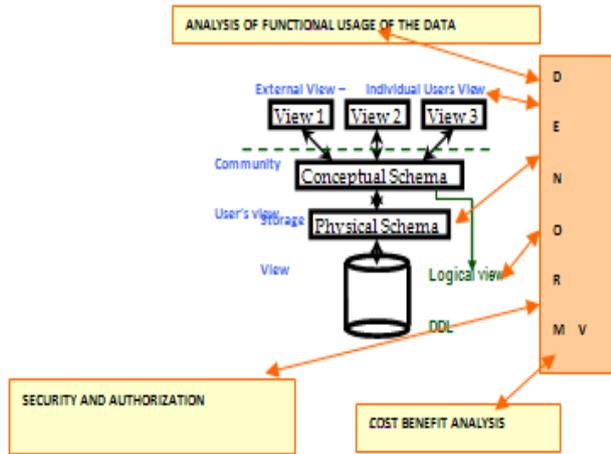


Figure 1: A High Level Architecture of the Relational Data Information Retrieval Model

To realize the potential of the Denormalization View, efficient solutions to the three encompassing issues are required:

Denormalization View design: Determining what data and how it is stored and accessed in the Denormalization Schema

Denormalization View maintenance: Methods to efficiently update the data in the Denormalized schema when base tables are updated.

Denormalization View exploitation: Making efficient use of denormalization views to speed up query processing (either entire queries or sub queries)

Extensive research has been done on subquery evaluation on materialized views [Afrati et al, 2001] [Chirkova et al, 2006] [Halevy, 2001]

The inputs that are required for the construction of the Denormalized schema can be identified as:

- the logical and external views schema design,
- the physical storage and access methods provided by the DBMS,
- the authorization the users have on the manipulation and access of the data within the database,
- the interaction (inter and intra) between the entities,

- the number of entities the queries involve,
- the usage of the data (ie, the kind of attributes and their frequency of extraction within queries and reports),
- the volume of data being analyzed and extracted in queries (cardinality and degree of relations, number and frequency of tuples, blocking factor of tuples, clustering of data, estimated size of a relation),
- the frequency of occurrence and the priority of the query,
- the time taken by the queries to execute(with and without denormalization).

The problem can now be stated as “Given a logical schema with its corresponding database statistics and a set of queries with their frequencies, arrive at a set of denormalized structures that enhances query performance”

A few definitions are required

Defn 1: A *Relational Data Information Retrieval System (RDIRS)* has as its core components (i) a set of Normalized Relations {R} (ii) a set of Integrity Constraints {ICs} (iii) a set of data access methods {A} (iv) a set of Denormalization Structures {DS} and (v) a set of queries and subqueries that can be defined and evaluated on these relations.

Each component of the RDIRS, by definition, can have dynamic elements resulting in a flexible and evolvable system.

Defn 2: A “*Denormalized Structure (DSM)*” is a relvar [Date, Kannan, Swamynathan] comprising of the Denormalized Schema Design and the Denormalized Structure Manager.

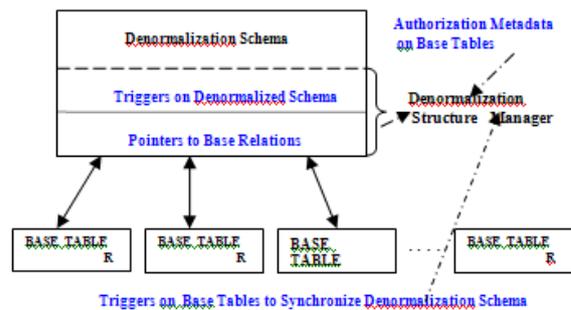


Figure 2: The Denormalization View

A system cannot enforce truth, only consistency. Internal Predicates (IPs) are what the data means to the system and External Predicates (EPs) are what data means to a user. The EPs result in criterion for acceptability of IUD operations on the data, which is an unachievable goal [Date, Kannan, Swamynathan], especially when Materialized Views are created. In the framework presented in this paper, IUDs on the Denormalized Structures are never rejected as these are automatically propagated to the base relations where the

Domain and Table level ICs are enforced. Once the base relations are updated, the Denormalized Schema Relation triggers are invoked atomically to synchronize the data, ensuring simultaneous consistency of Base and Denormalized tables. Further, the primary reason for the Denormalization Structures is faster information retrieval and not data manipulation; hence no updates need be made to the Denormalization Schema directly.

Every Normalized Relation requires a Primary Key which satisfies the Key Integrity Constraint. This PK maintains uniqueness of tuples in the database and is not necessarily the search key value for users. For the **RDIRS** we define

Defn 3: An *Information Retrieval Key (IRK)* is a (set of) attributes that the users most frequently extract from an entity. The IRK is selected from amongst the mandatory attribute values which gives the end user meaningful information about the entity.

For ex, an employee table may have an Empid as its PK, but the IRK could be EmpName and Contact No.

Defn 4: An *Information Retrieval Tree (IRT)* is a Query Evaluation Tree which has as its components the operators required to extract the information from the database and the relvars that contribute to an optimized Data Extraction Plan. The IRT consists of relational algebra operations along the intermediate nodes and the relvars in the leaf nodes (base relations, views, materialized views or denormalization structures) and is a requisite for cost benefit analysis and query rewrites.

Researchers and Practitioners [Inmon, 1987] [Shin and Sanders, 2006] [Mullins, 2009] create the denormalized tables by creating a schema with all the attributes from the participating entities. This results in (i) additional storage and redundancy (ii) slows down the system on updates to data (iii) creates a scenario for data anomalies.

Defn 5: The *Denormalization Schema (DS)* in the RDIR Model is a relation that has as its attributes only the PKs, the IRKs and the URowIds (Universal Row Id) of the participating or contributing Base Relations.

The storage of only the PK, IRKs and URowIds is justifiable as most often, end users are interested in only the significant attributes of an entity. If required, the remaining attributes can be obtained from the base table using the RowId field stored in the Denormalized Scheme. The URowIds are chosen as they can even support row-ids on remote foreign tables.

It is interesting to note that even when a “select * “ clause is used in an adhoc query, it is either because the user is unaware of the attributes of the entity or is uninterested in the attribute per se, but is actually looking for other information.

The Denormalization Schema Design is an input to the Query Optimizer for collapsing access paths, resulting in the IRT which is then submitted to the Query Evaluation Engine.

Although the metadata tables are query able at the server, the Denormalized Structure Manager can have its own metadata stored locally (at the node where the DSs are stored).

DS_Metadata_Scheme(DS_Name,DS_Trigger_Name,DS_Procedure_Name, DS_BT1_Name, Creator,DS_BT1_Trigger_Name,DS_BT2_Trigger_Name,DS_BT1_Authorization,DS_BT2_Authorization)

V. DENORMALIZATION TECHNIQUES

Denormalization looks at normalized databases which have operational data, but whose performance degrades during query evaluation. There are several indicators which will help to identify systems and tables which are potential denormalization candidates.

The techniques that can be used are summarized below:

a. Pre joined Tables

Application: When two or more tables need to be joined on a regular basis and the cost of joins is prohibitive.

This happens when Foreign Keys become a part of a relation or when transitive dependencies are removed.

Denormalization Technique: Collapse the relations.

b. Report Tables

Application: When the application requires creation of specialized reports that requires lot of formatting and data manipulation.

Denormalization Technique: The report table must contain the mandatory columns required for the report

c. Fragmenting Tables

Application: If separate pieces of a normalized table are accessed by different and distinct groups of users or applications, then the original relation can be split into two (or more) denormalized tables; one for each distinct processing group. The relation can be fragmented horizontally or vertically by preserving losslessness.

Denormalization Technique: When horizontal fragmentation is done, the predicate must be chosen such that rows are not duplicated.

When vertical fragmentation is done, the primary key must be included in the fragmented tables. Associations between the attributes of the relation must be considered. Projections that eliminate rows in the fragmented tables must be avoided.

5.1: An illustration of the above techniques

Consider the following Normalized database (3NF) relations:
(Primary Keys are in Red , Foreign keys are in Blue)

Customer (**CustomerNo**, CustomerName, ContactId)

Order **OrderNo**, **CustomerNo**, OrderDate, ShipRecdDate,
VATax, Local_Tax, **ShiptoContactId**, **BillToContactId**)

ContactInfo (**ContactId**, Name, Street, City, State Country,
Zip)

ContactPhone (**ContactId**, **PhoneNo**)

Item (**ItemNo**, ItemName, ItemPrice, ItemPart, **SubItemNo**)

OrderItem (**OrderNo**, **ItemSerialNo**, **ItemNo**, Quantity)

d. Redundant Data

Application: Sometimes one or more columns from one table are accessed whenever data from another table is accessed. If this happens frequently they could be stored as redundant data in the tables.

Denormalization Technique: The columns that are duplicated in the relation to avoid a lookup (join) should be used by a large number of users but should not be frequently updated.

e. Repeating Groups

Application: When repeating groups are normalized they are implemented as distinct rows instead of distinct columns resulting in less efficient retrieval. These repeating groups can be stored as a nested table within the original parent table.

Before deciding to implement repeating groups, it is important to consider if the data will be aggregated or compared within the row or if the data would be accessed collectively, otherwise SQL may slow down query evaluation.

Denormalization Technique: Repeating groups can be stored as “setoff(values)” - SQL Extensions - within the

table removing the restriction on the number of values that can repeat.

f. Derivable Data

Application: If the cost of deriving data using complicated formulae is prohibitive then the derived data can be stored in a column. It is imperative that the stored derived value needs to be changed when the underlying values that comprise the calculated value change.

Denormalization Technique: Frequently used aggregates can be precomputed and materialized in an appropriate relation.

g. Hierarchical Speed Tables

Application: A hierarchy or a recursive relation can be easily supported in a normalized relational table but is difficult to retrieve information from efficiently. Denormalized “Speed Tables” are often used for faster data retrieval.

Denormalization Technique: Not only the immediate parent of a node is stored, but all of the child nodes at every level are stored.

Some of the major reports identified and that need to be generated from this database:

- What are the current outstanding orders along with their shipping and Billing details
- For a given order, find all the parts that are ordered along with the subparts of that part.
- Prepare a voucher for a given order.
- For orders that were paid for on the same date that the Shipment was received, give a 10% discount if the amount exceeds a value ‘x’ and a 20% discount if the amount exceeds a value ‘y’.
- Retrieve all sub items that item number 100 contains
- Find all subparts that have no subpart.

The Denormalized Schema thus constructed over the Normalized Tables to improve performance and using the techniques described above:

DN_Oust_Order (**OrderNo**, **CustomerNo**, **OrderDate**, **ShipToContactInfo_Name**, **ShipToContactPhone_PhNo**, **BillToContactInfo_Name**, **BillToContactPhone_PhNo**, **ShipToContactInfo_URowId**, **BillToContactInfo_URowId**)

DN_Aggregate (OrderNo, OrderDate, TotalAmt, Discount)

DN_Voucher (OrderNo, OrderDate, ItemName, ItemPrice, Quantity, DN_Aggregate_RowId)

DN_Item_Hierarchy (**Main_ItemId**, **Sub_ItemId**, **Child_Level**, **Is_Leaf**, **Item_URowId**)

These tables can be created using the

**create materialized view
build immediate
refresh fast on commit
enable query rewrite**

clauses provided by the DBMSs. The URowIds of the Base Table rows can also be selected and inserted into the Denormalized Schema Extensions.

The DN_Aggregate Tables need to be created using the **with schemabinding**

clause .

The Denormalized Hierarchy tables can be created using the

**connect by prior
start with
level**

clauses.

The CONNECT BY prior clause can automatically handle insertions.

5.2: A Performance Study on Hierarchical Queries

The Hierarchical Technique for Denormalization needs to be further illustrated.

Considering the Normalized Item Data consisting of data shown below (partial view of the database)

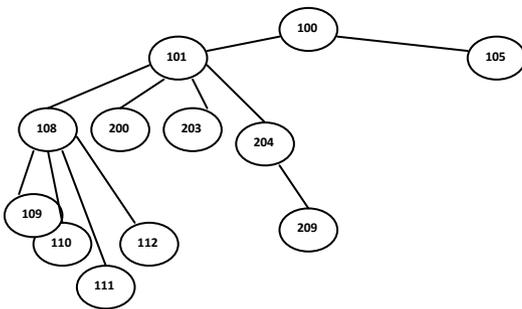


Figure 3: Partial Hierarchical Item Data

The **Normalized Relation** for the Hierarchical Item Table would be stored as

ItemNo	ParentItemNo	OtherItemDetails
100		...
101	100	...
105	100	...
108	101	...
200	101	...
203	101	...
204	101	...
109	108	...
110	108	...
111	108	...
112	108	...
209	204	...

```

2.
select itemno,itemname,parentitem from item start with
parentitem=100 connect by prior itemno=parentitem ;

69 rows selected.

Elapsed: 00:00:00.17

3.
select parentitem,childitemno,itemname from dn_item_hier
where parentitem=100
    
```

Consider a query **“Find all items that are contained in item 100”** that requires to be run on the above table. This involves finding the child nodes at every level of the hierarchy.

A Solution to the above query:

```

Select ItemNo from item where
ParentItemNo='100'
Union
Select ItemNo from item where ParentItemNo
in
(Select ItemNo from item
where ParentItemNo='100')
Union
Select ItemNo from item where ParentItemNo
in
(Select ItemNo from item where
ParentItemNo in
(Select ItemNo from item
where ParentItemNo='100'))
    
```

This retrieval query, besides being extremely inefficient, one needs to know the maximum depth of the hierarchy.

The **Denormalized Schema** for the Item Information in the RDIRS :

DN_Item_Hierarchy (ParentItemNo, ChildItemNo, ItemName, ChildLevel, IsLeaf, Item_URowId)

The ChildLevel ascertains the level in the hierarchy that the child node is at; IsLeaf specifies if that node has further child nodes and makes queries like **“Find all items that have no subparts”** solvable efficiently.

The (part) extension of the DN_Item_Hierarchy Schema

ParentItemNo ChildItemNo ItemName ChildLevel IsLeaf ItemRowId

100			101	SubPart1	
100	1	N		
100			105	SubPart2	1
100	N			
100			108	SubPart3	2
100	N			
100			200	SubPart4	
100	2	Y		
100			203	SubPart5	
100	2	Y		
100			204	SubPart6	
100	2	N		
100			109	SubPart7	
100	3	Y		
100			110	SubPart8	
100	3	Y		
100			111	SubPart9	
3	Y			
100			112	SubPart10	
100	3	Y		
100			209	SubPart11	
100	3	Y	...		
101			108	SubPart3	
101	2	N	...		
101			200	SubPart4	
101	2	N	...		
101			203	SubPart5	
101	2	N	...		
101			204	SubPart6	
101	2	N	...		
108			109	SubPart7	
108	3	Y	...		
108			110	SubPart8	
108	3	Y	...		
108			111	SubPart9	
108	3	Y	...		
108			112	SubPart10	
108	3	Y	...		
204			209	SubPart11	
204	3	Y	...		

The results are as shown :

```

1.
Set timing on;

select itemno,itemname,parentitem from item1 where
itemno in

(select itemno from item1 where parentitem=100

union

select itemno from item1 where parentitem in

(select itemno from item1 where parentitem=100)

union

select itemno from item1 where parentitem in

(select itemno from item1 where parentitem in

(select itemno from item1 where parentitem=100)));

69 rows selected.

Elapsed: 00:00:00.31
    
```

A Solution to the above query “*Find all items that are contained in item 100*” can now be written as:

Select itemno from dn_item_hierarchy where parentitemno=100;

To study the performance improvement using denormalization, the normalized item table was created with 100 tuples, 70 tuples had the main root level as 100. The maximum child level nodes was 4.

VI. CONCLUSIONS AND FUTURE WORK

Although each new RDBMS release usually brings enhanced performance and improved access options that may reduce the need for denormalization, there will be many occasion where even these popular RDBMSs will require denormalized data structures. Denormalization will continue to remain an integral part of DataBase Design. A detailed authorization and access matrix which is stored along with the Denormalization view will further enhance performance. This and a detailed strategy for cost benefit analysis will be the next stage in the subject of my research.

REFERENCES

[1] Afrati F., Chen Li, and Ullman J D. “Generating efficient plans using views”. In SIGMOD, pages 319–330, 2001.
 [2] Amghar Y. and Mezaine M., “Active database design” ,Comad 97, Chennai, India.
 [3] Chaudhuri, Krishnamurthy R, Potamianos S, and Shim K, “Optimizing Queries using materialized views”, In Proceedings of the 11th International Conference on Data Engineering (Taipei, Taiwan, Mar.), ,1995,pp. 190--200.

- [4] Chirkova R., Chen Li, and J Li, "Answering queries using materialized views with minimum size" ,. *Vldb Journal* 2006, 15 (3), pp. 191-210.
- [5] Date C.J, "The Normal is so ...interesting", *DataBase programming and Design*, Nov 1997,pp 23-25
- [6] Halevy A. "Answering queries using views: A survey." *In VLDB*, 2001.
- [8] Inmon W.H, "Denormalization for Efficiency, " *ComputerWorld*", Vol 21 ,1987 pp 19-21
- [9] Ross K., Srivastava D. and Sudarshan S., "Materialized View Maintenance and integrity constraint checking : trading space for time", *ACM Sigmod Conference* 1996,pp 447 -458
- [10] Rodgers U., "Denormalization: why, what and how?" *Database Programming and Design*,1989 (12) ,pp 46-53
- [11] Sanders G. and Shin S.K, "Denormalization Effects on Performance of RDBMS", *Proceedings of the 34th International Conference on Systems Sciences*, 2001
- [12] Schkolnick M., Sorenson P. , "Denormalization :A performance Oriented database design technique" , *Proceedings of the AICA 1980 Congress ,Italy*.
- [13] Shin S.K and Sanders G.L., " Denormalization strategies for data retrieval from data warehouses " ,*Decision support Systems*, VolVol. 42, No. 1, pp. 267-282, 2006
- [14] Vincent M., Mohania M. and Kambayashi Y., "A Self-Maintainable View maintenance technique for data warehouses" ,8th Int. Conf on Management of Data, Chennai,India
- [15] Wei Z., Dejun J., Pierre G.,Chi C.H, Steen M.,"Service-Oriented Data Denormalization for Scalable Web Applications" , *Proceedings of the 17th International WWW Conference* 2008, Beijing, China
- [16] Barquin R., Edelstein H., "Planning and Designing the Data Warehouse", Prentice Hall
- [7] Hauns M., "To normalize or denormalize, that is the question", *Proceedings of 19th Int.Conf for Management and Performance Evaluation of Enterprise computing Systems*, San Diego,CA,1994,pp 416-423
- [17] Date C.J. ,Kannan A., Swamynathan S.,"An Introduction to Database Systems " , ,8th Ed.,Pearson Education
- [18] Elmashree R. and Navathe S.,"Fundamentals of Database Systems",3rd Ed, Addison Weisley.
- [19] Davidson L., "Ten common design mistakes " , software engineers blog, Feb 2007
- [20] Downs K.,"The argument for Denormalization",*The Database Programmer*,Oct 2008
- [21] Khaldtiance S., "Evaluate Index Usage in Databases", *SQL Server Magazine*, October 2008
- [22] Lurie M.,IBM, "Winning Database Configurations
- [23] Mullins C, "Denormalization Guidelines " , *Platinum Tecnology Inc.,Data administration Newsletter*, Accessed June 2009.
- [24] Microsoft - *SQL Server 7.0 Resource Guide "Chapter 12 - Data Warehousing Framework"*
- [25] Roten-Gal-Oz A. Cirrus minor in "Making IT work" *Musings of an Holistic Architect*, Accessed June 2009
- [26] Van Couver D. on his blog "Van Couvering is not a verb", Accessed June 2009
- [27] Wiseth K, Editor-in-Chief of Oracle Technology News, in "Find Meaning",Accessed June 2009
- [28] Thought Clusters on software, development and programming, website -- March 2009
- [29] website – <http://www.sqlteam.com/Forums/>, Accessed July 2009

Experiments with Self-Organizing Systems for Texture and Hardness Perception

MAGNUS JOHNSON _ CHRISTIAN BALKENIUS
Lund University Cognitive Science, Lund University, Sweden

Abstract- We have experimented with different SOM-based architectures for bio-inspired self-organizing texture and hardness perception systems. To this end we have developed a microphone based texture sensor and a hardness sensor that measures the compression of the material at a constant pressure. We have implemented and successfully tested both monomodal systems for texture and hardness perception, bimodal systems that merge texture and hardness data into one representation and a system which automatically learns to associate the representations of the two submodalities with each other. The latter system employs the novel Associative Self- Organizing Map (A-SOM). All systems were trained and tested with multiple samples gained from the exploration of a set of 4 soft and 4 hard objects of different materials with varying textural properties. The monomodal texture system was good at mapping individual objects in a sensible way. This was also true for the hardness system which in addition divided the objects into categories of hard and soft objects. The bimodal system was successful in merging the two submodalities into a representation that performed at least as good as the best recognizer of individual objects, i.e. the texture system, and at the same time categorizing the objects into hard and soft. The A-SOM based system successfully found representations of the texture and hardness submodalities and also learned to associate These with each other.

Keywords: haptic perception, hardness perception, texture perception, self-organizing map, A-SOM

I. INTRODUCTION

Two important submodalities in haptic perception are texture and hardness perception. In non-interactive tasks, the estimation of properties like the size and the shape of an external object is often to a large extent based on vision only and haptic perception will only be employed when visual information about the object is not reliable. This might happen for example at bad lighting conditions or when the object is more or less occluded. Haptic submodalities like texture and hardness perception are different in this respect. These submodalities are especially important because they provide information about the outer world that is unavailable for all the other perception channels.

An efficient haptic perception system with several submodalities, as well as multimodal perceptual systems in general, should be able to associate different submodalities or modalities with each other. This provides an ability to activate the subsystem for a modality even when its sensory input is limited or nonexistent as long as there are activities

in subsystems for other modalities, which the subsystem has learned to associate with certain patterns of activity, which usually comes together with the patterns of activity in the other subsystems. For example, in humans the sensory information gained when the texture of an object is felt in the pocket can invoke visual images of the object or a feeling for its hardness.

There have been some previous studies of texture and hardness in robotics. For example Hosoda et al (7) have built an anthropomorphic fingertip with distributed receptors consisting of two silicon rubber layers of different hardness. The silicon rubber layers contain two different sensors, strain gauges and polyvinylidene fluoride films, which yield signals that in a test enabled the discrimination of five different materials pushed and rubbed by the fingertip. Mayol-Cuevas et al (18) describe a system for tactile texture recognition, which employs a sensing pen with a microphone that is manually rubbed over the explored materials. The system uses a supervised Learning Vector Quantization (LVQ) classifier system to identify with 93% accuracy 18 common materials after signal processing with the Fast Fourier Transform (FFT). Edwards et al (5) have used a vinyl record player with the needle replaced with an artificial finger with an embedded microphone to quantify textural features by using a group of manufactured discs with different textural patterns. Campos and Bajcsy (4) have explored haptic Exploratory Procedures (EPs) based on human haptic EPs proposed by Lederman and Klatzky, among them an EP for hardness exploration in which the applied force is measured for a given displacement.

Our previous research on haptic perception has resulted in the design and implementation of a number of versions of three different working haptic systems. The first system (8) was a system for haptic size perception. It used a simple three-fingered robot hand, the LUCS Haptic Hand I, with the thumb as the only movable part. The LUCS Haptic Hand I was equipped with 9 piezo electric tactile sensors. This system used Self-Organizing Maps (SOMs) (15) and a neural network with leaky integrators. A SOM is a self-organizing neural network that finds a low-dimensional discretized and topology preserving representation of the input space. The system successfully learned to categorize a test set of spheres and cubes according to size.

The second system (9) was a system for haptic shape perception and used a three-fingered 8 d.o.f. robot hand, the LUCS Haptic Hand II, equipped with a wrist for horizontal rotation and a mechanism for vertical repositioning. This

robot hand was equipped with 45 piezo electric tactile sensors. This system used active explorations of the objects by several sequential grasps to gather tactile information, which together with the positioning commands to the actuators (thus a kind of pseudoproprioception) were cross coded by, either tensor product (outer product) operations or a novel neural network, the Tensor Multiple Peak SOM (T-MPSOM) (9). The cross-coded information was categorized by a SOM. The system successfully learned to discriminate between different shapes as well as between different objects within a shape category when tested with a set of spheres, blocks and cylinders.

The third system (10) (13) was a bio-inspired selforganizing system for haptic shape and size perception based solely on proprioceptive data from a 12 d.o.f. anthropomorphic robot hand with proprioceptive sensors (11). Several kinds of self-organizing neural networks were successfully tested in this system. The system was trained with 10 different objects of different sizes from two different shape categories and tested with both the training set and a novel set with 6 previously unused objects. It was able to discriminate the shape as well as the size of the objects in both the original training set and the set of new objects.

This paper explores SOM-based systems of texture and hardness perception, bimodal systems that merges these submodalities (12) and a self-organizing texture and hardness perception system, which automatically learns to associate the representations of the two submodalities with each other (14). The latter system is based on a novel variant of the SOM, called Associative Self-Organizing Map, A-SOM, (14). All systems employ a microphone based texture sensor and/or a hardness sensor that measures the compression of the material at a constant pressure.

The systems are bio-inspired in the sense that they employ SOMs or a variation of the SOM to represent the two submodalities texture and hardness, and the SOM shares many features with cortical brain maps (16). Our approach is also biologically motivated in the sense that different submodalities are integrated. That different submodalities are integrated in unimodal association areas in the human brain is well established (19). The texture sensor is also bio-inspired. Our system is based on the transduction of vibrations from a metal edge which are transmitted to a microphone. This parallels the humans system where the mechanoreceptors respond to vibrations as well (6).

II. SENSORS IN THE EXPERIMENTS

All systems discussed in this paper employ at least one of two sensors (Fig. 1) developed at Lund University Cognitive Science (LUCS). One of these sensors is a texture sensor and the other is a hardness sensor. The texture sensor consists of a capacitor microphone with a tiny metal edge mounted at the end of a moveable lever, which in turn is

mounted on an RC servo. When exploring a material, the lever is turned by the RC servo, which moves the microphone with the attached metal edge along a curved path in the horizontal plane. This makes the metal edge slide over the explored material, which creates vibrations in the metal edge with frequencies that depend on the texture of the material. The vibrations are transferred to the microphone since there is contact between it and the metal edge. The signals are then sampled and digitalized by a NiDaq 6008 (National Instruments) and conveyed to a computer via a USB-port. The FFT is then applied to the input to yield a spectrogram of 2049 component frequencies.

The hardness sensor consists of a stick mounted on an RC servo. During the exploration of a material the RC servo tries to move to a certain position, which causes a downward movement of the connected stick at a constant pressure. In the control circuit inside the RC servo there is a variable resistor that provides the control circuit with information whether the RC servo has been reaching the wanted position or not. In our design, we measure the value of this variable resistor at the end of the exploration of the material and thus get a measure of the end position of the stick in the exploration. This end position is proportional to the compression of the explored material. The value of the variable resistor is conveyed to a computer and represented in binary form. The actuators for both the sensors are controlled from the computer via a SSC-32 controller board (Lynxmotion Inc.). The software for the system presented in this paper is developed in C++ and runs within the Ikaros system (1)(2). Ikaros provides an infrastructure for computer simulations of the brain and for robot control.

III. EXPLORATIONS OF OBJECTS

Each system described in this paper has been trained and tested with one or both of two sets of samples. One set consists of 40 samples of texture data and the other set consists of 40 samples of hardness data. These sets have been constructed by letting the sensors simultaneously.

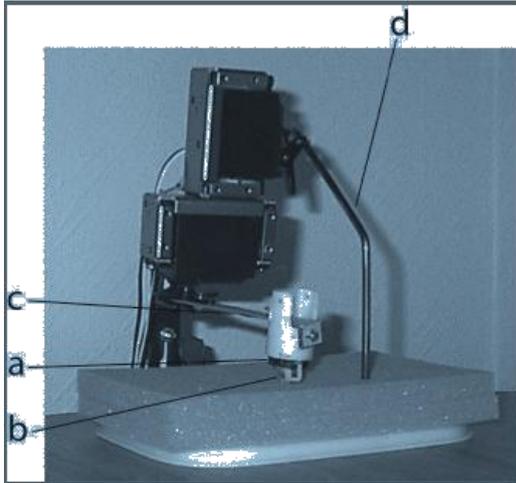


Figure 1: The texture and hardness sensors while exploring a piece of foam rubber. The texture sensor consists of a capacitor microphone (a) with a metal edge (b) mounted at the end of a moveable lever (c), which in turn is mounted on a RC servo. The hardness sensor consists of a stick (d) mounted on a RC servo. The servo belonging to the hardness sensor contains a variable resistor that provides a measure of the turning of the servo, and thus the displacement of the stick, which is proportional to the compression of the explored material. The actuators are controlled via a SSC-32 controller board (Lynxmotion Inc.). The measure of the resistance of the variable resistor in the RC servo for the hardness sensor and the microphone signal of the texture sensor are digitalized using a NiDaq 6008 (National Instruments) and conveyed to the computer via a USB-port. explore each of the eight objects described in Table 1 five times.

During the hardness exploration of an object the tip of the hardness sensor (Fig. 1d) is pressed against the object with a constant force and the displacement is measured.

The exploration with the texture sensor is done by letting its lever (Fig. 1c) turn 36 degrees during one second. During this movement the vibrations from the metal edge (Fig. 1b) slid over the object are recorded by the microphone (Fig. 1a) mounted at the end of the stick.

The output from the texture sensor from all these explorations has then been written to a file after the application of the FFT. Likewise, the output from the hardness sensor has been written to a file represented as binary numbers. The hardness samples can be considered to be binary vectors of length 18 whereas the texture samples can be considered to be vectors of length 2049. The eight objects have various kinds of texture and can be divided into two groups, one with four rather soft objects and one with four rather hard objects. During the exploration, the objects were fixed in the same location under the sensors.

IV. SOM-BASED SYSTEMS

1) A Texture Perception System

The texture perception system (Fig. 2A) is a monomodal system. This means that the raw sensor output from the texture sensor is transformed by the FFT into a spectrogram containing 2049 frequencies, and the spectrogram represented by a vector is in turn conveyed to a SOM, which uses softmax activation (3) with the softmax exponent equal to 10. After training the SOM will represent the textural properties of the explored objects.

We have experimented with different parameter settings of the texture SOM, both with the aim to get a well-working monomodal system and to get a system that would serve well as a part of a bimodal system, and we reached the conclusion that a well-working set of parameters is to use a SOM with 15×15 neurons with a plane topology. A torus topology was also tested but turned out to be less effective than a plane topology. The sort of topology used influences the behaviour of the SOM at the borders. With plane topology the activations

from the objects in the training set tend to be close to the borders, which turned out to be good when the texture perception system was used as a part of the combined monomodal/bimodal system described below. We also experimented with different decay rates of the Gaussian neighbourhood function. We came to the conclusion that a neighbourhood radius of 15 at the start of the training phase, which decreased gradually until it was approximately 1 after 1000 iterations, and stayed at 1 during the rest of the training phase, was satisfactory. This system and all the others were trained during 2000 iterations before evaluation. We reasoned that it would be good if the neighborhood had shrunk to a small value after about 1000 iterations in order to let the bimodal SOM of the combined system, described below, get enough iterations to self-organize. In other words, the idea was that the texture SOM should be rather well organized after 1000 iterations

2) A Hardness Perception System

The hardness perception system is also monomodal. In this system (Fig. 2B), the raw sensor output from the hardness sensor, represented as a binary number with 18 bits, is conveyed to a SOM, which like the texture system uses softmax activation with the softmax exponent equal to 10. After training, the SOM will represent the hardness property of the explored objects.

As in the case of the texture system we have experimented with different parameter settings of the hardness SOM, and for the same reasons. In this case we also tested a lot of different sizes of the monomodal. Table 1: The eight objects used in the experiments with all systems. The objects a-h were used both for training and testing. The materials of the objects are presented and they are subjectively classified as either hard or soft. A rough subjective estimation of their textural properties is also provided.

Label	Object	Estimated Hardness	Estimated Texture
a	Foam Rubber	Soft	Somewhat Fine
b	Hardcover Book	Hard	Shiny
c	Bundle of Paper	Hard	Fine
d	Cork Doily	Hard	Rough
e	Wood Doily	Hard	Fine
f	Bundle of Denim	Soft	Somewhat Fine
g	Bundle of Cotton Fabric	Soft	Somewhat Fine
h	Terry Cloth Fabric	Soft	Rough

SOM. This was because preliminary experiments indicated that it could be a good idea to use a very small sized SOM for hardness in the combined system described below. A small sized hardness SOM seemed to self-organize solely according to the hardness property and not distinguish individual objects, and since the texture SOM was better at distinguishing individual objects we did not want the hardness part to impair this although we wanted it to make the bimodal representation become organized according to hardness as well. We tried SOMs with planar as well as torus topology and with 15×15 , 10×10 , 5×5 , 2×2 or 1×2 neurons. All variants started with a neighbourhood size that covered the whole SOM and the rates of decay of the neighbourhood were adjusted so that the neighbourhood would shrink to a radius of approximately 1 after about 1000 iterations. As we had expected the 15×15 neurons SOM (with plane topology) was best in this monomodal system but we also found that, as suspected, all tested sizes but one indeed organized to divide the objects into the categories hard and soft. The exception was the SOM with only 1×2 neurons, which did not preserve the division of hard and soft objects in a good way.

3) A Combined Monomodal and Bimodal System

In this system (Fig. 2C) we experimented with different ways of combining the output from the monomodal SOMs to an input for the bimodal SOM. First we tried a method for cross coding that we have used in other contexts. In this method a two-vector input self-organizing neural network called T-MPSOM (9) that self-organizes into something similar to the tensor product operation, was used to combine the outputs from the monomodal SOMs. In previous research, the T-MPSOM was very successful in coding proprioceptive and tactile information and it also worked in the current system. However, we also experimented with a simpler method of combining the monomodal outputs, which was also superior for this aim. This method was simply to combine the activity of the monomodal SOMs, re-arranged into vectors, by creating a new vector by putting the hardness output vector after the texture output vector.

The monomodal texture SOM used 15×15 neurons with the same parameter setting as in the texture system. In the case of the monomodal hardness SOM we tried two different variations, namely a 2×2 neurons SOM and a 15×15 neurons SOM with the settings specified in the

hardness system above. Both worked fine but the variation with the 2×2 neurons SOM yielded the best representation in the bimodal SOM. The bimodal SOM had similar settings as the monomodal texture SOM, but the decay rate of the neighbourhood was set to decrease the neighbourhood radius to one in 2000 iterations.

4) A Bimodal System

In the bimodal system (Fig. 2D) we combined the output from the texture sensor, after transformation into a spectrogram by a FFT, with the raw hardness sensor output expressed as a binary number by the same method as in the combined system described above, i.e. by putting the output vector from the hardness sensor after the output vector from the FFT. This means that this system has no monomodal representations. The combined vector was used as input to a bimodal SOM with the same settings as in the combined system above. Also in this system we tried to use T-MPSOM but with a worse result than with this simpler method.

5) Results and Discussion

The mapping of the objects (a-h in Tab. 1) used in the experiments with the different SOM-based systems is depicted in Fig. 3. Each image in the figure corresponds to a SOM in a fully trained system and each cell in an image corresponds to a neuron in the corresponding SOM. A filled circle in a cell is supposed to mean that particular neuron is the centre of activation in one or several explorations. In Fig. 3A the mapping of individual texture explorations with the texture system have been encircled. As can be seen, most objects are mapped at separate sites in the SOM (c, d, e, f, h).

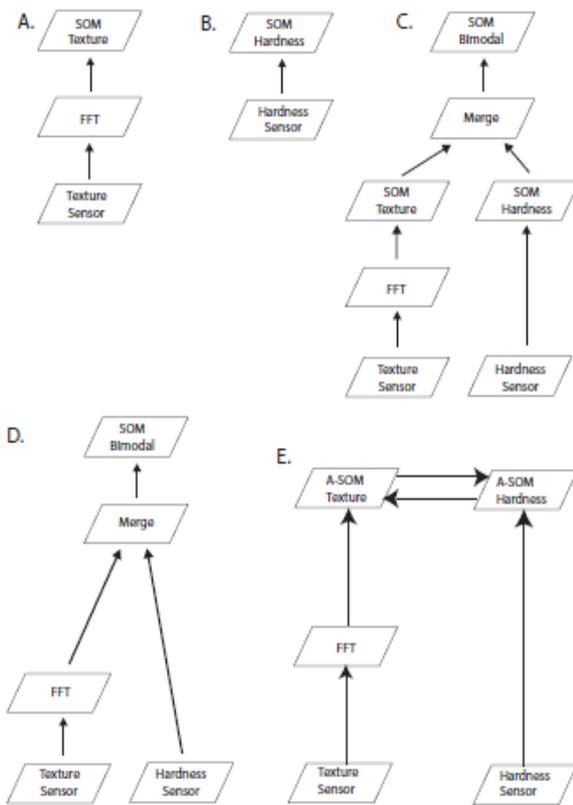


Figure 2: Schematic depiction of the systems architectures.

A: A monomodal system for texture perception. The raw sensor output is transformed by the FFT into a spectrogram containing 2049 frequencies. The spectrogram represented by a vector is conveyed to a SOM.

B: A monomodal system for hardness perception. The raw sensor output represented as a binary vector with 18 elements is conveyed to a SOM.

C: A system with both monomodal and bimodal representations. This system could be seen as a merging and an extension of the previous two systems, or likewise the previous two systems could be seen as the monomodal level of this system. The output from the texture SOM and the output from the hardness SOM is merged, i.e. a new vector is created by transforming the activations of the texture SOM and the hardness SOM into vectors and putting them after each other. The merged vector is used as input to a bimodal SOM. This means that in this system there are self-organizing representations of texture and hardness as well as a combined representation of both.

D: A bimodal system. This system directly combines the output from the FFT and the binary output from the hardness sensor into a new vector in the same way as described in the previous system, but without the step with monomodal representations. The combined vector is used as input to a bimodal SOM. **E:** The A-SOM based system. This system consists of two monomodal subsystems, which

develop monomodal representations (A-SOMs) of hardness and texture that learn to associate their activities. The hardness subsystem uses the raw sensor output from the hardness sensor as input to an A-SOM, which finds a representation of the hardness property of the explored objects. The texture subsystem transforms the raw sensory data by the aid of a FFT module and then forwards it to another A-SOM, which finds a representation of the textural properties of the explored objects. The two A-SOMs learn to associate their respective activities are some exceptions though, namely a, b and g. So the texture system is able to discriminate between individual objects, although not perfectly.

The SOM in the hardness system, depicted in Fig. 3B, also maps different objects at different sites in the SOM but not as good as the texture system. The hardness system recognizes b, f and h perfectly and blurs the other more or less. However, the system perfectly discriminates hard from soft objects.

The combined monomodal and bimodal system (Fig. 3C), which as mentioned above can be seen as a merging and extension of the texture system and the hardness system (with 2×2 neurons in the SOM), discriminate hard from soft objects well. In two explorations the hard/soft category is undetermined. This is so because one exploration of an object a and one exploration of an object g have the same centre of activation. It also discriminates perfectly between the objects b, d, f and h.

The bimodal system (Fig. 3D) discriminates perfectly between the objects c, d, e, f and h, i.e. the same objects as in the texture system. Moreover, it also discriminates hard from soft objects, although in seven explorations the hard/soft category is undetermined because three explorations of the object a and four explorations of the object b have the same centre of activation.

V. A BIMODAL SYSTEM WITH ASSOCIATED SOM REPRESENTATIONS

(1) A-SOM

The A-SOM (Fig. 4) can be considered as a Self-Organizing Map (SOM) (15) which learns to associate the activity of an external A-SOM or SOM with its own activity. It consists of an $I \times J$ grid of neurons with a fixed number of neurons and a fixed topology. Each neuron n_{ij} is associated with two weight vectors $w^a_{ij} \in R_n$ and $w^b_{ij} \in R_m$ where m equals the number of neurons in an external A-SOM or SOM. w^a_{ij} is initialized randomly to numbers between 0 and 1, whereas all elements of w^b_{ij} are initialized to 0.

At time t each neuron n_{ij} receives two normalized input vectors $x^a(t) \in R_n$ and $x^b(t) \in R_m$. The neuron c

associated with the weight vector $w_c^a(t)$ most similar to the input vector $x^a(t)$ is selected:

$$c = \arg \max_c \{ \|x^a(t)w_c^a(t)\| \} \quad (1)$$

The activity in the neuron n_{ij} is given by

$$y_{ij}(t) = [y_{ij}^{input}(t) + y_{ij}^{extern}(t)] / 2 \quad (2)$$

where

$$y_{ij}^{input}(t) = G(\|n_{ij} - c\|) \quad (3)$$

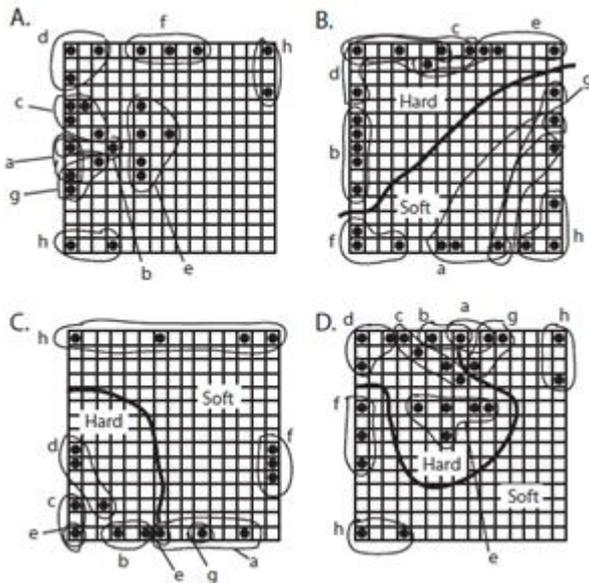


Figure 3: The mapping of the objects used in the experiments with the different SOM-based systems. The characters a-h refer to the different objects in Table 1. Each image in the figure corresponds to a SOM in a fully trained system and each square represents a neuron in the SOM, which consists of $15 \times 15 = 225$ neurons. A filled circle in a cell is supposed to mean that that particular neuron is the centre of activation for one or several explorations. The occurrence of a certain letter at more than one place means that the corresponding object has different centres of activation during different explorations of the same object, i.e. all letters of a certain kind represents all occurring centres of activation in the SOM when the system was tested with the corresponding object. A: The monomodal SOM in the texture system. The centres of activation of all instances of each object have been encircled. The objects c, d, e, f and h are mapped at non-overlapping sites in the SOM, whereas the objects a, b and g are not. This can be interpreted as that the texture system is able to discriminate between individual objects, although not perfectly. B: The monomodal SOM in the hardness system. In this system the objects b, f and h are

perfectly recognized, whereas the others are not. Moreover, the system perfectly discriminates hard from soft objects. C: The bimodal SOM in the combined monomodal and bimodal system. In this system the objects b, d, f and h are perfectly recognized, whereas the others are not. It also discriminates hard from soft objects. In two explorations the hard/soft category is undetermined, because one exploration of an object a and one exploration of an object g have the same centre of activation. D: The bimodal SOM in the bimodal system. In this system the objects c, d, e, f and h are perfectly discriminated, i.e. the same objects as in the texture system. Moreover, it also discriminates hard from soft objects, although in seven explorations the hard/soft category is undetermined because three explorations of the object a and four explorations of the object b have the same centre of activation.

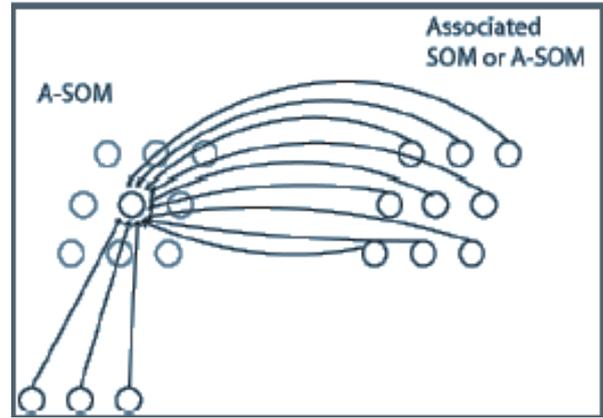


Figure 4: The connectivity of the A-SOMnetwork. During training each neuron in an A-SOM receives two kinds of input. One kind of input is the native input, which corresponds to the input an ordinary SOM receives. The other kind of input is the activity of each neuron in an associated SOM or A-SOM. In the fully trained A-SOM, activity can be triggered by either native input or by activity in the associated SOM or ASOM, or both.

and

$$y_{ij}^{extern}(t) = x^b(t)w_{ij}^b(t) \quad (4)$$

$G()$ is a Gaussian function with $G(0) = 1$, and $k \cdot k$ is the Euclidean distance between two neurons.

$G()$ is a Gaussian function with $G(0) = 1$, and $\| \cdot \|$ is the Euclidean distance between two neurons.

The weights w_{ijk}^a are adapted by

$$w_{ijk}^a(t+1) = w_{ijk}^a(t) + \alpha(t)G_{ijc}(t) [x_k^a(t) - w_{ijk}^a(t)] \quad (5)$$

where $0 \leq \alpha(t) \leq 1$ is the adaptation strength with $\alpha(t) \rightarrow 0$ when $t \rightarrow \infty$ and the neighbourhood function $G_{ijc}(t)$ is a Gaussian function decreasing with time.

The weights w_{ijl}^b are adapted by

$$w_{ijl}^b(t+1) = w_{ijl}^b(t) + \beta x_l^b(t) [y_{ij}^{input}(t) - y_{ij}^{extern}(t)] \quad (6)$$

where β is the constant adaptation strength.

(2) The System

This system is a bimodal model of haptic hardness and texture perception (Fig. 3). It consists of two monomodal subsystems (hardness and texture), which develop monomodal representations (A-SOMs) that are associated with each other. The subsystem for hardness uses the raw sensor output from the hardness sensor, represented as a binary number with 18 bits and conveys it to an A-SOM with 15×15 neurons with plane topology. After training, this A-SOM will represent the hardness property of the explored objects.

In the subsystem for texture, the raw sensor output from the texture sensor is transformed by a FFT module into a spectrogram containing 2049 frequencies, and the spectrogram which is represented by a vector, is in turn conveyed to an A-SOM with 15×15 neurons with plane topology. After training, this A-SOM will represent the textural properties of the explored objects.

The two subsystems are coupled to each other in that their A-SOMs also receive their respective activities as associative input.

Both A-SOMs began their training with the neighborhood radius equal to 15. The neighbourhood radius was decreased at each iteration by multiplication with 0.998 until it reached the minimum neighbourhood size 1. Both A-SOMs started out with $\alpha(0) = 0.1$ and decreased it by multiplication with 0.9999. β where set to 0.35 for both A-SOMs.

The system was trained with samples from the training set, described above, by 2000 iterations before evaluation.

(3) Results and Discussion

The results of the experiment with the A-SOM based system are depicted in Fig. 5. The 6 images depict the centres of activation when the fully trained system was tested with the test set (described above) constructed with the aid of the

objects a-h in Table 1. Images 5A, 5B and 5C correspond to the texture representing A-SOM. Likewise the images 5D, 5E and 5F correspond to the hardness representing A-SOM. Each cell in an image represents a neuron in the A-SOM. In the images 5A, 5B, 5D and 5E there are black circles in some of the cells. This means that the corresponding neurons in the A-SOM are the centre of activation for one or several of the samples in the test set. The centers of activation from the samples in the test set corresponding to each object in Tab 1 when only native input was provided have been encircled in 5A and 5D to show where different objects are mapped in the A-SOMs. Native input should be understood as texture input for the texture representing A-SOM, and hardness input for the hardness representing A-SOM. These results with only native input to the A-SOMs are similar to our earlier results with the hardness and texture sensors together with ordinary SOMs described above and in (12). The encircling are also present in the other four images to show how the A-SOMs are activated when there are both native and external input provided to the system (5B and 5E), and when there are only external input provided (5C and 5F). External input should be understood as hardness input in the case of the texture representing A-SOM, and as texture input in the case of the hardness representing ASOM.

Fig. 5A depicts the texture representing A-SOM in the fully trained system when tested with the test set (only native texture input). As can be seen, most objects are mapped at separate sites in the A-SOM (c, d, e, f, h). There are some exceptions though, namely a, b and g. So the system is able to discriminate between individual objects when provided with native input only, although not perfectly.

The hardness representing A-SOM in the fully trained system when tested with the test set (only native hardness input), depicted in Fig. 5D, also maps different objects at different sites in the A-SOM but not as good as the texture representing A-SOM. The hardness representing A-SOM recognizes b, f and h perfectly and blurs the other more or less. However, the hardness representing A-SOM perfectly discriminates hard from soft objects.

When the texture representing A-SOM receives native texture input as well as external hardness input (as can be seen in Fig. 5B) its activations are very similar to those in Fig. 5A. Likewise when the hardness representing A-SOM receives native hardness input as well as external texture input (as can be seen in Fig. 5E) its activations are very similar to those in Fig. 5D.

Fig. 5C depicts the activations in the texture representing A-SOM when it receives only external hardness input. As can be seen this external hardness input very often triggers an activity similar to the activity following native texture input. Likewise, Fig. 5F depicts the activity in the hardness representing A-SOM when it receives only external texture

input. Even in this case the external input very often triggers an activity similar to the activity following native input. This means that when just one modality in the system receives input, this can trigger activation in the other modality similar to the activation in that modality when receiving native input. Thus an object explored by both sensors during training of the system can trigger a more or less proper activation in the representations of both modalities even when it can be explored by just one sensor during testing. However, as can be seen in Fig. 5C and Fig. 5F, the activity triggered solely by external input does not map every sample properly. The worst cases are the objects c, d and g in the texture representing A-SOM (Fig. 5C) and the objects a, b and g in the hardness representing A-SOM (Fig. 5D). As can be seen in Fig. 5D, the objects c, d and g are not distinguishable in the hardness representing A-SOM, and the objects a, b and g are not distinguishable in the texture representing A-SOM (Fig. 5A). Thus the external activity patterns for these objects are overlapping and the receiving A-SOM cannot be expected to learn to map these patterns correctly even if the objects were well separated by the A-SOM when it received native input.

VI. CONCLUSION

We have experimented with several self-organizing systems for object recognition based on textural and/or hardness input. The texture sensor employed is based on the transmission of vibrations to a microphone when the sensor slides over the surface of the explored material. The hardness sensor is based on the measurement of displacement of a stick when pressed against the material at a constant pressure. The results are encouraging, both for the monomodal systems, the bimodal systems and the A-SOM based system. The bimodal systems seem to benefit from both submodalities and yield representations that are better than those in the monomodal systems. This is particularly true because the bimodal representations preserve the discrimination ability of the monomodal texture system and also seem to preserve the way that the system groups the objects. The influence of the hardness input makes the bimodal representation organized according to hardness as well. The A-SOM based system is able to discriminate individual objects based on input from each submodality and to discriminate hard from soft objects. In addition input to one submodality can trigger an activation pattern in the other submodality, which resembles the pattern of activity the object would yield if explored with the sensor for this other submodality.

Our experiments with texture complement those done by Edwards et al (5) and Hosoda et al (7) because they only show that the signals from their sensors are in principle useful as texture sensors whereas we actually implemented working self-organizing systems. When compared to the work done by Mayol-Cuevas et al (18) our texture experiments differ in that we use a sensor that is not

manually rubbed over the material as their pen, but moved by an actuator built into the sensor. A couple of extensions in our experiments when compared to all the previously mentioned experiments and to the work done by Campos and Bajcsy (4) are that we also experimented with both hardness and texture and the association between these two submodalities.

Because of the successful results of basing a texture sensor on a microphone and basing hardness perception on the measurements of displacements at a constant applied pressure, we will in the future try to integrate this approach with our haptic systems. In other words, we will carry out experiments in which we equip future robot hands with microphone based texture sensors and measure hardness by letting a finger press on the explored material at a constant pressure while measuring the displacement. This could result in systems that explore objects and more or less immediately gain information about the objects shape, size, hardness and textural properties. This will yield a system that is able to discriminate between equally shaped and sized objects made of different materials.

We will also continue our experimentations with the A-SOM. We will continue by testing the ability of the ASOM when there are very many categories to see if the A-SOM works equally well in that case. We will also try to implement an extended version of the A-SOM, which can be associated with several external A-SOMs or SOMs. In this way we could build not only bimodal but multimodal systems. Such systems would trigger proper activation patterns in the other modalities when receiving input from just one modality. This extension should be quite straight forward. It could be accomplished by just adding a new weight vector to each neuron for every new associated A-SOM or SOM. The activity of the neurons would be calculated by adding the native activity and the activities coming from all associated A-SOMs or SOMs and divide the sum with the total numbers of activities.

Another very interesting continuation would be to test the A-SOM technology in systems that integrate visual and haptic subsystems. This would allow the visual system to trigger a proper apprehension of a robot hand when it is about to grasp an object.

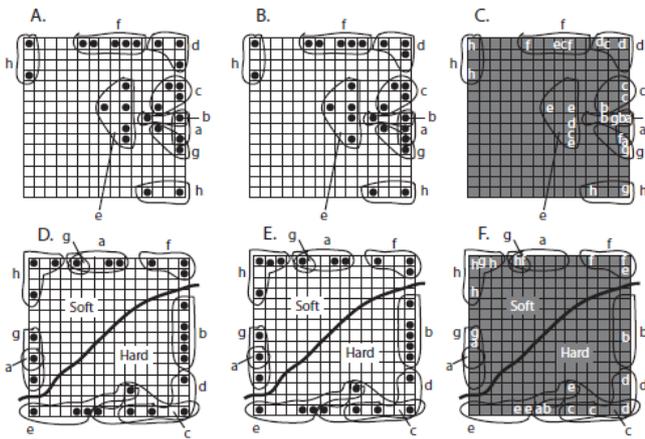


Figure 5: The mapping of the objects used in the experiments with the A-SOM based system. The characters a-h refers to the objects in Table 1. The images in the uppermost row correspond to the texture representing A-SOM and the images in the lowermost row correspond to the hardness representing A-SOM. Each cell in an image represents a neuron in the A-SOM, which consists of $15 \times 15 = 225$ neurons. A filled circle in a cell is supposed to mean that that particular neuron is the centre of activation for one or several explorations. The occurrence of a certain letter in the rightmost images means that there are one or several centers of activation for that particular object at that particular place. The centers of activation from the samples in the test set corresponding to each object in Tab 1 when only native input was provided have been encircled in the images. A: The texture representing A-SOM when tested with native texture input. Most objects are mapped at separate sites so the system is able to discriminate between individual objects when provided with native input, although not perfectly. B: The texture representing A-SOM when tested with native texture input together with external hardness input. Its activations are very similar to those in A. C: The texture representing A-SOM when it receives only external hardness input. This often triggers an activity similar to the activity following native texture input. D: The hardness representing ASOM when tested with native hardness input maps different objects at different sites and it perfectly discriminates hard from soft objects. E: The hardness representing A-SOM when tested with native hardness input together with external texture input. Its activations are very similar to those in D. F: The hardness representing A-SOM when it receives only external texture input. This often triggers an activity similar to the activity following native hardness input.

REFERENCES

[1] Balkenius, C., & Mor'en, J. (2003). From isolated components to cognitive systems. *ERCIM News*, April 2003, 16.

- [2] Balkenius, C., Mor'en, J. & Johansson, B. (2007). Building system-level cognitive models with Ikaros. *Lund University Cognitive Studies*, 133.
- [3] Bishop, C. M. (1995). *Neural Networks for Pattern Recognition*. Oxford University Press, Oxford, New York.
- [4] Campos, M. & Bajcsy, R. (1991). A Robotic Haptic System Architecture, *Proceedings of the 1991 IEEE International Conference on Robotics and Automation*, 338-343.
- [5] Edwards, J., Melhuish, C., Lawry, J., & Rossiter, J. (2007). Feature Identification for Texture Discrimination from Tactile Sensors. *Proceedings of TAROS 2007*, University of Wales, Aberystwyth, UK, 115-121.
- [6] Gardner, E.P., Martin, J.H., & Jessell, T.M. (2000). The bodily senses. In Kandel, E.R., Schwartz, J.H., & Jessell, T.M., (ed.). *Principles of neural science*, 430-450, McGraw-Hill.
- [7] Hosoda, K., Tada, Y., & Asada, M. (2006). Anthropomorphic robotic soft fingertip with randomly distributed receptors, *Journal of Robotics and Autonomous Systems*, 54, 2, 104-109.
- [8] Johnsson, M., & Balkenius, C. (2006). Experiments with Artificial Haptic Perception in a Robotic Hand, *Journal of Intelligent and Fuzzy Systems*.
- [9] Johnsson, M., & Balkenius, C. (2007a). Neural Network Models of Haptic Shape Perception, *Journal of Robotics and Autonomous Systems*, 55, 720-727
- [10] Johnsson, M., & Balkenius, C. (2007b). Experiments with Proprioception in a Self-Organizing System for Haptic Perception. *Proceedings of TAROS 2007*, University of Wales, Aberystwyth, UK, 239-245.
- [11] Johnsson, M., & Balkenius, C. (2007c). LUCS Haptic Hand III - An Anthropomorphic Robot Hand with Proprioception. *LUCS Minor 13*.
- [12] Johnsson, M., & Balkenius, C. (2008a). Recognizing Texture and Hardness by Touch. *Proceedings of IROS 2008*, Nice, France.
- [13] Johnsson, M., Gil Mendez, D., & Balkenius, C. (2008b). Touch Perception with SOM, Growing Cell Structures and Growing Grids. *Proceedings of TAROS 2008*, University of Edinburgh, Edinburgh, UK, 79-85.
- [14] Johnsson, M., & Balkenius, C. (2008c). Associating SOM Representations of Haptic Submodalities. *Proceedings of TAROS 2008*, University of Edinburgh, Edinburgh, UK, 124-129.
- [15] Kohonen, T. (1988). *Self-Organization and Associative Memory*, Berlin Heidelberg, Springer-Verlag.
- [16] Kohonen, T. (1990). The self-organizing map. *Proceedings of the IEEE*, 78, 9, 1464-1480.
- [17] Mazid, A.M. & Russell, R.A. (2006). A Robotic Optotactile Sensor for Assessing Object Surface Texture. *IEEE Conference on Robotics, Automation and Mechatronics*, 2006, 1 - 5.
- [18] Mayol-Cuevas, W. W., Juarez-Guerrero, J., & Munoz-Gutierrez, S. (1998). A First Approach to Tactile Texture Recognition, *IEEE International Conference on Systems, Man, and Cybernetics*, 5, 4246-4250.

- [19] Saper, C.B., Iversen, S., & Frackowiak, R. (2000). Integration of sensory and motor function: The association areas of the cerebral cortex and the cognitive capabilities of the brain. In Kandel, E.R., Schwartz, J.H., & Jessell, T.M., (ed.). *Principles of neural science*, 349-380, McGraw-Hill.
- [20] Takamuku, S., Gómez, G., Hosoda, K. and Pfeifer, E. (2007). Haptic discrimination of material properties by a robotic hand, in: Proc. of the 6th IEEE Int. Conf. on Development and Learning.

Diagnosing Parkinson by using Artificial Neural Networks and Support Vector Machines

DAVID GIL A, MAGNUS JOHNSON B

^aComputing Technology and Data Processing, University of Alicante, Spain

^bLund University Cognitive Science, Sweden

Abstract- Parkinson's Disease (PD) is the second most common neurodegenerative affliction only surpassed by Alzheimer's Disease (AD). Moreover, it is expected to increase in the next decade with accelerating treatment costs as a consequence. This situation leads us towards the need to develop a Decision Support System for PD. In this paper we propose methods based on ANNs and SVMs to aid the specialist in the diagnosis of PD. Data recorded during 195 examinations carried out on 31 patients was used to verify the capacity of the proposed system. The results show a high accuracy of around 90%.

Key words: Parkinson diagnosis, Parkinson, mental disorder, expert systems in medicine, artificial intelligence in medicine, artificial neural networks, support vector machines.

I. INTRODUCTION

Parkinson's Disease (PD) is the second most common neurodegenerative affliction after Alzheimer's disease (AD). Studies from Olmsted County (Mayo Clinic) [Elbaz et al., 2002] have computed the lifetime risk of developing Parkinson's disease to 2 percent for men and 1.3 percent for women. The greater incidence in men is repeatedly confirmed.

PD is a progressive neurological disorder characterised by tremor, rigidity, and slowness of movements. It is associated with progressive neuronal loss in the substantia nigra and other brain structures. Non-motor features, such as dementia and dysautonomia, occur frequently, especially in advanced stages of the disease. Diagnosis depends on the presence of two or more cardinal motor features such as rest tremor, bradykinesia, or rigidity [Hughes et al., 1992]. Functional neuroimaging holds the promise of improved diagnosis and allows assessment in early disease [Piccini and Whone, 2004]. Two studies draw attention to the difficulties in the diagnosis of the disease in the early stages [Tolosa et al., 2006]. In a prospective clinicopathological study, Rajput [Rajput et al., 1991] showed that initial clinical diagnosis within 5 years from the disease onset was correct in 65% of the cases. After a mean duration of 12 years, the final diagnosis of PD by the clinician was confirmed with autopsy in 76% of cases. Similarly, among 800 patients in the Deprenyl and Tocopherol Antioxidative Therapy for PD study with mild early parkinsonism [Jankovic et al., 2000] judged to have PD, 89%

were later reported to have an alternative diagnosis on the basis of multi-factorial, clinical diagnostic criteria.

Having so many factors to analyze to diagnose PD, specialist normally makes decisions by evaluating the current test results of their patients. Moreover, the previous decisions made on other patients with a similar condition are also done by them. These are complex procedures, especially when the number of factors that the specialist has to evaluate is high (high quantity and variety of these data). For these reasons, PD diagnosis involves experience and highly skilled specialists.

The use of classifier systems in medical diagnosis is increasing gradually. Recent advances in the field of artificial intelligence have led to the emergence of expert systems and Decision Support Systems (DSS) for medical applications.

Moreover, in the last few decades computational tools have been designed to improve the experiences and abilities of doctors and medical specialists in making decisions about their patients. Without doubt the evaluation of data taken from patients and decisions of experts are still the most important factors in diagnosis. However, expert systems and different Artificial Intelligence (AI) techniques for classification have the potential of being good supportive tools for the expert. Classification systems can help in increasing accuracy and reliability of diagnoses and minimizing possible errors, as well as making the diagnoses more time efficient [Akay, 2008].

Some of the related work about using AI techniques to aid in PD diagnosis and other types of mental disorder are [Cohen, 1994] [Cohen, 1998] [Björne and Balkenius, 2005] [Berdia and Metz, 1998] [Ivanitsky and Naumov, 2008] [Loukas and Brown, 2004].

Motivated by the usefulness of such an expert system or DSS, the aim of this work is to propose a method to aid the specialist in the diagnosis of PD, thus increasing accuracy and reducing costs. The quantity and variety of the data recorded during examinations makes AI tools useful to improve the final diagnosis. AI tools are also useful in retrospective studies. Nowadays such historical studies are easier, better and more precise due to the increased use of automated tools that allow storage and retrieval of large

volumes of medical data. The proposal is to build a system using Artificial Neural Networks (ANNs) and Support Vector Machines (SVMs). These two classifiers, which are widely used for pattern recognition, should provide a good generalization performance in the diagnosis task. The usage of such classifiers would reinforce and complement the diagnosis of the specialists and their methods in the diagnosis tasks.

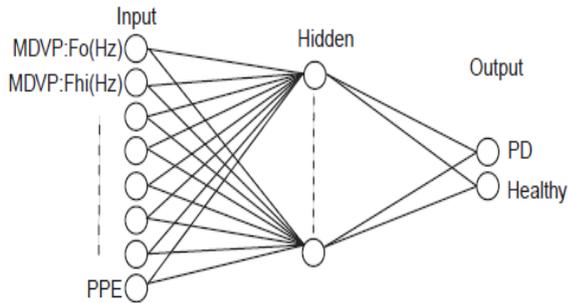


Fig.1. The architecture of the MLP network (input layer, hidden layer and output layer). The input layer represents the input data (the input data is described in section 4.1). The usage of a hidden layer enables the representation of data sets that are not linearly separable. The output layer represents the classification result. The weights and the threshold of the MLP are calculated during an adaptation process.

(ANNs) and Support Vector Machines (SVMs). These two classifiers, which are widely used for pattern recognition, should provide a good generalization performance in the diagnosis task. The usage of such classifiers would reinforce and complement the diagnosis of the specialists and their methods in the diagnosis tasks.

The remaining part of the paper is organized as follows: First, we explain the Parkinson data set used in the experimentation; second, we give a brief description of ANNs; Third, we describe the basic concepts of SVMs; Fourth, we describe our testing of the system and analyze the results; Finally, we draw the relevant conclusions and suggest future lines of research.

II. MULTILAYER PERCEPTRON

In this study we have used a Multi-Layer Perceptron (MLP) network with two layers. A two-layer MLP network is a fully connected feedforward neural network consisting of an input layer (which is not counted since its neurons are only for representation and thus do no processing), a hidden layer, and an output layer (healthy or ill) which represents

the classification result (see figure 1) [Ripley, 1996] [Haykin, 1998][Bishop]. Each neuron (see figure 2) in the input and hidden layers is connected to all neurons in the next layer by weighted connections. These neurons (see figure 2) compute weighted sums of their inputs and adds a threshold. The resulting sums are used to calculate the activity of the neurons by applying a sigmoid activation function.

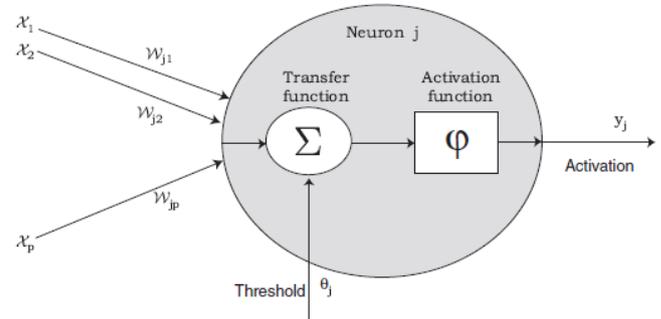


Fig. 2. A neuron in the hidden or output layer in the MLP. In the experimentation section the number hidden neurons in the MLP will be established.

This process is defined as follows:

$$v_j = \sum_{i=1}^p w_{ji}x_i + \theta_j, \quad y_j = f_j(v_j) \quad (1)$$

where v_j is the linear combination of inputs $x_1; x_2; \dots; x_p$; and the threshold θ_j , w_{ji} is the connection weight between the input x_i and the neuron j , and f_j is the activation function of the j th neuron, and y_j is the output. The sigmoid function is a common choice of the activation function. It is defined as:

$$f(t) = \frac{1}{1 + e^{-t}} \quad (2)$$

A single neuron in the MLP is able to linearly separate its input space into two subspaces by a hyper plane defined by the weights and the threshold. The weights define the direction of this hyper plane whereas the threshold term μ_j offsets it from origo.

The MLP network uses the backpropagation algorithm [Rumelhart et al., 1986], which is a gradient descent method, for the adaptation of the weights. This algorithm works as follows

The backpropagation MLP is a supervised ANN. This means the network is presented with input examples as well as the corresponding desired output. The backpropagation

algorithm adapts the weights and the thresholds of the neurons in a way that minimizes the error function E

$$E = \frac{1}{2} \sum_{p=1}^n (d_p - y_p)^2$$

where y_p is the actual output and d_p the desired output for input pattern p .

The minimization of E can be accomplished by gradient descent, i.e. the weights are adjusted to change the value of E in the direction of its negative gradient. The exact updating rules can be calculated by applying derivatives and the chain rule (for the weights between the input and the hidden layer).

III. SVM

In this section, the basic concepts of the SVM are described. More thorough descriptions can be found in [Burges, 1998] [Theodoridis and Koutroumbas, 2003] [Hsu et al., 2003]. A typical two class problem as the one shown in figure 3 is similar to the problem of diagnosing patients as either ill or healthy.

For a classification problem, it is necessary to first try to estimate a function

$$f : \mathbb{R}^N \rightarrow \{\pm 1\}$$

using training data, which are l N -dimensional patterns x_i and class labels y_i , where $(x_1; y_1); \dots; (x_l; y_l) \in \mathbb{R}^N \times \{\pm 1\}$

(4)

such that f will classify new samples $(x; y)$ correctly.

Given this classification problem the SVM classifier, as described by [Vapnik, 1995] [Guyon et al., 1992] [Cortes and Vapnik, 1995], satisfies the following conditions:

$$\begin{cases} w^T \varphi(x_i) + b \geq +1 & \text{if } y_i = +1, \\ w^T \varphi(x_i) + b \leq -1 & \text{if } y_i = -1, \end{cases} \quad (5)$$

which is equivalent to

$$y_i [w^T \varphi(x_i) + b] \geq 1, \quad i = 1, 2, \dots, l. \quad (6)$$

Here training vectors x_i are mapped into a higher dimensional space by the function φ . The equations of (8) construct a hyper plane $w^T \varphi(x_i) + b = 0$ in this higher dimensional space that enables discrimination between the two classes (figure 3). Each of the two half-space defined by this hyper plane corresponds to one class, H_1 for $y_i = +1$ and H_2 for $y_i = -1$. Therefore the SVM classifier corresponds to the decision functions:

$$y(x) = \text{sign}[w^T \varphi(x_i) + b] \quad (7)$$

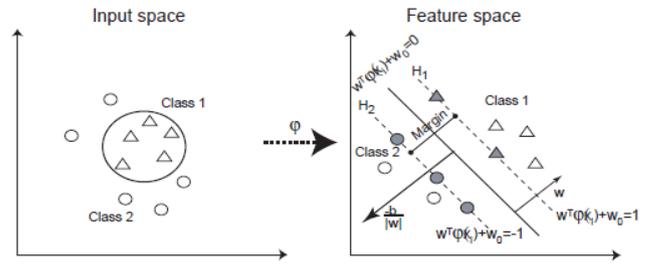


Fig.3. The mapping between input space and feature space in a two class problem with the SVM. Mapping the training data non-linearly into a higher dimensional feature space via function φ . H_1 and H_2 are parallel since they have the same normal w and perpendicular distance from the origin, $|\pm 1 - b|/\|w\|$, and that no training points fall between them. The support vectors are the gray triangles and circles respectively located on H_1 and H_2 . The distance from w to these support vectors is $1/\|w\|$ and the margin is simply $2/\|w\|$.

Thus the SVM finds a linear separating hyper plane with the maximal margin in this higher dimensional space. The margin of a linear classifier is the minimal distance of any training point to the hyper plane which is the distance between the dotted lines H_1 and H_2 and the solid line showed in figure 3. The points x which lie on the solid line satisfy $w^T \varphi(x) + b = 0$, where w is normal to the hyper plane, $|b|/\|w\|$ is the perpendicular distance from the hyper plane to the origin, and $\|w\|$ is the Euclidean norm of w . $1/\|w\|$ is the shortest distance from the separating hyper plane to the closest positive (negative) example. Therefore, the margin of a separating hyper plane will be $1/\|w\| + 1/\|v\|$. To calculate the optimal separating plane is equivalent to maximizing the separation margin or distance between the two dotted lines H_1 and H_2 .

$H_1 : w^T \varphi(x_i) + b = 1$ and $H_2 : w^T \varphi(x_i) + b = -1$ are parallel since they have the same normal w and perpendicular distance from the origin, $|1-b|/\|w\|$ for H_1 and $|-1-b|/\|w\|$ for H_2 , and that no training points fall between them. Thus we expect the solution for a typical two dimensional problem to have the form shown in figure 3. Those training points which give equality in (9) are lying on one of the hyper planes H_1 and H_2 and are called support vectors. They are indicated in figure 3 by gray color.

For non-separable classes, the optimization process needs to be modified in an efficient and elegant manner. In mathematical terms, the maximal margin hyper plane for non-separable data is selected by minimizing the cost function:

$$J(w, b, \xi) = \frac{1}{2} w^T w + C \sum_{i=1}^l \xi_i \quad (8)$$

subject to the constraints:

$$y_i(w^T \varphi(x_i) + b) \geq 1 - \xi_i, \quad i = 1, 2, \dots, l. \quad (9)$$

Where,

$$\xi_i \geq 0, \quad i = 1, 2, \dots, l.$$

where the variables ξ_i are known as slack variables. Note that, the goal of the optimization task is to make the margin as large as possible and reduce the number of points with $\xi_i > 0$. The parameter C is a positive constant that controls the relative influence of the two competing terms.

When no linear separation of the training data is possible, SVM can work in combination with kernel techniques so that the hyper plane defining the SVM corresponds to a nonlinear decision boundary in the input space. If the data is mapped to some other (possibly infinite dimensional)

Euclidean space using a mapping $\Phi(x)$, the training algorithm only depends on the data through dot products in such a Euclidean space, i.e. on functions of the form $\Phi(x) \cdot \Phi(x')$.

If a kernel function K is defined as:

$$K(x, x') = \Phi(x) \cdot \Phi(x') \quad (11)$$

then, it is not necessary to know the Φ function during the training process. In the test phase, an SVM is used by computing dot products of a given test point x with w , or more specifically by computing the sign of:

$$f(x) = \sum_{i=1}^l y_i \Phi(s_i) \cdot \Phi(x) + b = \sum_{i=1}^l y_i K(s_i, x) + b \quad (12)$$

where s_i are support vectors.

Figure 3 shows the basic idea of the SVM in which the use of kernels in SVM enables the mapping of the data into some other dot product space (called feature space) F via a nonlinear transformation.

$$\Phi: \mathbb{R}^N \rightarrow F \quad (13)$$

and perform the above linear algorithm in F . Note that, all the points belonging to a given class remain at a given side of the separating hyper plane and the data becomes linearly separable. In the input space, the hyper plane corresponds to a non-linear decision function whose form is determined by the kernel.

We now provide a description of the tools to construct nonlinear classifiers. We substitute $\Phi(x_i)$ for each training example x_i , and perform the optimal hyper plane algorithm in F . Since we are using kernels, we will end up with a nonlinear decision function of the form.

$$y(x) = \text{sign} \left[\sum_{i=1}^l y_i K(x, x') + b \right]. \quad (14)$$

There are many possibilities to define a function to be used as a Kernel. However, typical examples of kernels used in SVM, which have been successfully applied to a wide variety of applications, are linear, polynomial, radial basis functions and the hyperbolic tangent:

$$\text{Linear Kernel} : k(x, x') = x \cdot x' \quad (15)$$

$$\text{Polynomial Kernel} : k(x, x') = (x \cdot x' + c)^d \quad (16)$$

$$\text{RBF Kernel} : k(x, x') = \exp \left(-\frac{\gamma(x - x')^2}{\sigma} \right) \quad (17)$$

$$\text{Sigmoid Kernel} : k(x, x') = \tanh(\gamma(x \cdot x') + c) \quad (18)$$

Beside the possible kernels defined in this section there are others and much of the most current research is oriented to improve and to increase the efficiency of the SVM method. In subsections 4.3 and 4.4 the peculiarities of the kernels used in our experiments will be explained. These are the linear and the puk kernel.

IV. EXPERIMENTATION

4.1 Parkinson data

The Parkinson database used in this study is taken from the University of California at Irvine (UCI) machine learning repository [Asuncion and Newman, 2007] [Little et al., 2007] [Little et al., 2008]. It was used for training and testing experiments. The reason to use these sets of data is that the data sets of this website have been donated from hospitals. These data have been studied by many professionals of artificial intelligence departments. The dataset is composed of a range of biomedical voice measurements from 31 people, 23 with PD. Each column in table 1 is a particular voice measure, and each row corresponds to one of 195 voice recordings of these individuals ("name" column). Table 1 shows the fields of this database and a brief description of each input variable.

Table 1

List of measurement methods applied to acoustic signals recorded from each subject.

Field name	Description
name	ASCII subject name and recording number
MDVP:Fo(Hz)	Average vocal fundamental frequency
MDVP:Fhi(Hz)	Maximum vocal fundamental frequency
MDVP:Flo(Hz)	Minimum vocal fundamental frequency
MDVP:Jitter(%)	Five measures of variation in fundamental frequency
MDVP:Jitter(Abs)	
MDVP:RAP	
MDVP:PPQ	
Jitter:DDP	
MDVP:Shimmer	Six measures of variation in amplitude
MDVP:Shimmer(dB)	
Shimmer:APQ3	
Shimmer:APQ5	
MDVP:APQ	
Shimmer:DDA	
NHR	Two measures of ratio of noise to tonal components in the voice
HNR	
RPDE	Two nonlinear dynamical complexity measures
D2	
DFA	Signal fractal scaling exponent
spread1	Three nonlinear measures of fundamental frequency variation
spread2	
PPE	
Status	Output - Health of the subject (1) - Parkinson's, (0) - healthy

In this study we have used the Weka program package. The Weka program package is a JAVA software package from

the University of Waikato, New Zealand [Witten and Frank, 2005] issued under the GNU General Public License. This

software has been used and referenced in many works and projects [Massarelli et al., 2009] [Huang et al., 2009] [Ahmed et al., 2008] [Tari et al., 2008].

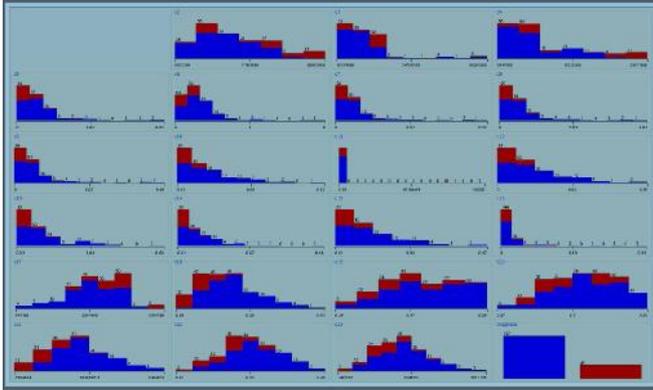


Fig.4. Relations input data and output (diagnosis). Every histogram in the above figure show the relation between each one of the input and output (the diagnosis) data. The goal is to provide a fast visual perception to appreciate the degree of influence between a specific input data, which has been measured by the specialists, and the final diagnosis.

The first step of the experimentation is to load the data. Figure 4 shows the relation between every input field and the diagnosis. That relation reflects just how small or big the influence between a specific input data and the final diagnosis is. The method to evaluate our system is to obtain some measures as classification accuracy, sensitivity, specificity, positive predictive value, negative predictive value and a confusion matrix. A confusion matrix [Kohavi and Provost, 1998] contains information about actual and predicted classifications done by a classification system.

4.2 MLP

In this section we test the Parkinson database by using MLP. For the construction of the architecture of the MLP we proceed as follows:

a) Layer 1 corresponds directly to the input vector, that is, all the parameters fields of the patient's record.
 b) Layer 2 (the hidden layer). The number of hidden neurons for this layer is the most elaborated question in the network's architecture. This number represents a trade of between performance and the risk of over fitting. In fact, the number of neurons in a hidden layer will significantly influence the ability of the network to generalize from the training data to the unknown examples [Pal and of Nottingham , GB]. By doing some experiments we discovered that:

- With a low number of neurons for this layer training and test sets performed badly.
- With a high number of neurons the training set performed well. However there is a high risk of over fitting.

- The optimal solution for this layer was found to be 13 neurons.

Therefore, the best solution for this hidden layer has been found with 13 neurons.

c) Layer 3 (the output layer) (ill and healthy patients). Table 2, 3 and 4 show the confusion matrix for a two class classifier. Classification accuracy, sensitivity, specificity, positive predictive value and negative predictive value can be defined by using the elements of the confusion matrix.

Table 2

Definition of the confusion matrix with the value for every measure for the MLP classifier:

Actual	Predicted	
	Positive	Negative
Positive	True positive (TP)=138	False negative (FN)=9
Negative	False positive (FP)=6	True negative (TN)=42

Classif:

$$\text{accuracy}(\%) = \frac{TP + TN}{TP + FP + FN + TN} \times 100 = 92.31\% \quad (19)$$

Sensitivity(%)

$$= \frac{TP}{TP + FN} \times 100 = 93.88\% \quad (20)$$

Specificity(%)

$$= \frac{TN}{FP + TN} \times 100 = 87.50\% \quad (21)$$

Positive predictive value(%)

$$= \frac{TP}{TP + FP} \times 100 = 95.83\% \quad (22)$$

Negative predictive value(%)

$$= \frac{TN}{FN + TN} \times 100 = 82.35\% \quad (23)$$

4.3 SVM with linear kernel

In the next two sections we do some experimentation with the Parkinson database by using the SVM with different kernels in order to test the accuracy. The SVM produces better results than the MLP tested in the previous section. In particular we use a new algorithm for training the SVM: Sequential Minimal Optimization (SMO) which is a faster

training method for SVMs. SVMs have empirically been shown to have good generalization performance on a wide variety of problems. However, the use of SVMs is still limited to a small group of researchers. One possible reason is that training algorithms for SVMs are slow, especially for large problems. Another explanation is that SVM training algorithms are complex, subtle, and difficult for an average engineer to implement. Training a SVM requires the solution of a very large Quadratic Programming (QP) optimization problem. SMO breaks this large QP problem into a series of smallest possible QP problems [Platt, 1998] [Platt, 1999] [Keerthi et al., 2001]. This implementation globally replaces all missing values and transforms nominal attributes into binary ones. It also normalizes all attributes by default.

Table 3

Definition of the confusion matrix with the values for every measure of the SVM classifier and linear kernel.

Actual	Predicted	
	Positive	Negative
Positive	True positive (TP)=146	False negative (FN)=1
Negative	False positive (FP)=15	True negative (TN)=33

Classif: accuracy(%)

$$= \frac{TP + TN}{TP + FP + FN + TN} \times 100 = 91.79\% \quad (24)$$

Sensitivity(%)

$$= \frac{TP}{TP + FN} \times 100 = 99.32\% \quad (25)$$

Specificity(%)

$$= \frac{TN}{FP + TN} \times 100 = 68.75\% \quad (26)$$

Positive predictive value(%)

$$= \frac{TP}{TP + FP} \times 100 = 90.68\% \quad (27)$$

Negative predictive value(%)

$$= \frac{TN}{FN + TN} \times 100 = 97.06\% \quad (28)$$

4.4 SVM with puk kernel

In this section we test the parkinson database with the SVM method by using an universal Pearson VII function based kernel (puk kernel) [ÄUstÄun et al., 2006]. This new method improves the accuracy of our system.

The applicability, suitability, performance and robustness of this alternative kernel in comparison to the commonly applied kernels is investigated by applying this to simulated as well as real-world data sets. From the outcome of these examinations, it was concluded that the PUK kernel is robust and has an equal or even stronger mapping power as compared to the standard kernel functions leading to an equal or better generalization performance of SVMs. In general, PUK can be used as a universal kernel that is able to serve as a generic alternative to the common linear, polynomial and RBF kernel functions [ÄUstÄun et al., 2007].

Table 4

Definition of the confusion matrix with the value for every measure of the SVM classifier and puk kernel.

Actual	Predicted	
	Positive	Negative
Positive	True positive (TP)=139	False negative (FN)=8
Negative	False positive (FP)=5	True negative (TN)=43

Classif: accuracy(%)

$$= \frac{TP + TN}{TP + FP + FN + TN} \times 100 = 93.33\% \quad (29)$$

Sensitivity(%)

$$= \frac{TP}{TP + FN} \times 100 = 94.56\% \quad (30)$$

Specificity(%)

$$= \frac{TN}{FP + TN} \times 100 = 89.58\% \quad (31)$$

Positive predictive value(%)

$$= \frac{TP}{TP + FP} \times 100 = 96.53\% \quad (32)$$

Negative predictive value(%)

$$= \frac{TN}{FN + TN} \times 100 = 84.31\% \quad (33)$$

One of the reasons why such a high degree of accuracy is obtained, is due to the data cleaning procedure; the data preprocessing. The data preprocessing of the databases collected directly from a hospital or Health centers, is necessary in order to homogenize the data before applying them to artificial intelligence methods. Moreover, it increases the accuracy of the classification methods.

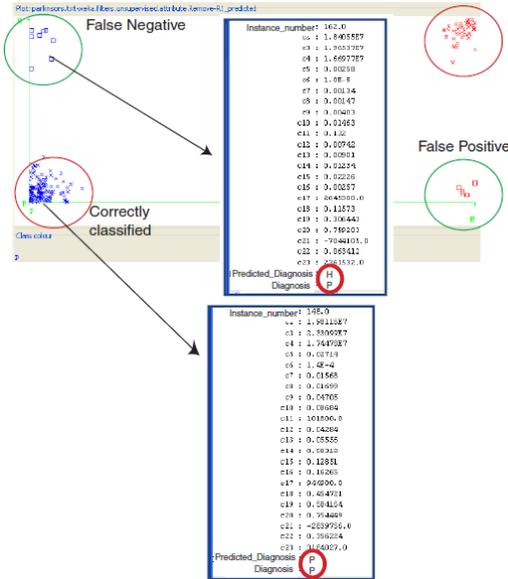


Fig. 5. The classifier errors. The two errors showed in the figure are the two instances of the Parkinson class. One of them is correctly classified (predicted Diagnosis and the real diagnosis coincide). However, the other one does not (predicted Diagnosis indicates H - Healthy whereas the real diagnosis is P - Parkinson). This is a false negative and the goal is to learn of these errors, try to find out what fields are implicating in that and to make the system robust.

By using these three types of tools, not only will it be possible to make comparisons between them to see which present the higher precision but also to complement them. In particular, MLP has a high value of "Positive predictive value" equal to 95.83%, SVM with linear kernel has the highest value of "Sensitivity" equal to 99.32% and "Negative predictive value" equal to 97.06%. Finally, SVM with kernel puk presents the highest values of "Classification of accuracy" equal to 93.33% and "Positive predictive value" equal to 96.53%.

V. CONCLUSIONS AND FUTURE WORK

In this paper we have evaluated the performance of a classifier constructed by means of ANN and SVM. The results presented by these three methods (MLP and SVM with the two kernel types) have both a high precision level of the confusion matrix regarding the different measurement parameters (accuracy, sensitivity, specificity, positive predictive value and negative predictive value).

The accuracy of the ANN and SVM were very good. They showed a high degree of certainty, above 90%. Furthermore, some of the parameters reach very high accuracy such as "Sensitivity" and "Negative predictive value" with 99.32% and 97.06% respectively.

Consequently, we propose a hybrid system combining ANN and SVM classifiers (SVM is tested with two different kernels). The goal is not only to establish a comparison between all of them but also to benefit from the highest accuracies of each classifier. The diagnosis must be reinforced and complemented in order to provide a better generalization in the same way that two or more specialists (or a specialist group) co-operate with each their methods, in order to obtain a final common diagnosis. As illustrated in Figure 5, our system allows finding out which instances are correctly or incorrectly classified. A future line of the system is an exhaustive study of all the fields, thus allowing us to determine why the errors occurred, and learning how to avoid this from happening in the future.

We have found that the outliers and the imbalanced data directly affected the classification performance and effectiveness of the classifiers. There are 147 registers with PD and 48 healthy ones. The accuracy of the classifiers will be improved by eliminating a number of outliers from both the minority and majority classes, and increasing the size of the minority class to the same size of the majority class.

Once the AI methods have been separately and/or individually tested, the next step will be to use a clustering method also called a metalearning. Metalearning algorithms take classifiers and turn them into more powerful learners with a higher generalization degree. They carry out the classifications either by averaging probability estimation or by voting and they always take advantage of every particular method.

VI. ACKNOWLEDGMENT

We want to express our acknowledgements to Max Little of the University of Oxford, who has created the database, in collaboration with the National Centre for Voice and Speech, Denver, Colorado, who recorded the speech signals. The original study published the feature extraction methods for general voice disorders.

REFERENCES

- B.A. Ahmed, M.E. Matheny, P.L. Rice, J.R. Clarke, and O.I. Ogunyemi. A comparison of methods for assessing penetrating trauma on retrospective multi-center data. *Journal of Biomedical Informatics*, 2008.
- M.F. Akay. Support vector machines combined with feature selection for breast cancer diagnosis. *Expert Systems With Applications*, 2008.
- A. Asuncion and D.J. Newman. UCI machine learning repository, 2007. URL <http://www.ics.uci.edu/~mllearn/{MLR}epository.html>.

- S. Berdia and JT Metz. An artificial neural network stimulating performance of normal subjects and schizophrenics on the Wisconsin Card Sorting Test. *Artificial Intelligence in Medicine*, 13(1-2):123{138, 1998.
- CM Bishop. 1995, *Neural Networks for Pattern Recognition*, Oxford: Oxford University Press.
- P. Björne and C. Balkenius. A model of attentional impairments in autism: first steps toward a computational theory. *Cognitive Systems Research*, 6: 193{204, 2005.
- C.J.C. Burges. A Tutorial on Support Vector Machines for Pattern Recognition. *Data Mining and Knowledge Discovery*, 2(2):121{167, 1998.
- I. Cohen. Neural network analysis of learning in autism. *Neural Networks and Psychopathology: Connectionist Models in Practice and Research*, page 274, 1998.
- IL Cohen. An artificial neural network analogue of learning in autism. *Biol Psychiatry*, 36(1):5{20, 1994.
- C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20(3): 273{297, 1995.
- A. Elbaz, J.H. Bower, D.M. Maraganore, S.K. McDonnell, B.J. Peterson, J.E. Ahlskog, D.J. Schaid, and W.A. Rocca. Risk tables for parkinsonism and Parkinson's disease. *Journal of clinical epidemiology*, 55(1):25{31, 2002.
- I. Guyon, B. Boser, and VN Vapnik. A training algorithm for optimal margin classifiers. In *Proc. of the 5th annual workshop of computational learning theory*, ACM, pages 144{152, 1992.
- S. Haykin. *Neural Networks: A Comprehensive Foundation*, Englewoods Cliffs, 1998.
- C.W. Hsu, C.C. Chang, C.J. Lin, et al. A practical guide to support vector classification, 2003.
- S.H. Huang, L.R. Wulsin, H. Li, and J. Guo. Dimensionality reduction for knowledge discovery in medical claims database: Application to antidepressant medication utilization study. *Computer Methods and Programs in Biomedicine*, 93(2):115{123, 2009.
- AJ Hughes, SE Daniel, L. Kilford, and AJ Lees. Accuracy of clinical diagnosis of idiopathic Parkinson's disease: a clinico-pathological study of 100 cases. *British Medical Journal*, 55(3):181{184, 1992.
- GA Ivanitsky and RA Naumov. Recognition of ongoing mental activity with artificial neural network. *International Journal of Psychophysiology*, 69(3): 180{180, 2008.
- J. Jankovic, A.H. Rajput, M.P. McDermott, and D.P. Perl. The evolution of diagnosis in early Parkinson disease, 2000.
- SS Keerthi, SK Shevade, C. Bhattacharyya, and KRK Murthy. Improvements to Platt's SMO algorithm for SVM classifier design. *Neural Computation*, 13(3):637{649, 2001.
- R. Kohavi and F. Provost. Glossary of terms. *Machine Learning*, 30(2/3): 271{274, 1998.
- M.A. Little, P.E. McSharry, S.J. Roberts, D.A.E. Costello, and I.M. Moroz. Exploiting Nonlinear recurrence and Fractal scaling properties for voice disorder detection. *BioMedical Engineering OnLine*, 6(1):23, 2007.
- M.A. Little, P.E. McSharry, E.J. Hunter, J. Spielman, and L.O. Ramig. Suitability of dysphonia measurements for telemonitoring of Parkinson's disease. *IEEE transactions on bio-medical engineering*, 2008.
- C. Loukas and P. Brown. Online prediction of self-paced hand-movements from subthalamic activity using neural networks in Parkinson's disease. *Journal of neuroscience methods*, 137(2):193{205, 2004.
- I. Massarelli, M. Imbriani, A. Coi, M. Saraceno, N. Carli, and A.M. Bianucci. Development of QSAR models for predicting hepatocarcinogenic toxicity of chemicals. *European Journal of Medicinal Chemistry*, 2009.
- M. Pal and University of Nottingham (GB). *Factors Influencing the Accuracy of Remote Sensing Classification: A Comparative Study*. University of Nottingham, 2002.
- P. Piccini and A. Whone. Functional brain imaging in the differential diagnosis of Parkinson's disease. *Lancet Neurology*, 3(5):284{290, 2004.
- J. Platt. Machines using sequential minimal optimization. *Advances in Kernel Methods-Support Vector Learning*, 1998.
- J. Platt. Sequential minimal optimization: A fast algorithm for training support vector machines. *Advances in Kernel Methods-Support Vector Learning*, 208, 1999.
- AH Rajput, B. Rozdilsky, and A. Rajput. Accuracy of clinical diagnosis in parkinsonism{a prospective study. *The Canadian journal of neurological sciences. Le journal canadien des sciences neurologiques*, 18(3):275, 1991.
- B.D. Ripley. *Pattern recognition and neural networks*. Cambridge university press, 1996.
- D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-propagating errors. *Nature*, 323:533{536, 1986.
- L. Tari, C. Baral, and S. Kim. Fuzzy c-means clustering with prior biological knowledge. *Journal of Biomedical Informatics*, 2008.
- S. Theodoridis and K. Koutroumbas. *Pattern Recognition (2nd)*, 2003.
- E. Tolosa, G. Wenning, and W. Poewe. The diagnosis of Parkinson's disease. *Lancet Neurology*, 5(1):75{86, 2006.
- B. Åstun, WJ Melssen, and LMC Buydens. Facilitating the application of Support Vector Regression by using a universal Pearson VII function based kernel. *Chemometrics and Intelligent Laboratory Systems*, 81(1):29{40, 2006.
- B. Åstun, WJ Melssen, and LMC Buydens. Visualisation and interpretation of support vector regression models. *Analytica Chimica Acta*, 595(1-2):299{309, 2007.
- V.N. Vapnik. *The Nature of Statistical Learning Theory [M]*, 1995.
- I.H. Witten and E. Frank. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, San Francisco, 2005.

Secured Data Comparison in Bioinformatics using Homomorphic Encryption Scheme

Gorti VNKV Subba Rao,
Associate Professor, CSE Dept,
Nalla Malla Reddy Engineering College, Divya Nagar – Hyderabad. 9440986317,
gvnkvsubarao@yahoo.com

Abstract- The databanks available globally having the diagnosis results of patients about chronic diseases like Diabetics, Blood Pressure etc offer wealth information that reveal among others, predisposition to various diseases and paternity relations. But these data requires privacy as some times the parties may not interest to reveal the data to each other. They need a protocol or the technique to preserve the privacy. Here we are implementing the homomorphic encryption scheme to compare the cipher text as plaintext so that the privacy may be preserved. Here we are discussing the new protocol for the existing schemes. We also compare the computational costs of the some homomorphic encryption schemes.

Keywords

Homomorphic encryption, electronic medical records (EMR), Mixed Multiplicative Homomorphism (MMH), Privacy Preserving Hamming distance, Edit distance problem.

I. INTRODUCTION

Over the past decade, there has been a growing need for large -scale privacy preserving systems spanning several databases distributed over the Internet [23]. One motivating example is the nation-wide electronic medical records (EMR) effort within the US which hopes to integrate the EMR of patients across a large number of hospitals while mandating stringent privacy requirements for patient records as specified in the HIPAA regulations [1]. Over the years, the research community has developed a wide range of privacy-preserving techniques for answering different types of queries [2, 3, 4, and 13] without revealing information of any individual database which is irrelevant to the queries.

Human Desoxyribo-Nucleic Acid (DNA) sequences offer a wealth of information that reveal among others predisposition to various diseases and paternity relations. The breadth and personalized nature of this information highlights the need for privacy –preserving protocols. The human genome contains a wealth of information about a person's body broad access to the genome is likely to revolutionize medical diagnosis and treatment [23]. A doctor can avail this stored data and find out whether the patient has predisposition towards developing a specific diseases like diabetics, blood pressure etc. and also he can understand that the patient reactions towards the specific

drug composition [23]. Or whether the treatment will likely fail, there by reducing the overall costs and increasing the effectiveness of the therapy. Finally, it may be possible to create an individual drug therapy for each patient by analyzing his genetic profile and predicting his response to different medications [22]. .If the data is available in internet are with the private organizations then if a patient has digital or image data about his DNA are genome then he wants to give it to another party for diagnosis then obviously both parties wants to maintains secrecy as Patient feels My symptoms and history are personal Diagnose.com feels My diagnostic is proprietary and valuable [23].

When the diagnostic and the data are both private then we need a procedure to compare both and results will be forward to patient. There are some disadvantages with this kind of data available on internet or with the private databanks .For instance a person carrying the cancer symptoms may not get insurance coverage which will be rejected at starting itself by insurance companies. In other scenario an employee may be rejected from his permanent job work because of his history from the databanks. Privacy concerns about this information have traditionally been addressed through laws and procedures .Healthcare professionals are required to keep sensitive data confidential and make it available only with explicit consent of the patient .so far this kind of traditional approach has worked reasonably well, due to limited data availability at established centers. This kind of traditional form of protecting the sensitive information leakage is insufficient. We may understand that the cryptographic privacy preserving protocols will become invaluable components to over come the procedural approach.

One of the fundamental methods for molecular sequence comparison and alignment is the Needleman-Wunsch algorithm [6], which is used in software for detecting similarities between two DNA sequences. The underlying sequence comparison and alignment problem is also known as the string edit problem in the literature.

II. RELATED LITERATURE SURVEY

The problem discussed here is without revealing the data of the both parties providing the result for further actions to the doctors and diagnosis centers. here we used the homomorphic property to provide security as the computations takes place on encrypted data. Here we are going to discuss some protocols edit distance, Hamming distance for finding equalities in encrypted data given by some research scholars and also we are providing a protocol using the various homomorphic encryption schemes such as Elgamal, Elliptic curve and privacy homomorphism.

They given an efficient protocol for sequence comparisons of the edit distance kind, such that neither party reveals anything about their private sequence to the other party (other than what can be inferred from the edit distance between their two sequences {which is unavoidable because computing that distance is the purpose of the protocol). The amount of communication done by our protocol is proportional to the time complexity of the best-known algorithm for performing the sequence comparison The problem of determining the similarity between two sequences arises in a large number of applications, particularly in bioinformatics. In these application areas, the edit distance is one of the most widely used notions of sequence similarity: It is the least-cost set of insertions, deletions, and substitutions required to transform one string into the other. The generalizations of edit distance that are solved by the same kind of dynamic programming recurrence relation as the one for edit distance, cover an even wider domain of applications.

III. BUILDING BLOCKS

Homomorphic encryption

Homomorphic encryption is a special type of cryptography in which the sum of two encrypted values is equal to the encrypted sum of values [21]. The high level formula for this is. In simple mathematics, this is equivalent to the communicative property of multiplication, where. For a majority of cryptographic algorithms, this does not hold true. In most cases, it is undesirable because it may help reveal information which can be used to break the encryption. However, this is a desirable property if one wishes to have the sum of a group of encrypted values verified without revealing those encrypted values. In voting protocols, this is used to verify the tally of the ballots without revealing what those ballots are.

Scenario 1

Alice wants to query a database but she does not want the database to learn what she is querying [21].

Alice has a function that she does not want to reveal and some secret inputs. Bob has some secret inputs and wants to compare with Alice are they want to compare with data base banks for getting the information without revealing data to others.

- **Homomorphic Encryption**

Scheme (HES) enables direct computation on encrypted data without decryption. Properties of HES that are needed [10, 11]:

- **Additively homomorphic:**

Computing $E(x+y)$ from $E(x)$ and $E(y)$ without revealing x and y

- **Multiplicatively homomorphic:**

computing $E(xy)$ from $E(x)$ and $E(y)$ without revealing x and y .

- **Mixed-multiplicatively**

Homomorphic: computing $E(xy)$ from $E(x)$ and y without revealing x .

IV. MIXED MULTIPLICATIVE

Homomorphism

The mixed multiplicative homomorphism [5] is the simplified version of Domingo-Ferrer's field encryption [4].

1) *Encryption/Decryption*

This new cryptosystem, uses large number m , where $m = p * q$. Here p and q are large prime numbers, which are kept secret. Let $Z_p = \{ x/x <= p \}$ be the set of original plaintext messages, $Z_m = \{ x/x < m \}$ be the set of ciphertext messages and $Q_p = \{ a/a \notin Z_p \}$ be the set of encryption clues.

The *encryption operation* is performed by choosing a plaintext 'x' belonging to Z_p and a random number 'a' in Q_p such that $x = a \bmod p$, here p is kept secret. The cipher text y is calculated as $y = E_p(x) = a \bmod m$.

In *decryption operation* the plaintext x is recovered as $x = D_p(y) = y \bmod p$, where p is the secret key.

2) *Properties*

Following are the properties of the mixed multiplicative homomorphism:

- **Additive Homomorphic:**

Addition is done on cipher text and the decrypted result is same as the sum of plaintexts.

- **Multiplicative Homomorphic:**

Multiplication is done on cipher text and the decrypted result is same as the product of plaintexts.

- **Mixed-Multiplicative**

Homomorphic: Multiplication is done on a cipher text and Plaintext and the decrypted result is same as the product of plaintexts.

Let us look into an example, which explains this protocol in more detail.

Let $p = 13$, $q = 5$, $n = p * q = 65$, $x_1 = 3$, $E_p(x_1) = 29$, $x_2 = 7$ and $E_p(x_2) = 46$.

Additive homomorphism: In additive homomorphism decryption of the sum of two cipher text is same as the addition of two plaintext.

$$E_p(x_1) + E_p(x_2) = 29 + 46 = 75 \text{ mod } 65 = 10$$

By decrypting the cipher text 10 we get, $D_p(10) = 10 \text{ mod } 13 = 10$, which is same as the plaintext $x_1 + x_2 = 10$. This shows that addition can be performed on the cipher text, as if performed on the plaintext.

Multiplicative homomorphism: In multiplicative homomorphism decrypted result of the product of two cipher text is same as the multiplication of two plaintext.

$$E_p(x_1) * E_p(x_2) = 29 * 46 = 1334 \text{ mod } 65 = 34.$$

By decrypting the ciphertext 34 we get, $D_p(34) = 34 \text{ mod } 13 = 8$, which is same as the plaintext $x_1 * x_2 = 21 \text{ mod } 13 = 8$. This shows that multiplication can be performed on the cipher text, as if performed on the plaintext.

Mixed Multiplicative Homomorphism: In Mixed Multiplicative Homomorphism $E_p(x_1 * x_2) = E_p(x_1) * x_2$

$$E_p(x_1) * x_2 = 29 * 7 = 203 \text{ mod } 65 = 8.$$

By decrypting 8 we get, $D_p(8) = 8 \text{ mod } 13 = 8$, which is same as the plaintext $x_1 * x_2 = 21 \text{ mod } 13 = 8$. This shows that mixed multiplicative homomorphism can be performed on the ciphertext, as if performed on the plaintext.

3) Security

The proposed protocol though exhibits the property of homomorphism is not very secure against known plain-text attack. The cryptosystem is however secured against known cipher text attack.

Let us look into the known plaintext attack and known cipher text attack in more detail with respect to this cryptosystem.

• Known plaintext attack:

If (x, y) are the known plaintext ciphertext pair, finding key p is relatively easy. We know $x = D_p(y) = y \text{ mod } p$, where p is the prime key and $p > x$. We know that $p \mid (y - x)$, that is p completely divides $y - x$. Here x and y are known, so finding p is relatively easy as p is the divisor of $(y - x)$.

• Known ciphertext attack:

We know $x = y \text{ mod } p$ and if y is known, x and p are still unknown. We know, $y = x + rp$ and it is difficult to determine y as x and p are unknown.

V. EXPERIMENT PROCEDURE AND RESULTS

The problem can be described as follows:

The client has an algorithm to compute a function $f(x)$.

The client sends it over to the server which computes the function on an input x . x can be encrypted form of client's data stored by the server. Therefore the function f should have the capability to decode it. The server should not be

able to learn anything substantial about f or the intermediate data it uses for execution.

A simple protocol for doing this will look like

1. Client encrypts f .
2. Client creates a program $P(E(f))$ which implements $E(f)$.
3. Client sends $P(E(f))$ to server.
4. Server executes $P(E(f))$ with x as input.
5. Server sends $P(E(f))(x)$ to client.

Alice or doctor wants report of his or his patient diagnosis data/image of his genome or DNA for understanding about his future medical suggestions so he will send the data to the database bank or distributed server or Bob/another doctor through the application .the application itself encrypts and sends data for result .At other side this data will be compared with the available data in banks and sends the report without revealing the data to each other. The application is simulated in c under Linux platform and the computational costs and information rates in encryption are compared and the graphical information has been furnished here.

The advantages over the existing ones: The first is that we lower the level of trust on the involved individual principals. The second is that extra attention has been paid to the privacy issues related to the sensitive relationship between a biometric feature and the relevant identities. Specifically, this relationship is unknown to the database and the matcher.

The central idea is to develop a system on which we can evaluate encrypted functions without decrypting it. The functions can be encrypted such that the resulting transformation can be implemented as a program that can be executed on an unreliable host. The executing computer will not be able to see the program's unencrypted instructions but will not be able to make out what the function implements.

The important implication here is that the executing system won't be able to modify the encrypted function in goal – oriented way. We will be looking at non interactive evaluation of encrypted functions, where in, we encrypt a function such that it still remains executable. The first observation we can make is that it is not enough to secure only the secret function or secret data, the whole program has to be made secure.

Otherwise an attacker can modify the clear text parts of the program and make the secure parts do something other than what it is meant to do.

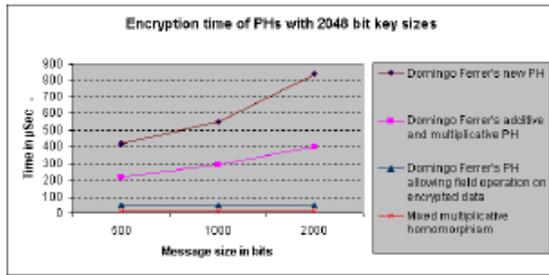


Figure: 1 Execution time of PHs in μSec with 2048 bit key size

Table: 1 Execution time of PHs in μSec with 2048 bit key size

MESSAGE SIZE IN BITS	500	1000	2000
DF's new PH with d=2	418	550	807
DF's add and mul PH with d=2	218	294	400
DF's field PH	48	48	48
MMH	10	0	11

Table: 1 Execution time of PHs in μSec with 2048 bit key size

From above Figure 1 and respective Table it is clear that MMH is much faster than DF's new Privacy Homomorphic encryption scheme, DF's additive and multiplicative Privacy Homomorphic encryption scheme and DF's field Privacy Homomorphic encryption scheme. We also see from Figure 1 that the encryption timing of DF's new Privacy Homomorphic encryption scheme, DF's additive and multiplicative Privacy Homomorphic encryption scheme and DF's field Privacy Homomorphic encryption scheme increases with the increase in encryption keys but the encryption timing of MMH remains almost the same with the increase in the encryption key size. From Figures 2, 3 and 4 and corresponding Tables we also see that the encryption timing of DF's new PH and DF's additive and multiplicative PH increases with the increase in message size. However the encryption timing of DF's field Privacy Homomorphic encryption scheme and MMH remains almost the same with the increase in the message size. In determining the encryption timing of DF's new Privacy Homomorphic encryption scheme and DF's additive and

multiplicative Privacy Homomorphic encryption scheme, the encryption split (d) is fixed to the value 2.

VI. CONCLUSION

Hence, the most important points those make our protocol more appropriate for biometrics authentication protocols are the following. Firstly, no secret information storage is required at the client side. Secondly, the protocol guarantees the privacy of the relationship between the user's identity and its biometric data, and the privacy of the user's biometric information. We considered a biometric authentication protocol where confidentiality is required for biometric data solely for privacy reasons. We captured these notions into a security model and introduced a protocol which is proved secure in this security model. It remains an interesting issue to improve its performance.

VII. ACKNOWLEDGMENT

I sincerely thanks to my supervisor Dr.G.Uma for her continuous support in carrying out this research work. I also thank the principal Dr.Divya of Nalla Malla Reddy Engineering College, Hyderabad for giving permission to conduct tests in the research Lab.

REFERENCES

- [1] Centers for Medicare & Medicaid Services. The Health Insurance Portability and Accountability Act of 1996 (HIPAA). <http://www.cms.hhs.gov/hipaaGenInfo>.
- [2] R. Agrawal, D. Asonov, M. Kantarcioglu, and Y. Li. Sovereign Joins. In *ICDE 2006*, page 26. IEEE Computer Society, 2006.
- [3] F. Emekci, D. Agrawal, A. E. Abbadi, and A. Gulbeden. Privacy Preserving Query Processing Using Third Parties. In *ICDE 2006*, page 27. IEEE Computer Society, 2006.
- [4] M. Naor, B. Pinkas, and R. Sumner. Privacy Preserving Auctions and Mechanism Design. In *Electronic Commerce 1999*, pages 129–139. ACM, 1999.
- [5]. A. C. Yao. Protocols for secure computations. In *Proceedings of the 23rd Annual IEEE Symposium on Foundations of Computer Science*, 1982.
- [6] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, 1978.
- [7] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Transactions on Information Theory*, vol. 31, no. 4, pp. 469–472, 1986.
- [8] T. Sander and C. Tschudin. Towards mobile cryptography. Technical report, International Computer Science Institute, Berkeley, 1997.
- [9] T. Sander and C. Tschudin. On software protection via function hiding. In *Information Hiding*, pages 111–123, 1998.
- [10] T. Sander and C. Tschudin. Protecting Mobile Agents Against Malicious Hosts. In G. Vigna, editor, *Mobile Agent Security*, pages 44–60. Springer-Verlag: Heidelberg, Germany, 1998.

- [11] H. Lee. *Mobile Agent: Evaluating Encrypted Functions*. PhD thesis, Department of Computer Science, University of Idaho, August 2002.
- [12] D. Naccache and J. Stern. A new public-key cryptosystem. In *Theory and Application of Cryptographic Techniques*, pages 27–36, 1997.
- [13] Y. Sakurai, M. Yokoo, and K. Kamei. An efficient approximate algorithm for winner determination in combinatorial auctions. In *Proceedings of the Second ACM Conference on Electronic Commerce (EC-00)*, pages 30–37, 2000.
- [14] T. Sandholm. An algorithm for optimal winner determination in combinatorial auction. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI-99)*, pages 542–547, 1999.
- [15] M. H. Rothkopf, A. Pekec, and R. M. Harstad. Computationally manageable combinatorial auctions. *Management Science*, 44(8):1131–1147, 1998.
- [16] Y. Fujishima, K. Leyton-Brown, and Y. Shoham. Taming the computation complexity of combinatorial auctions: Optimal and approximate approaches. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI-99)*, pages 548–553, 1999.
- [17] M. Tennenholtz. Some tractable combinatorial auctions. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence (AAAI-2000)*, pages 98–103, 2000.
- [18] R. Bellman. *Dynamic Programming*. Princeton University Press, Princeton, NJ, 1957.
- [12] S. B. Needleman and C.D. Wunsch. A General Method Applicable to the Search for Similarities in the Aminoacid Sequence of Two Proteins, *Journal of Molecular Biology* 48, pp.443{453 (1973).
- [19]. B. Schoenmakers and P. Tuyls. Efficient binary conversion for Paillier encrypted values. In S. Vaudenay, editor, EUROCRYPT, volume 4004 of Lecture Notes in Computer Science, pages 522–537. Springer, 2006.
- [20] Gorti V N K V SubbaRao, “Application of HES in Nueral networks”, presented in CCSB-09 conducted by Andhra University.
- [21] Gorti V N K V SubbaRao, “Application of HES in various Fields”, vol 1 PP 5 SSSSS-08 conducted by chaitanya Engg college,Vizag.
- [22] Gorti V N K V SubbaRao, “Application of HES in WSN”, presented in CCSB-09 conducted by Andhra University
- [23] Allam Appa Rao ,Gorti V N K V SubbaRao, , Manga Tayaru.N “Analysis of Resistin Protien involved in diabetes associated with obesity using homomorphic Encryption”, published in IJCEIT, April, 2009.

Performance Evaluation of Message Encryption Scheme Using Cheating Text

Ch.Rupa,
Student, (Ph.D),
rupamtech@gmail.com

P.S.Avadhani,
Professor,
psavadhani@yahoo.co.in

Phone No: +91 {9848690640, 9885187815}
Department of Computer Science and Systems Engineering
Andhra University
Visakhapatnam, Andhra Pradesh, India

Abstract- In this paper we evaluate the performance of our proposed Message Encryption scheme using cheating text [6] with respect to response time and user load. In this method the authentication of the message is also possible because of the hashing by Modified message digest Algorithm [6]. This scheme can be applied for authentication like security in data bases. The response time of the algorithm is implemented in java is calculated with user load and the different sizes of the data file. The results are also compared with the performance evaluation done by Priya Dhawan [8], Aamer Nadeem [7] and found to be efficient.

I. INTRODUCTION

The popular method of authentication is by using a hash algorithm. MD5 is a hash algorithm to prepare a message digest for a given plaintext. However, this suffers from Wang's collision attack [2]. Modified Message Digest algorithm is modified to sustain the Wang's collision attack. The idea is to use 64-bit chaining variables instead of 32-bit. The performance of our MMD encryption scheme is evaluated and compared with the Priya Dhawan [8], Aamer Nadeem [7]. The results are shown in section 3 and section 4.

Keywords: Junit, MMD, Wang's Collision.

II. AUTHENTICATION SYSTEM EVALUATION

1) Performance Evolution Methodology

This section describes the techniques and simulation choices made to evaluate the performance of the compared algorithms. In addition to that, this section will discuss the methodology related parameters like: system parameters, experiment factors, and experiment initial settings. It includes an overview of Junit test which is used for finding the response time of the algorithms. The experiments are conducted using 64-bit processor with 1GB of RAM [6]. The simulation program is compiled using the default settings in Java, windows applications. The experiments are performed number of times to assure that the results are consistent and are valid to compare the different algorithms.

In order to evaluate the performance of the compared algorithms, the parameters that the algorithms must be tested for, must be determined since the security features of each algorithm and their strength against cryptographic attacks is already known [2]. The chosen factor here to determine the performance is the algorithm's speed to find the hash value with the proposed algorithm of the data blocks of various sizes. The response time of the algorithm is implemented in java is calculated with user load and the different sizes of the data file. By considering different sizes of data blocks (4KB,135KB) the algorithms were evaluated in terms of the time required to hash functions using modified message digest algorithm and All the implementations were exact to make sure that the results will be relatively fair and accurate.

III. TESTING FRAME WORK

J-Unit is a unit-testing framework that can be used to test the functionality of java classes [11]. Writing a test is a method that exercises the code to be tested and defining the expected result. The framework provides the context for running the test automatically and as part of a collection of other tests. This investment in testing will continue to pay us back in time and quality.

a. Need of JUnit

1. Junit is useful to write tests that exercise our code and incrementally add tests as the software grows.
2. Testing become tough if we have to manually compare the expected and actual result of tests, and it slows us down. JUnit tests can be run automatically and they check their own results. When we run tests, we get simple and immediate visual feedback as to whether the tests passed or failed. There's no need to manually comb through a report of test results.

b. Features of JUnit

1. For testing java classes written by us we need to write test classes using features of JUnit.
2. **TestCase** is a command Object. Test Classes must extend this **TestCase** class. A **TestCase** can define any number

of public `testXXX()` methods. When we want to check the expected and actual test results, we can invoke a variation of the `assert()` method.

3. TestCase subclasses that contain multiple `testXXX()` methods can use the `setUp()` and `tearDown()` methods to initialize and release any common objects under test, referred to as the test fixture. Each test runs in the context of its own fixture, calling `setUp()` before and `tearDown()` after each test method to ensure there can be no side effects among test runs.

4. TestCase instances can be composed into TestSuite hierarchies that automatically invoke all the `testXXX()` methods defined in each TestCase instance. A TestSuite is a composite of other tests, either TestCase instances or other TestSuite instances. The composite behavior exhibited by the TestSuite allows you to assemble test suites of test suites of tests, to an arbitrary depth, and run all the tests automatically and uniformly to yield a single pass or fail status.

IV. RELATED WORK

In this section discusses the results obtained from Priya Dhawan [8], Aamer Nadeem [7]. By considering the response time, complexity and security will improve with the Advanced Authentication Method. The following graph is taken into consideration by considering different User load. Dhawaan [8], Nadeem [7] has done experiments for finding the performance of MD5. The method computes the hash of data stored in a file. It performed the tests with a data size of 4 KB, 135 KB to see how the size of data impacts performance. The algorithms were implemented in a Java, using their standard specifications, and were tested to compare their performance.

Fig 1 and 2 gives the Existing experiments report for performance of 4 KB and 135KB data files with respect of response time.

Fig 3 and 4 gives the Modified Message digest Algorithm reports for performance of 4 KB and 135KB data files with respect of response time.

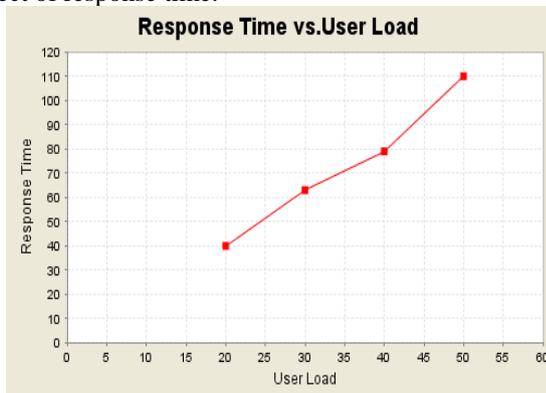


Fig 1. Graph for 4 KB data File

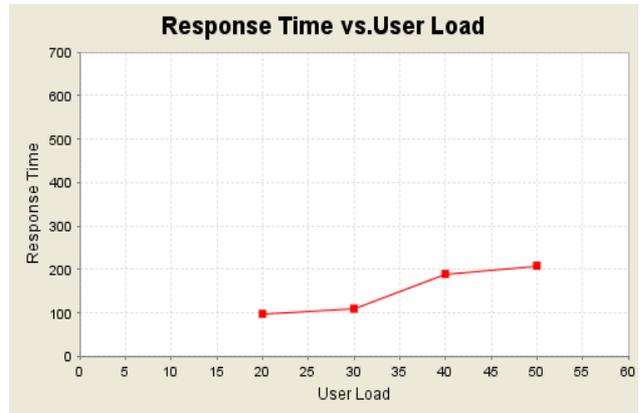


Fig 2. Graph for 135KB data File

V. PERFORMANCE METHODOLOGY ADOPTED FOR THE PROPOSED SCHEME

In this section a step by step procedure is given to illustrate the methodology adopted for the performance evaluation of the message encryption scheme proposed in [3,6].

Step 1: Developed unit test cases for the algorithm using JUnit Testing tool.

Step 2: Developed wrapper test cases using JUnitPerf testing tool for the test cases that were developed using JUnit as mentioned in step2.

Step 3: Using Load method of JUnitPerf, set the maximum number of concurrent users to execute Modified Message Digest [6] functionality simultaneously.

Step 4: Change in the response time can be observed with the change of user load (max number of concurrent users) with respect of executing the above steps.

Modified Message Digest algorithm [3,6] is getting the results in 'milli Seconds'. Table 1 and Table 2 contains the speed benchmarks for the 4KB data file using proposed algorithm and Existing Algorithm. Table 3 and Table 4 contains the speed benchmarks for the 135KB data file using proposed algorithm [3,6] and Existing Algorithm [2,5]. Existing Algorithm [2] is having inconsistency as shown in Fig 3 along with the collision attacks [2]. Proposed algorithm [3,6] overcome the limitations what are existed.

Load	Time Taken(ms)
20	0.015
30	0.032
40	0.062
50	0.078
60	0.109

Table 1 Response time for 4 KB data file with respect of user load (in milli seconds)

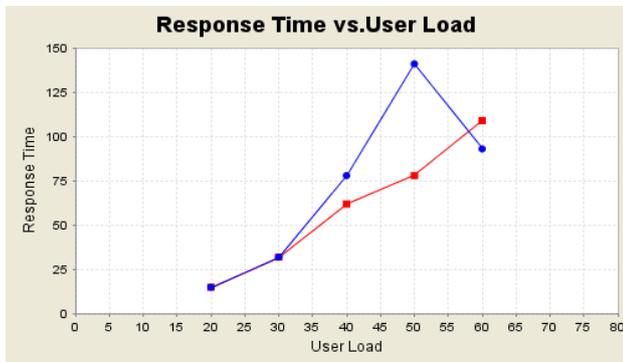
Load	Time Taken(ms)
10	0.016
20	0.015
30	0.032
40	0.078
50	0.141
60	0.093
70	0.203

Table 3 Response time for 135 KB data file with respect of user load (in milli seconds)

Load	Time Taken(ms)
10	0.016
20	0.015
30	0.032
40	0.078
50	0.141
60	0.093
70	0.203

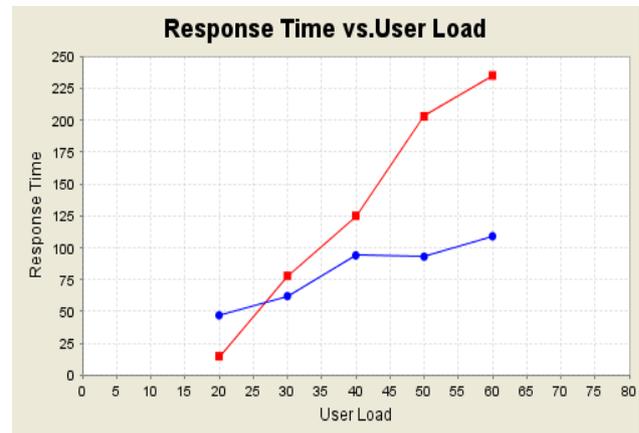
Table 2 Response time for 4 KB data file with respect of user load using Existing algorithm.

Table 4 Response time for 135 KB data file with respect of user load using Existing algo



.....Existing Algorithm(MD5)
Proposed Algorithm(MMD5)

Fig 3 Comparison results for 4 KB files with the existing algorithm.



...Existing Algorithm(MD5)
Proposed Algorithm(MMD5)

Fig 4. Comparison results for 4 KB files with the existing algorithm.

Load	Time Taken(ms)
20	0.015
30	0.078
40	0.125
50	0.203
60	0.235

VI. CONCLUSION

The presented testing results showed that MMD has a better performance . As these tests demonstrate, authentication

schemes and hashing algorithms carry varying amounts of overhead, and therefore have vastly different performance characteristics. The size of data being passed to hashing algorithms, as well to cryptography techniques, is also significant.

VII. ACKNOWLEDEMENT

I thank full to Prof P.S. Avadhani and Prof Nagalakshmi for his esteemed guidance and encouragement during all the time of this work.

REFERENCES

- [1] H. J. Highland, “**Data encryption: a non-mathematical approach-Part 5,**” Journal of computer and Security, pp.93-97, 1995.
- [2] Xiaoyun Wang, Dengguo Feng, Xuejia Lai and Hongbo Yu, “**Collisions for Hash Functions MD4, MD5, HAVAL-128 and RIPEMD**”, Science and Communication, <http://eprint.iacr.org/2004/199.pdf>, 2004.
- [3] Ch. Rupa and P. S. Avadhani, “**An Improved Method to Reduce the Occurrence of Collision Attack on Hash Function**”, Int. J. computing mathematical applications, vol2, No1-2, pp.121-131, 2008.
- [4] H. Dobbertin, “**Cryptanalysis of MD5 Compress**”, proc. of Eurocrypt '96, 1996.
- [5] R.L. Rivest “**RFC 1321: The MD5 Message-Digest Algorithm**”, M.I.T. Laboratory for Computer Science and RSA Data Security, Inc., 1992
- [6] Ch. Rupa and P. S. Avadhani, “**Message Encryption Scheme Using Cheating Text**”, ITNG ISBN: 978-0-7695-3596-8/09(indexed by IEEE,dblp), pp. 470-475, 2009.
- [7] Aamer Nadeem et al, “**A Performance Comparison of Data Encryption Algorithms**”, IEEE information and communication, pp .84-89, 2005.
- [8] Priya Dhawan., “**Performance Comparison: Security Design Choices**”, Microsoft Developer NetworkOctober2002.<http://msdn2.microsoft.com/en-us/library/ms978415.aspx>
- [9] R Weis and S Lucks, “**Cryptographic Hash Functions-Recent Results on Cryptanalysis and their Implications on System Security**”, 5th System Administration and Network Engineering Conference, pp 15-19, 2006.
- [10] M. Bellare, R. Canetti and H. Krawczyk, “**Keying hash functions for message Authentication**”, Journal of Advances in Cryptology — Crypto '96, pp.1-15, 1996.
- [11] P. Louridas, “**JUnit: unit testing and coiling in tandem**”, Software, IEEE, Vol 22, pp.,12- 15, 2005.
- [12] Ch.Rupa, Prof P.S. Avadhani and Ch. Devi Chamundeswari, “**An Encrypto Stego Technique in wireless communication**”, IEEE – sipicom, pp. 28-30, 2006.

Finding Error Correction of Bandwidth on Demand Strategy for GPRS Using Constant Modulus Algorithm

K.Ramadevi

Assistant Professor, Dept of Computer Science, LBReddy College of Engineering, Mylavaram
E-mail: ramadevi_k32@yahoo.com

K. Nageswara Rao

Professor, HOD - Dept of CSE, PVP Siddhartha Institute of Technology, Vijayawada-7

J.Ramadevi

Assistant Professor, Dept of CSE, PVP Siddhartha Institute of Technology, vijayawada-7

Abstract- In this paper an attempt is made to Study behavior of the Constant Modulus Algorithm (CMA) as used to adapt a Fractionally Spaced Equalizer (FSE). It is illustrated below using simple examples. Some important conclusions are then drawn from these examples: (1) in the presence of noise or channel under modeling, proper initialization may be necessary to successfully equalize the channel, and (2) channels containing nearly reflected roots exhibit high levels of noise gain. Classical Equalization techniques employ a timeslot during which a training signal known in advance by the receiver is transmitted. As inclusion of such signals sacrifices valuable channel capacity To avoid this disadvantage Here we are using the BERgulator which is a MATLAB 5-based interactive simulation tool aimed to generate CMA for studying the behavior of the Constant Modulus Algorithm (CMA). By adjusting channel coefficients, dB SNR, and selecting appropriate channel type, spacing we can obtain perfect Equalizability with well behaved channel which reduces ISI at the receiver. General packet radio service is a global system for mobile communications. Packet data service which provide efficient access to the internet from mobile networks. In order to efficiently accommodate internet traffic that is bursty in nature while maintaining the desire service quality of GSM calls. We propose a GPRS bandwidth allocation strategy called bandwidth on demand strategy. The BOD strategy is adaptive to the change of traffic conditions and thus dynamically adjusts the number of channels for GSM and GPRS traffic loads. We also propose a analytical model to study the GSM call blocking probability and GPRS dropping probability. Based on the analytical model an iteration algorithm is proposed to determine the optimum bandwidth allocation strategy for GSM and GPRS traffic loads. We are newly implement the CMA algorithm to reduce the error rates. To determine the optimum values.

Keywords:

Fractionally Spaced Equalizer (FSE), Constant modulus Algorithm (CMA), Inter Symbol Interference (ISI), Binary Phase Shift Keying (BPSK), General packet radio service (GPRS), GlobalSystem for

mobile communications (GSM), Quality of service (Qos), Internet, bandwidth-on-demand strategy (BOD)

I. INTRODUCTION

General Packet Radio Service (GPRS) is a Global System for Mobile Communications (GSM) packet data service, which provides efficient access to the Internet from mobile networks. The main advantage is to accommodate Internet traffic that is bursty in nature while maintaining the desired service quality of GSM calls. A GPRS bandwidth allocation strategy called the BANDWIDTH-ON-DEMAND (BoD) strategy. The BoD strategy is adaptive to the change of traffic conditions, dynamically adjust the number of channels for GSM and GPRS traffic. An iterative algorithm is proposed to determine the optimum bandwidth allocation for GSM and GPRS traffic. Efficiently bandwidth utilization with high QoS requirements for both GSM and GPRS traffic. Information bearing signals transmitted between remote locations often encounter a signal altering physical channel. Examples of common physical channels include co axial, fiber optic or twisted pair cable in wired communications and the atmosphere or ocean in wireless communications. Each of these physical channels may cause signal distortion, including echoes and frequency selective filtering of the transmitted signal. In digital communications, a critical manifestation of distortion is inter symbol interference (ISI), whereby symbols transmitted before and after a given symbol corrupt the detection of that symbol. All physical channels tend to exhibit ISI. The presence of ISI is readily observable in the sampled impulse response of a channel, an impulse response corresponding to a lack of ISI contains a single spike of width less than the time between the symbols. Linear channel equalization, an approach commonly used to counter the effects of linear channel distortion, can be viewed as application of linear filter to the received signal. The equalizer attempts to extract the transmitted symbol sequence by counteracting the effects of

ISI, thus improving the probability of correct symbol detection. Since it is common for the channel characteristics to be unknown or to change over time, the preferred embodiment of the equalizer is a structure adaptive in nature. Classical Equalization techniques employ a timeslot during which a training signal known in advance by the receiver is transmitted. The receiver adopts the equalizer so that output Closely matches the known reference training signal. As inclusion of such signals sacrifices valuable channel capacity, adaption without resort to training i.e. blind adoption is preferred. So here we are implementing blind adoption algorithm constant Modulus Algorithm (CMA).

II. ARCHITECTURE

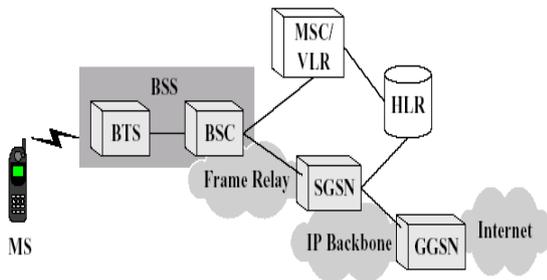


Figure 1: GPRS system architecture.

- The GGSN can be seen as an edge Internet protocol (IP) router providing connectivity to the Internet, which routes packets between the Internet and the SGSN.
- The SGSN is responsible for delivering packets to the mobile stations (MSs), and also responsible for performing security, mobility management, and session management functions.
- The SGSN is connected to the GGSN over an IP backbone network on one side and to the base station subsystem (BSS) over a Frame Relay network on the other side.
- The BSS consists of the base transceiver station (BTS) and the base station controller (BSC).

The BTS handles the radio interfaces toward the MSs. The BSC controls the Usage of the radio resources, where the physical radio channels can be used as either circuit-switched GSM channels or packet-switched GPRS channels.

When the GPRS-attached MS attempts to send/receive packet data to/from the Internet, a packet data protocol (PDP) context with a specific quality-of-service (QoS) profile between the MS and the GGSN shall be activated [3]. As shown in Figure 2, the MS activates a PDP context by sending an ACTIVATE PDP CONTEXT REQUEST message with the required QoS

profile to the SGSN. Upon receipt of this message, the SGSN sends a CREATE PDP CONTEXT REQUEST message to request the GGSN to establish an IP tunnel with the required QoS profile between the SGSN and the GGSN. The GGSN then returns a CREATE PDP CONTEXT RESPONSE message to indicate whether the establishment of an IP tunnel was successful. Besides, in order to request the creation of the packet flow context (PFC) between the MS and the SGSN, the SGSN sends a CREATE BSS PFC message with the aggregate BSS QoS profile (ABQP)2 to the BSC. If the PFC creation request was accepted, a CREATE BSS PFC ACK is returned to the SGSN. Finally, if the IP tunnel establishment and the PFC creation were successful, the SGSN returns an ACTIVATE PDP CONTEXT ACCEPT message to the MS. The MS is now able to access mobile Internet services.

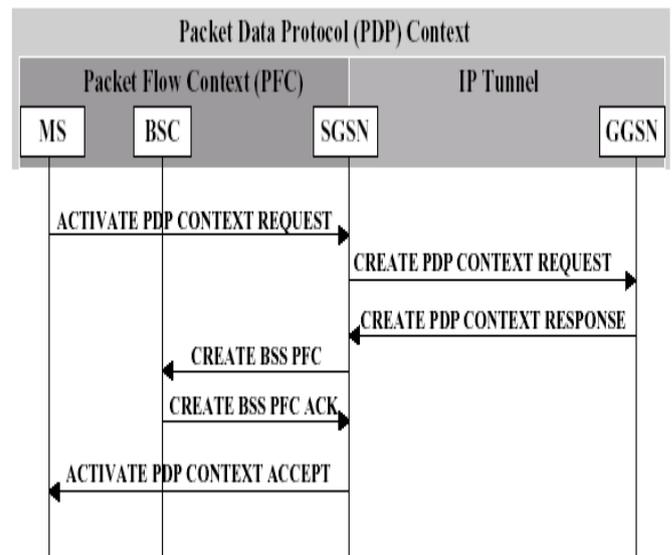


Figure 2: PDP context activation procedure.

III. THE BANDWIDTH-ON-DEMAND (BOD) STRATEGY

Let C be the Number of Channels in the BSC. Let M be the M Number of Channels designated for GSM. Let G be the Hand – off GSM calls among M Channels. A hand-off GSM call is accepted as long as there are free channels. A new GSM call is accepted only if the number of free channels is greater than G. A GPRS packet is served only if the number of GPRS packets being served is less than c–m otherwise it is queued in a buffer with capacity B

Two performance measures are considered in our study: The GSM-call blocking probability PB GPRS-packet dropping probability PD QoS is defined as:

$$Q = \alpha (1 - P_B) + (1 - \alpha)(1 - P_D)$$

$$0 \leq \alpha \leq 1$$

The value of α depends on the stress laid on the QoS requirements for GSM and GPRS traffic. Consequently, the core idea of the BoD strategy is to dynamically adjust the values of m and g based on the traffic load conditions so as to maximize the quality of service index Q .

Figure3 illustrates the BSC architecture for supporting the BoD strategy, which consists of the following components.

Traffic Measurement Unit (TMU) is responsible for traffic measurement at the BSC. The measurement is based on statistics collected during fixed time intervals called measurement periods. In each measurement period, the TMU determines the offered GSM and GPRS traffic loads, and informs the bandwidth controller (BC).

Bandwidth Controller (BC) dynamically allocates bandwidths on the basis of traffic information sent from the TMU. The BC determines the optimum values of m and g according to the measured traffic loads and the QoS index Q defined in (1). In each measurement period, the BC informs the access controller (AC) of the optimum values of m and g . The threshold-based policy is used to manage the variations of m and g by tracking traffic load fluctuations. Specifically, the load-to-threshold mapping is available as a look-up table whose entries are *a priori* calculated according to the QoS index.

Access Controller (AC) is responsible for assigning bandwidth resources to GSM and GPRS traffic according to the values of m and g sent from the BC.

Hand-off GSM call is accepted as long as there are available channels. A new GSM call, however, is accepted only when the number of idle channels is greater than g . A GPRS packet is served only if the number of GPRS packets being served is less than $c-m$ and there are channels free. Otherwise, it is queued in a buffer.

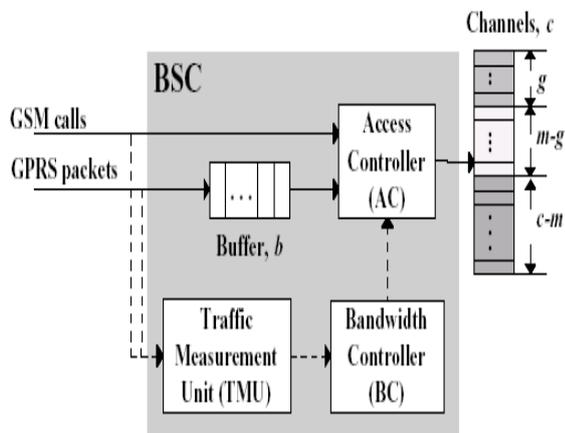
Figure3 illustrates the BSC architecture for supporting the BoD strategy, which consists of the following components.

Traffic Measurement Unit (TMU) is responsible for traffic measurement at the BSC. The measurement is based on statistics collected during fixed time intervals called measurement periods. In each measurement period, the TMU determines the offered GSM and GPRS traffic loads, and informs the bandwidth controller (BC).

Bandwidth Controller (BC) dynamically allocates bandwidths on the basis of traffic information sent from the TMU. The BC determines the optimum values of m and g according to the measured traffic loads and the QoS index Q defined in (1). In each measurement period, the BC informs the access controller (AC) of the optimum values of m and g . The threshold-based policy is used to manage the variations of m and g by tracking traffic load fluctuations. Specifically, the load-to-threshold mapping is available as a look-up table whose entries are *a priori* calculated according to the QoS index.

Access Controller (AC) is responsible for assigning bandwidth resources to GSM and GPRS traffic according to the values of m and g sent from the BC.

Hand-off GSM call is accepted as long as there are available channels. A new GSM call, however, is accepted only when the number of idle channels is greater than g . A GPRS packet is served only if the number of GPRS packets being served is less than $c-m$ and there are channels free. Otherwise, it is queued in a buffer..



IV. THE ANALYTICAL MODEL

The Analytical Model is developed to study the performance of the BOD strategy

The following parameters and assumptions are made:

1. The arrivals of new and hand-off GSM calls are Poisson distributed with rates λ_v and λ_{vh} respectively. λ_{vh} is correlated with other parameters (eg., call arrival rate, portable mobility, service time etc) and can be determined from them.
2. The arrivals of GPRS data packets are Poisson distributed with rate λ_d . A GPRS packet being served or queued is immediately dropped, if the corresponding Mobile station leaves the BSC service area.
3. The service rates of GSM calls and GPRS packets are exponentially distributed with means $1/\mu_v$ and $1/\mu_d$ respectively.
4. The BSC area residence times for GSM and GPRS MS's follow exponential distribution with means $1/\eta_v$ and $1/\eta_d$.

State S_{ij} where i and j denote the number of existing GSM calls and GPRS packet in BSC. The State Space is denoted by $E = \{s_{ij} | 0 \leq i < m, 0 \leq j \leq c - m + b \text{ and } m \leq i \leq c, 0 \leq j \leq c - i + b\}$

With normalization condition

$$\left(\sum_{i=0}^{m-1} \sum_{j=0}^{c-m+b} P_{ij} + \sum_{i=m}^c \sum_{j=0}^{c-i+b} P_{ij} \right) = 1$$

1) GSM-Call Blocking Probability

Consider the new-call and hand-off blocking probabilities for GSM calls in the BSC. A new GSM call is granted to access a free channel only if the number of free channels is greater than g . Thus, if the number of existing GSM calls is less than $m-g$, a new GSM call is accepted. If the number of existing GSM calls is greater than or equal to $m-g$, a new GSM call is accepted only when the channel occupancy is smaller than $c-g$. Thus, the new-call blocking probability of GSM calls P_{NB} is given by

$$P_{NB} = 1 - \left(\sum_{j=0}^{m-g-1} \sum_{j=0}^{c-m+b} P_{ij} + \sum_{i=m-g}^{c-g-1} \sum_{j=0}^{c-i-g} P_{ij} \right)$$

A hand-off GSM call is accepted as long as there are free channels. If the number of existing GSM calls is less than m , a hand-off GSM call is accepted. When the number of existing GSM calls is greater than or equal to m , A hand-off GSM call is accepted when the number of busy channels is

less than c . The hand-off blocking probability of GSM calls P_{HB} can thus be expressed as

$$P_{HB} = 1 - \left(\sum_{i=0}^{m-1} \sum_{j=0}^{c-m+b} P_{ij} + \sum_{i=m}^{c-1} \sum_{j=0}^{i+j < c} P_{ij} \right)$$

Assume homogeneous BSCs in the GPRS network. Since the probability that an accepted GSM call will attempt to hand off is $\eta_v / (\eta_v + \mu_v)$ the rate of hand-off GSM call arrivals can be expressed as

$$\lambda_v^h = \left(\frac{\eta_v}{\eta_v + \mu_v} \right) \left[\lambda_v (1 - P_{NB}) + \lambda_v^h (1 - P_{HB}) \right]$$

The GSM-call blocking probability is computed as follows. η_N be the number of new call requests, η_H be the number of hand-off call requests, η_{NB} be the Number of blocked new call requests, η_{HB} be the number of blocked hand-off

Call requests in a time interval of length t . As t goes to ∞ , the GSM-call blocking Probability P_B can be expressed as:

$$P_B = \eta_{NB} + \eta_{HB} / \eta_N + \eta_H$$

2) GPRS – PACKET DROPPING PROBABILITY

Now consider the GPRS-packet dropping probability in the BSC. A GPRS packet may be dropped for one of three reasons. The first is that as a GPRS packet arrives, no legitimate channel is available and the buffer is full, called *immediate dropping*. The second is that although a GPRS packet has been accepted and is waiting in the queue, it fails to access a free channel before the corresponding MS leaves the current BSC area. This packet is removed from the queue, and the action is called *queued dropping*. The third is that although a GPRS packet has been granted access to a free channel, it fails to be transmitted before the corresponding MS leaves the current BSC area, called *granted dropping*. The immediate dropping probability P_D^I

$$P_D^I = \sum_{i=0}^{m-1} P_{i, c-m+b} + \sum_{i=m}^c P_{i, c-i+b}$$

- The first is that as a GPRS packet arrives, no legitimate channel is available and the buffer is full, called *immediate dropping*.
- The second is that although a GPRS packet has been accepted and is waiting in the queue, it fails to access a free channel before the corresponding MS leaves the current BSC area. This packet is removed from the queue, and the action is called *queued dropping*.
- The third is that although a GPRS packet has been granted access to a free channel, it fails to be transmitted before the corresponding MS leaves the current BSC area, called *granted dropping*.

V. OPTIMUM BANDWIDTH ALLOCATION

In the BoD strategy, determining the optimum bandwidth allocation for GSM and GPRS traffic is equivalent to determining the optimum values for m and g that maximize the QoS index Q defined in (3). Let m^* and g^* be the optimum values for m and g respectively that achieve the maximal QoS index Q . This section proposes an iterative algorithm to determine m^* and g^* using equations derived.

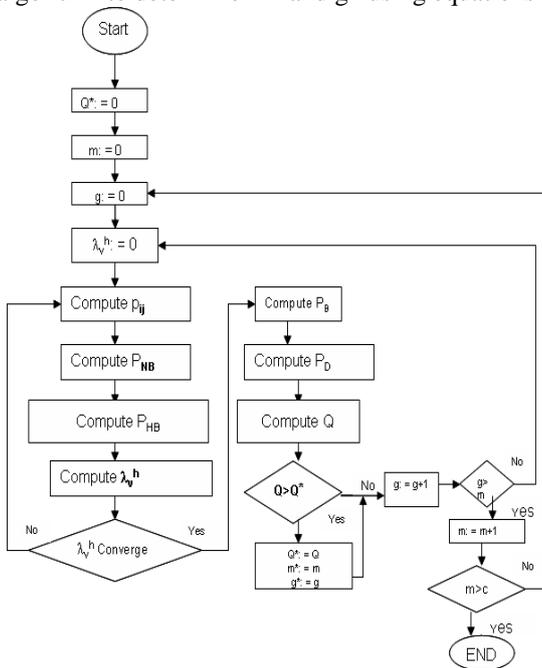


Figure 4: Iterative algorithm for determining m^* and g^* .

VI. ILLUSTRATIONS OF THE MA-FSE COST SURFACE

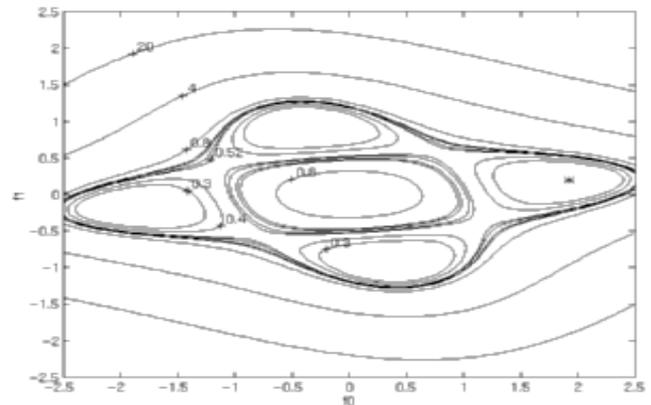
In one regard, the goal of CMA is to transfer a set of initial equalizer parameters to the lowest point on the CMA cost surface. As evident below, there may be multiple locations on the cost surface attaining the minimum cost. As such, they represent equally desirable solutions.

All of the contour plots exhibit the following properties: at the origin there exists a *maximum*, surrounded by a ring of

up to four *minima*. Minima reflected across the origin correspond to solutions differing only in sign, which is inconsequential when using differential coding. Finally, as we travel further from the origin, the surface rises steeply in all directions.

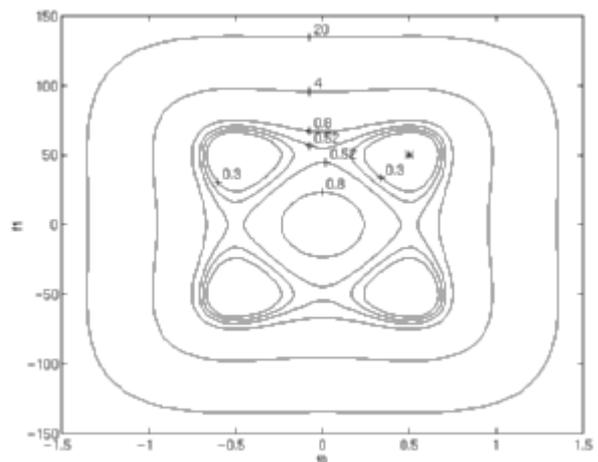
It should be noted that, though the following surfaces were created from particular channels, they are intended to be prototypical of their respective channel classes. For simplicity, BPSK was chosen as the model for the source statistics, though the thrust of the conclusions below applies to higher-order constellations (e.g. 64-QAM) as well.

Perfect Equalizability with Well-Behaved Channel



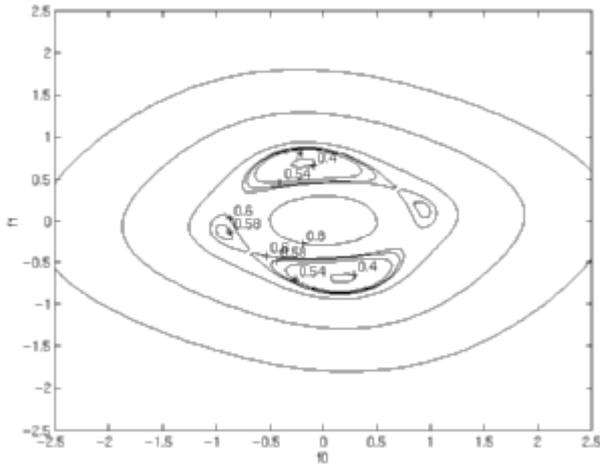
In the case of perfect equalizability, all four minima correspond to the same (optimal) CMA cost of zero. Each minima reflects a particular combination of system delay and system polarity. The asterisk indicates the location of a particular Wiener solution. Since there is no noise, it appears coincident with the CMA cost minimum of corresponding system delay and polarity.

Perfect Equalizability with Nearly-Reflected Channel Roots



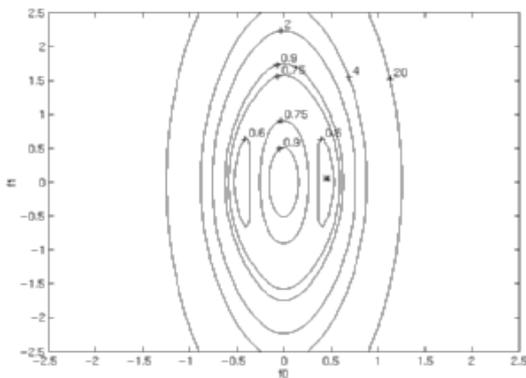
As channel roots become more nearly reflected, the surface becomes more elongated. Note the scaling on the vertical axis is 100 times that on the horizontal axis! This characteristic will be manifested as an extremely slow convergence of the 'f1' equalizer tap due to the shallowness of the gradient along that dimension. Theoretically, however, perfect equalizability *is* still possible in the noiseless case. In other words, all minima correspond to zero cost and are aligned with the respective Wiener solutions.

Noisy but Well-Behaved Channel



When the well-behaved channel (i.e. the first plot above) is operated in a 7dB SNR environment, note the appearance of *local minima* distinct from the *global minima*. For particular initializations, local minima are capable of capturing the parameter trajectory and preventing it from attaining an optimum setting. Note that all minima correspond to costs greater than zero since perfect equalization is not possible in the presence of noise.

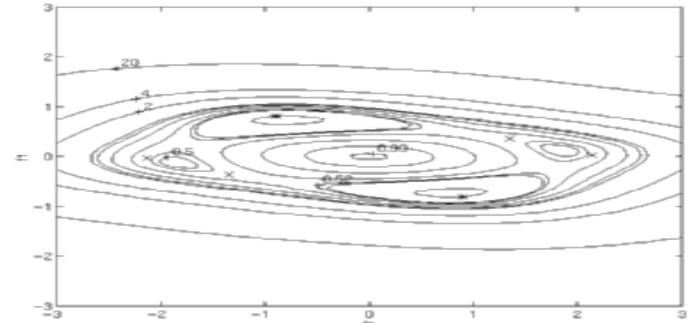
Noisy Channel with Nearly-Reflected Roots



When the channel with nearly reflected roots (above) is operated in a 7dB SNR environment, we see a dramatic change in the cost surface. Note the equivalent scaling of horizontal and vertical axes; the cost surface has become much less elongated than in the noiseless case (see the

second picture in the series). Also note the merging of two pairs of minima into one pair. For this channel, the optimal solution in the presence of noise is very different from the solution which perfectly equalizes the channel.

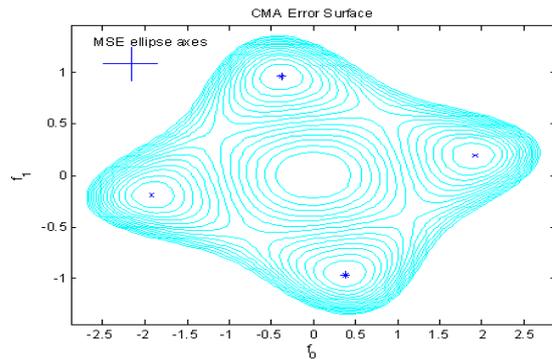
Undermodelled but Well-Behaved Channel, No Noise



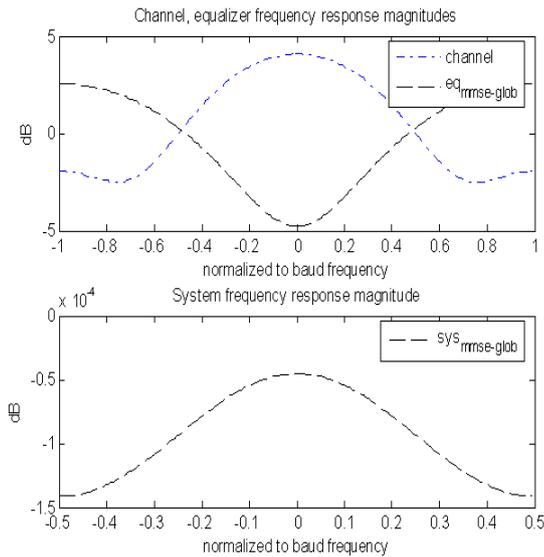
When the delay spread of the channel is *longer* than the delay spread of the FSE, we lose the ability to perfectly equalize. Thus, none of the minima correspond to zero cost. More importantly, note the existence of local minima with greater CMA cost than the global minima. This implies that the equalizer initialization will determine the optimality of the resulting CMA solution.

VII. RESULTS

To obtain perfect Equalizability with well behaved channel the values are given to the system are channel type 4 -tap, spacing (BS), source type (Bpsk), channel coefficients (0.2, 0.5, 1, - 0.1) and Dbsnr (50). Then we can obtain the following **result**.



In this system, we can obtain the Frequency response magnitude. The values are given to the system are channel type 4 -tap, spacing (BS), source type (Bpsk), channel coefficients (0.2 0.5 1 - 0.1) and Dbsnr (50). Then we can obtain the following result.



The equalizer attempts to extract the transmitted symbol sequence by counteracting the effects of ISI, thus improving the probability of correct symbol detection. Here we are using the BERGulator which is a MATLAB 5-based interactive simulation tool aimed to generate CMA for studying the behavior of the Constant Modulus Algorithm (CMA). By adjusting channel coefficients, dB SNR, and selecting appropriate channel type, spacing we can obtain perfect Equalizability with well behaved channel which attempts to extract the transmitted symbol sequence by counteracting the effects of ISI, thus improving the probability of correct symbol detection.

REFERENCES:

- [1] Babak h. Khalaj, Arogyaswami Paulraj, and Thomas Kailath. 2 D Rake Receivers for CDMA Cellular Systems. "1994".
- [2] Esmael H. Dinan and Bijan Jabbari. Spreading Codes for Direct Sequence CDMA and Wideband CDMA Cellular Networks. "1998".
- [3] J.G. Proakis, *Digital communications*. "1995".
- [4] Kevin Laird, Nick Whinnet, and Soodesh Buljore. A Peak-To-Average Power Reduction Method for Third Generation CDMA Reverse Links. "1999".
- [5] Jochen Schiller, "Mobile communications", second Edition, Pearson Education. "2009"
- [6] ETSI, GSM 03.60, "General packet radio service (GPRS); service description; stage 2," v7.4.1, September 2000.
- [7] Y.-B. Lin and I. Chlamtac, *Wireless and Mobile Network Architecture*, John Wiley & Sons, 2001.
- [8] Y.-R. Haung and Y.-B. Lin, "A software architecture for GPRS session management," *Wireless Communications and Mobile Computing*, 2(2), pp. 151–167, 2002.
- [9] ETSI, GSM 08.18, "General packet radio service; base station system (BSS) – serving GPRS support node (SGSN); BSS GPRS protocol (BSSGP)," v8.3.0, May 2000.
- [10] Y.-B. Lin, Y.R. Haung, Y.K. Chen, and I. Chlamtac, "Mobility management: from GPRS to UMTS," *Wireless Communications and Mobile Computing*, 1(4), pp. 339–359, 2001.
- [11] D. K. Kim and D. K. Sung, "Traffic management in a multicode CDMA system supporting of handoffs," *IEEE Trans. Veh. Technol.*, 51(1), pp. 52–62, 2002.
- [12] Y.-R. Haung and J.-M. Ho, "Distributed call admission control for a heterogeneous PCS network," *IEEE Trans. Computers*, 51(12), pp. 1400–1409, 2002.
- [13] W. Zhuang, B. Bensaou, and K. C. Chua, "Adaptive quality of service handoff priority scheme for mobile multimedia networks," *IEEE Trans. Veh. Technol.*, 49(2), pp. 494–505, 2000.
- [14] L. Ortigoza-Guerrero and A. H. Aghvami, "A prioritized handoff dynamic channel allocation strategy for PCS," *IEEE Trans. Veh. Technol.*, 48(4), pp. 1203–1215, 1999.
- [15] C.-J. Chang, T.-T. Su, and Y.-Y. Chiang, "Analysis of a cutoff Priority cellular radio system with finite queuing and reneing/dropping," *IEEE/ACM Trans. Networking*, 2(2), pp. 166–175, 1994.
- [16] R. Cooper, *Introduction to Queuing Theory*, New York: North-Holland, 1981.

An Introduction to DNA Computing

SARAVANAN C^α

VEL SRI RANGA SANKU COLLEGE, AVADI

Abstract- DNA (Deoxyribose Nucleic Acid) computing, also known as molecular computing is a new approach to massively parallel computation based on groundbreaking work by Adleman. DNA computing was proposed as a means of solving a class of intractable computational problems in which the computing time can grow exponentially with problem size (the 'NP-complete' or non-deterministic polynomial time complete problems). A DNA computer is basically a collection of specially selected DNA strands whose combinations will result in the solution to some problem, depending on the problem at hand. Technology is currently available both to select the initial strands and to filter the final solution. DNA computing is a new computational paradigm that employs (bio)molecular manipulation to solve computational problems, at the same time exploring natural processes as computational models. In 1994, Leonard Adleman at the Laboratory of Molecular Science, Department of Computer Science, University of Southern California surprised the scientific community by using the tools of molecular biology to solve a different computational problem. The main idea was the encoding of data in DNA strands and the use of tools from molecular biology to execute computational operations. Besides the novelty of this approach, molecular computing has the potential to outperform electronic computers. For example, DNA computations may use a billion times less energy than an electronic computer while storing data in a trillion times less space. Moreover, computing with DNA is highly parallel: In principle there could be billions upon trillions of DNA molecules undergoing chemical reactions, that is, performing computations, simultaneously.

I. HISTORY & MOTIVATION:

"Computers in the future may weigh no more than 1.5 tons." So said Popular Mechanics in 1949. Most of us today, in the age of smart cards and wearable PCs would find that statement laughable. We have made huge advances in miniaturization since the days of room-sized computers, yet the underlying computational framework has remained the same. Today's supercomputers still employ the kind of sequential logic used by the mechanical dinosaurs of the 1930s. Some researchers are now looking beyond these boundaries and are investigating entirely new media and computational models. These include quantum, optical and DNA-based computers. It is the last of these developments that this paper concentrates on.

The current Silicon technology has following limitations:

- Circuit integration dimensions
- Clock frequency
- Power consumption
- Heat dissipation.

The problem's complexity that can be afforded by modern processors grows up, but great challenges require computational capabilities that neither most powerful and distributed systems could reach.

The idea that living cells and molecular complexes can be viewed as potential machinic components dates back to the late 1950s, when Richard Feynman delivered his famous paper describing "sub-microscopic" computers. More recently, several people have advocated the realization of massively parallel computation using the techniques and chemistry of molecular biology. DNA computing was grounded in reality at the end of 1994, when Leonard Adleman, announced that he had solved a small instance of a computationally intractable problem using a small vial of DNA. By representing information as sequences of bases in DNA molecules, Adleman showed how to use existing DNA-manipulation techniques to implement a simple, massively parallel random search. He solved the traveling salesman problem also known as the "Hamiltonian path" problem.

There are two reasons for using molecular biology to solve computational problems.

- (i) The information density of DNA is much greater than that of silicon: 1 bit can be stored in approximately one cubic nanometer. Others storage media, such as videotapes, can store 1 bit in 1,000,000,000,000 cubic nanometer.
- (ii) Operations on DNA are massively parallel: a test tube of DNA can contain trillions of strands. Each operation on a test tube of DNA is carried out on all strands in the tube in parallel.

II. DNA FUNDAMENTALS

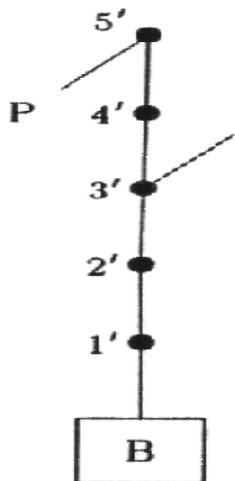
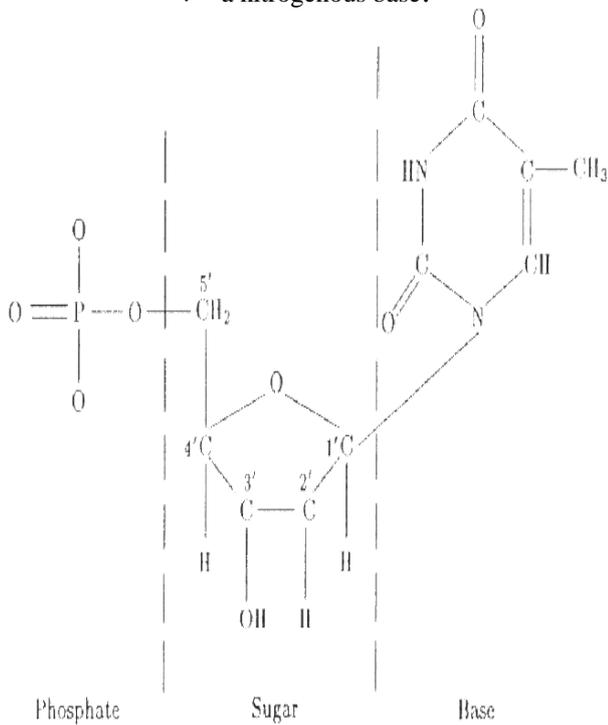
DNA (deoxyribonucleic acid) is a double stranded sequence of four nucleotides; the four nucleotides that compose a

strand of DNA are as follows: adenine (A), guanine (G), cytosine (C), and thymine (T); they are often called bases. DNA supports two key functions for life:

- ❖ coding for the production of proteins,
- ❖ self-replication.

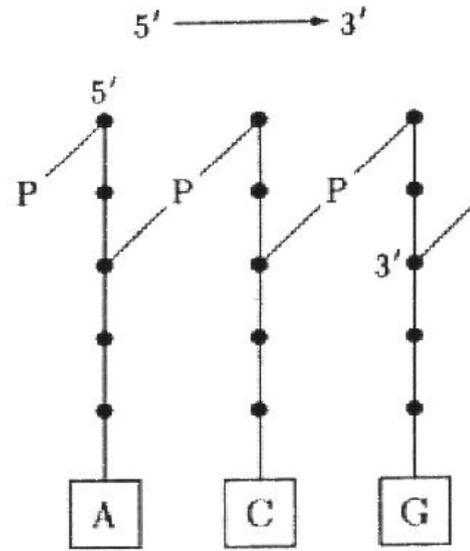
Each deoxyribonucleotide consists of three components:

- ❖ a sugar — deoxyribose
 - five carbon atoms: 1' to 5'
 - hydroxyl group (OH) attached to 3' carbon
- ❖ a phosphate group
- ❖ a nitrogenous base.

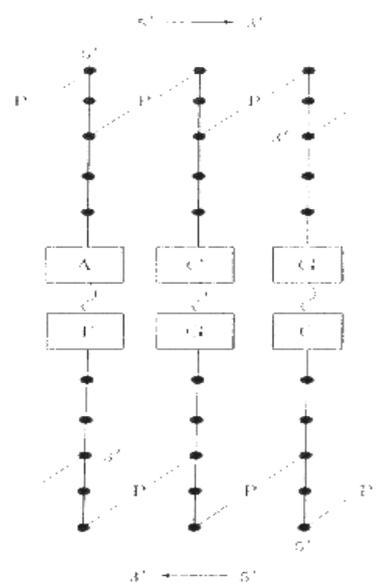


The chemical structure of DNA consists of a particular bond of two linear sequences of bases. This bond follows a property of Complementarity: adenine bonds with thymine (A-T) and vice versa (T-A), cytosine bonds with guanine (C-G) and vice versa (G-C). This is known as Watson-Crick complementarity.

The DNA monomers can link in two ways:

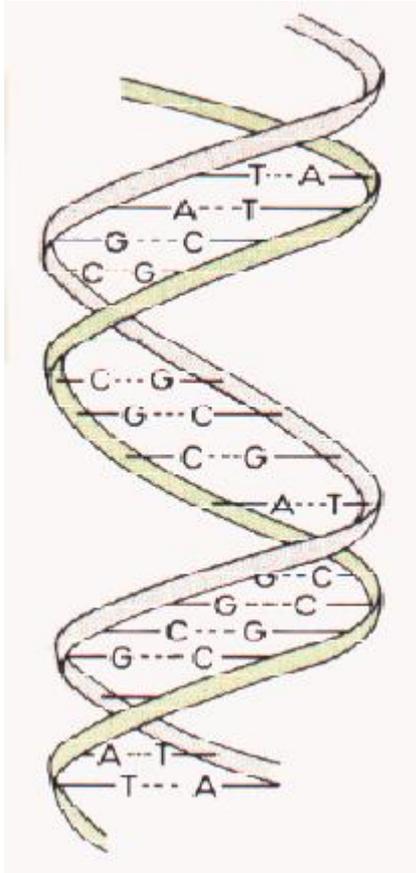


Phosphodiester bond



Hydrogen bond

The four nucleotides adenine (A), guanine (G), cytosine (C), and thymine (T) compose a strand of DNA. Each DNA strand has two different ends that determine its polarity: the 3' end, and the 5' end. The double helix is an parallel anti- (two strands of opposite polarity) bonding of two complementary strands.



The structure of DNA double helix

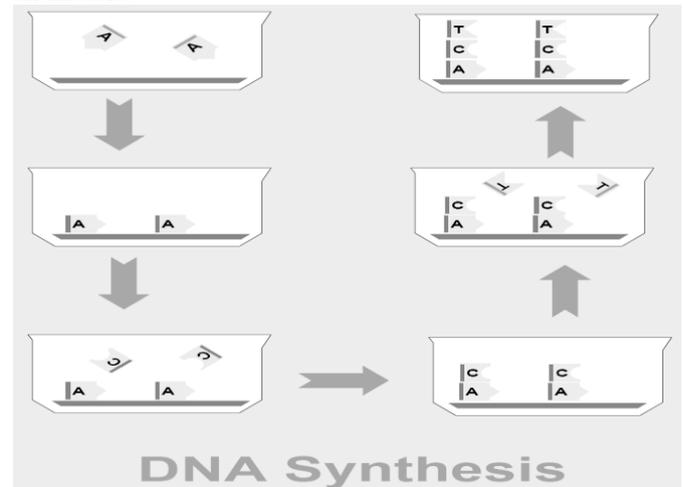
III. PRINCIPLES OF DNA COMPUTING

DNA is the major information storage molecule in living cells, and billions of years of evolution have tested and refined both this wonderful informational molecule and highly specific enzymes that can either duplicate the information in DNA molecules or transmit this information to other DNA molecules. Instead of using electrical impulses to represent bits of information, the DNA computer uses the chemical properties of these molecules by examining the patterns of combination or growth of the molecules or strings. DNA can do this through the manufacture of enzymes, which are biological catalysts that could be called the 'software', used to execute the desired calculation.

A single strand of DNA is similar to a string consisting of a combination of four different symbols A G C T. Mathematically this means we have at our disposal a letter alphabet, $\Sigma = \{A G C T\}$ to encode information which is

more than enough considering that an electronic computer needs only. Two digits and for the same purpose. In a DNA computer, computation takes place in test tubes or on a glass slide coated in 24K gold. The input and output are both strands of DNA, whose genetic sequences encode certain information. A program on a DNA computer is executed as a series of biochemical operations, which have the effect of synthesizing, extracting, modifying and cloning the DNA strands. As concerning the operations that can be performed on DNA strands the proposed models of DNA computation are based on various combinations of the following primitive bio-operations:

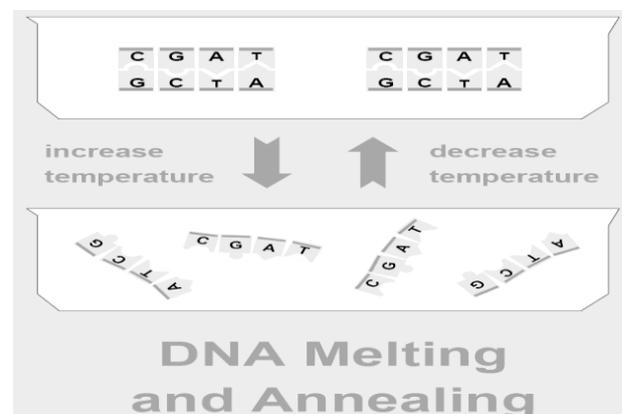
❖ **Synthesizing** a desired polynomial-length strand used in all models.



❖ **Mixing**: combine the contents of two test tubes into a third one to achieve union.

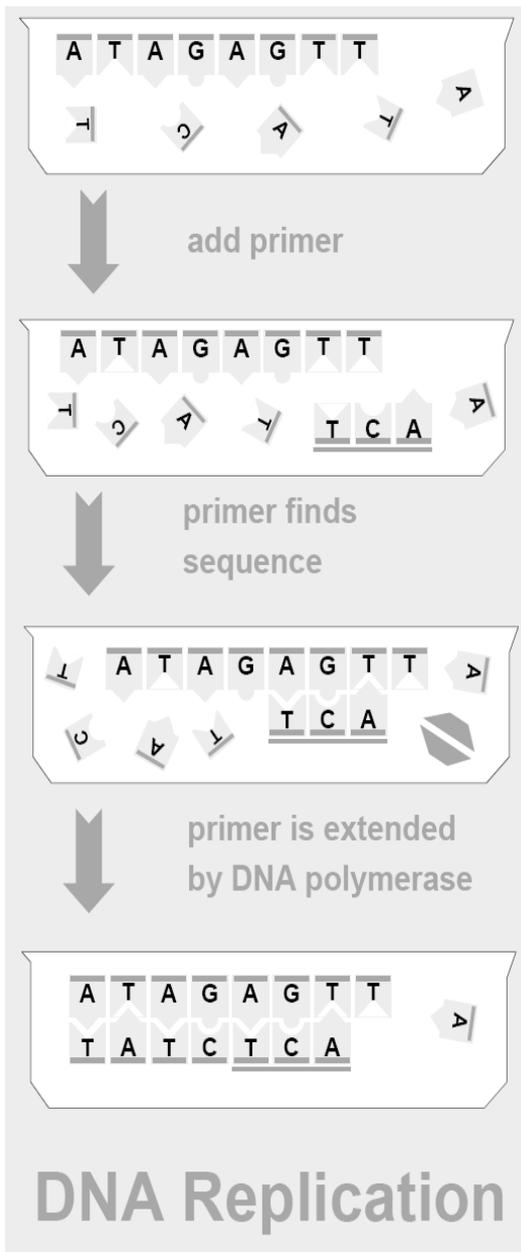
❖ **Annealing**: bond together two single-stranded complementary DNA sequences by cooling the solution. Annealing in vitro is known as hybridization.

❖ **Melting**: break apart a double-stranded DNA into its single-stranded complementary components by heating the solution. Melting in vitro is also known under the name of denaturation.

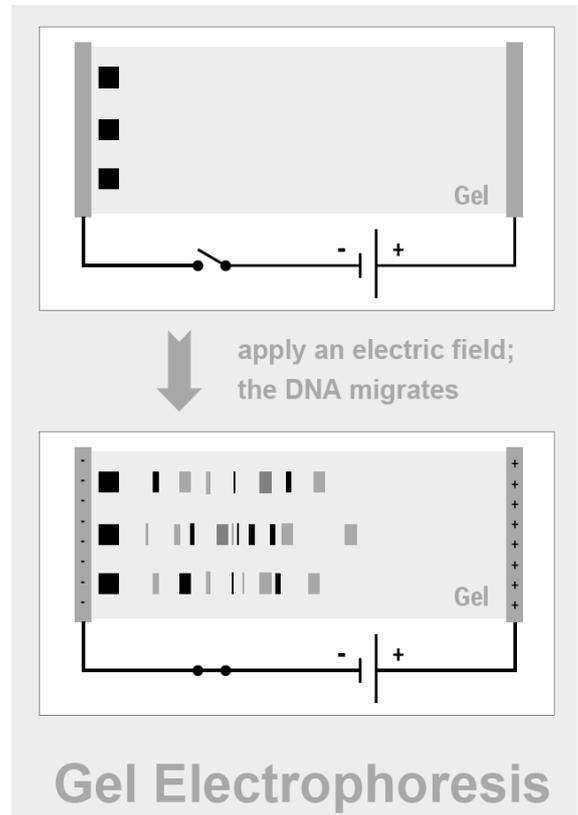


❖ **Amplifying (copying)**: make copies of DNA strands by using the Polymerase Chain Reaction PCR. The DNA polymerase enzymes perform several functions including replication of DNA. The replication

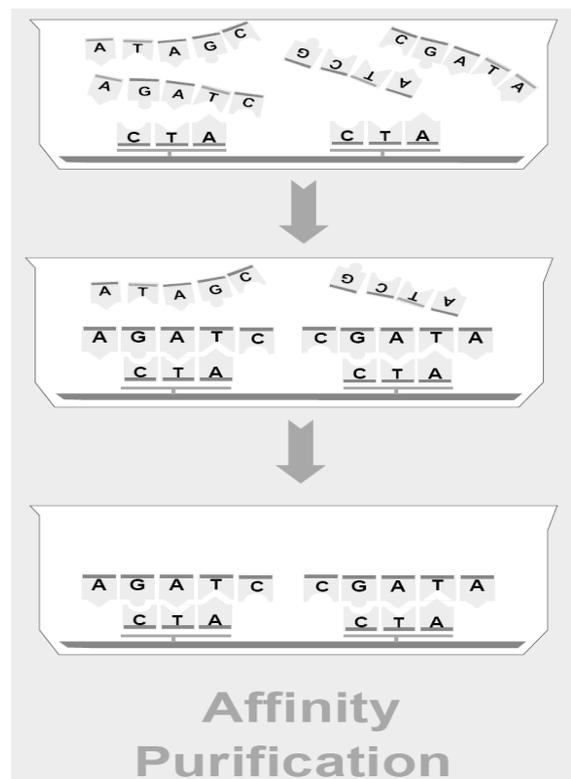
reaction requires a guiding DNA single-strand called **template**, and a shorter oligonucleotide called a **primer**, that is annealed to it.



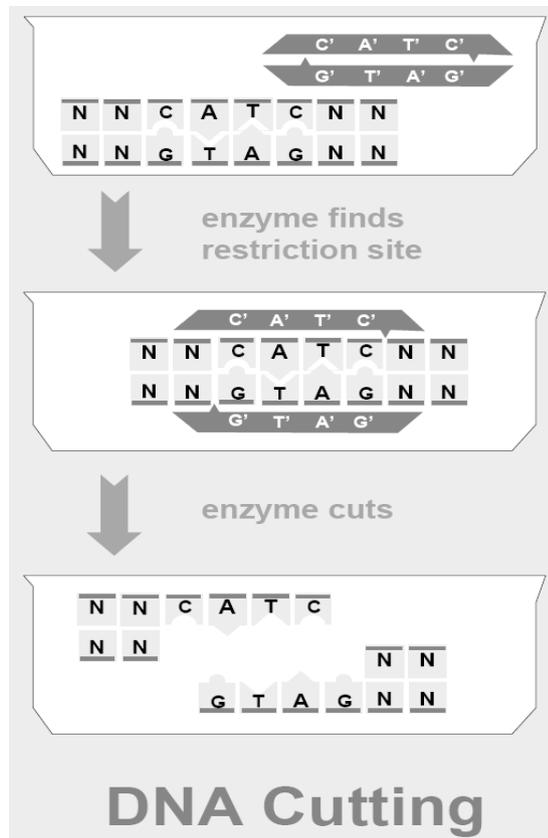
❖ **Separating** the strands by length using a technique called gel electrophoresis that makes possible the separation of strands by length.



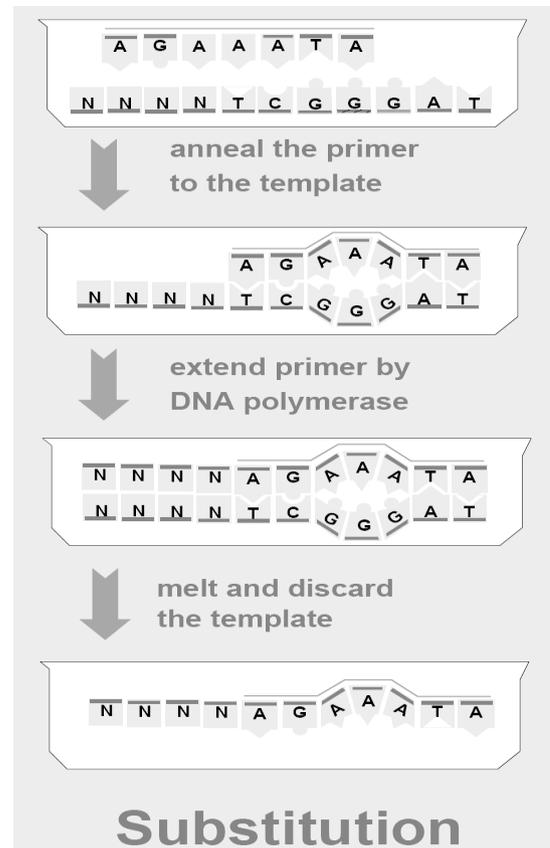
❖ **Extracting** those strands that contain a given pattern as a substring by using affinity purification.



- ❖ **Cutting** DNA double-strands at specific sites by using commercially available restriction enzymes. One class of enzymes, called restriction endonucleases, will recognize a specific short sequence of DNA, known as a restriction site. Any double-stranded DNA that contains the restriction site within its sequence is cut by the enzyme at that location.



- ❖ **Ligating:** paste DNA strands with compatible sticky ends by using DNA ligases. Indeed, another enzyme called **DNA ligase**, will bond together, or "ligate", the end of a DNA strand to another strand.
- ❖ **Substituting:** substitute, insert or delete DNA sequences by using PCR site-specific oligonucleotide mutagenesis.



- ❖ **Marking** single strands by hybridization: complementary sequences are attached to the strands, making them double-stranded. The reverse operation is **unmarking** of the double-strands by denaturing, that is, by detaching the complementary strands. The marked sequences will be double-stranded while the unmarked ones will be single-stranded.
- ❖ **Destroying** the marked strands by using exonucleases, or by cutting all the marked strands with a restriction enzyme and removing all the intact strands by gel electrophoresis. (By using enzymes called **exonucleases**, either double-stranded or single-stranded DNA molecules may be selectively destroyed. The exonucleases chew up DNA molecules from the end inward, and exist with specificity to either single-stranded or double-stranded form.)
- ❖ **Detecting and Reading:** given the contents of a tube, say "yes" if it contains at least one DNA strand, and "no" otherwise. PCR may be used to amplify the result and then a process called **sequencing** is used to actually read the solution.

In Short, DNA computers work by encoding the problem to be solved in the language of DNA: the base-four values A, T, C and G. Using this base four number system, the solution to any conceivable problem can be encoded along a DNA strand like in a Turing machine tape. Every possible sequence can be chemically created in a test tube on trillions of different DNA strands, and the correct sequences can be filtered out using genetic engineering tools.

IV. EXAMPLE OF DNA COMPUTING : THE HAMILTONIAN PATH PROBLEM

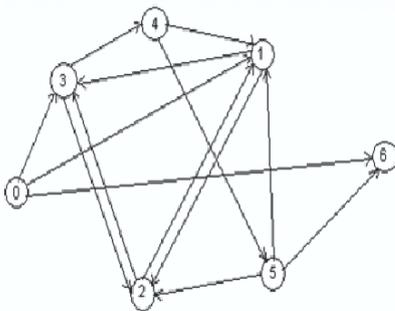
In 1994 Leonard M. Adleman showed how to solve the Hamilton Path Problem, using DNA computation.

Hamiltonian Path Problem:

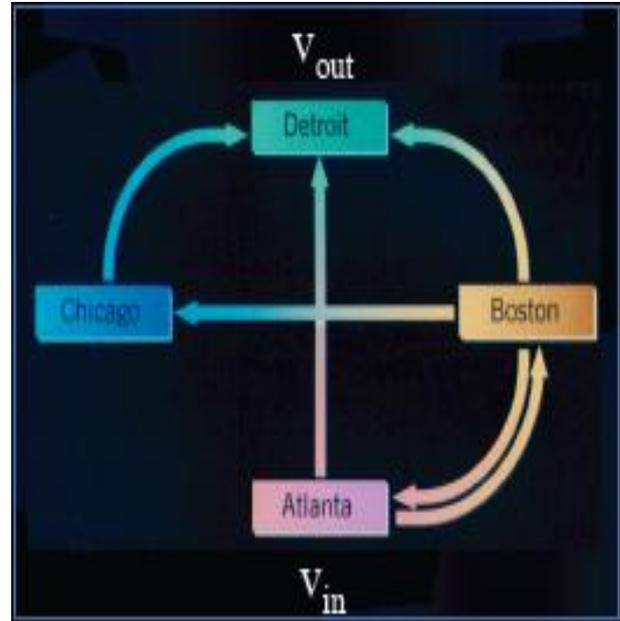
A directed graph G with designated nodes vin and $vout$ is said to have a Hamiltonian path if and only if there exists a sequence of compatible one-way edges $e1, e2, \dots, en$ that begins at vin , ends at $vout$ and enters every other node exactly once. A simplified version of this problem, known as the traveling salesman problem, poses the following question: given an arbitrary collection of cities through which a salesman must travel, what is the shortest route linking those cities?

This problem is difficult for conventional computers to solve because it is a "non-deterministic polynomial time problem". These problems, when the instance size is large, are intractable with conventional computers, but can be solved using massively parallel computers like DNA computers. NP problems are intractable with deterministic (conventional/serial) computers, but can be solved using non-deterministic (massively parallel) computers.

A DNA computer is a type of non-deterministic computer. The Hamiltonian Path problem was chosen by Adleman because it is known as "NP-complete".



Directed graph with node 0 as source (V_{in}) and node 6 as destination (V_{out})



Simplified graph

Hamiltonian path : Atlanta–Boston–Chicago–Detroit

Adleman's Algorithm

Input: A directed graph G with n vertices, and designated vertices vin and $vout$.

Step 1: Generate paths in G randomly in large quantities.

Step 2: Reject all paths that

- do not begin with vin and
- do not end in $vout$.

Step 3: Reject all paths that do not involve exactly n vertices.

Step 4: For each of the n vertices v :

- reject all paths that do not involve v .

Output: YES, if any path remains; NO, otherwise.

To implement step 1, each node of the graph was encoded as a random 20-base strand of DNA. Then, for each edge of the graph, a different 20-base oligonucleotide was generated that contains the second half of the source code plus the first half of the target node.

City	DNA Name	Complement
Atlanta	Atlanta	TGAACGTC
Boston	TCGGACTG	AGCCTGAC
Chicago	GGCTATGT	CCGATACA
Detroit	CCGAGCAA	GGCTCGTT

City	DNA Flight Number
Atlanta - Boston	CCAGTCGG
Atlanta - Detroit	GCAGCCGA
Boston - Chicago	ACTGGGCT
Boston - Detroit	ACTGCCGA
Boston - Atlanta	ACTGACTT
Chicago - Detroit	ATGTCCGA

To implement step 2, the product of step 1 was amplified by PCR using oligonucleotide primers representing *vin* and *vout* and *ligase enzyme*. This amplified and thus retained only those molecules encoding paths that begin with *vin* and end with *vout*. ~10¹⁴ computations are carried out in a single second.

For implementing step 3, agarose gel electrophoresis allowed separation and recovery of DNA strands of the correct length. The desired path, if it exists, would pass through all seven nodes, each of which was assigned a length of 20 bases. Thus PCR products encoding the desired path would have to be 140 bp.

Step 4 was accomplished by successive use of affinity purification for each node other than the start and end nodes.

- The solution strand has to be filtered from the test tube:

**GCAG TCGG ACTG GGCT ATGT
CCGA**

Atlanta → Boston → Chicago →

Detroit

Thus we see in a graph with n vertices, there are a possible $(n-1)!$ permutations of the vertices between beginning and ending vertex.

To explore each permutation, a traditional computer must perform $O(n!)$ operations to explore all possible cycles. However, the DNA computing model only requires the representative oligos. Once placed in solution, those oligos will anneal in parallel, providing all possible paths in the graph at roughly the same time. That is equivalent to $O(1)$ operations, or constant time. In addition, no more space than what was originally provided is needed to contain the constructed paths.

V. PRESENT & FUTURE DNA COMPUTER

A year ago, researchers from the Weizmann Institute of Science in Rehovot, Israel, unveiled a programmable molecular computing machine composed of enzymes and DNA molecules instead of silicon microchips. "This re-designed device uses its DNA input as its source of fuel," said Ehud Shapiro, who led the Israeli research team. This computer can perform 330 trillion operations per second, more than 100,000 times the speed of the fastest PC.

While a desktop PC is designed to perform one calculation very fast, DNA strands produce billions of potential answers simultaneously. This makes the DNA computer suitable for solving "fuzzy logic" problems that have many possible solutions rather than the either/or logic of binary computers. In the future, some speculate, there may be hybrid machines that use traditional silicon for normal processing tasks but have DNA co-processors that can take over specific tasks they would be more suitable for.

VI. ADVANTAGES

- Perform millions of operations simultaneously.
- Generate a complete set of potential solutions and conduct large parallel searches.
- Efficiently handle massive amounts of working memory.
- They are inexpensive to build, being made of common biological materials.
- The clear advantage is that we have a distinct memory block that encodes bits.
- Using one template strand as a memory block also allows us to use its complement. As another memory block, thus effectively doubling our capacity to store information.

VII. DISADVANTAGES

- Generating solution sets, even for some relatively simple problems, may require impractically large amounts of memory (lots and lots of DNA strands are required)

Many empirical uncertainties, including those involving: actual error rates, the generation of optimal encoding techniques, and the ability to perform necessary bio-operations conveniently in vitro (for every correct answer there are millions of incorrect paths generated that are worthless).

- DNA computers could not (at this point) replace traditional computers. They are not programmable and the average dunce can not sit down at a familiar keyboard and get to work.

REFERENCES

- Paun, G., Rozenberg, G., and Salomaa, A., *DNA Computing*, Springer, 1998.
- DNA COMPUTING-GRAPH ALGORITHMS [lec-12.pdf] G. P. Raja Sekhar, Dept. of Mathematics, IIT, Kharagpur
- Leonard M. Adleman, *Computing with DNA*, Scientific American, August 1998.
- From Microsoft to Biosoft Computing with DNA, Lila Kari, Department of Computer Science University of Western Ontario
- L. Adleman. On constructing a molecular computer. 1st DIMACS workshop on DNA based computers, Princeton, 1995. In DIMACS series, vol.27 (1996)
- L. Adleman, P. Rothemund, S. Roweis, E. Winfree. On applying molecular computation to the Data Encryption Standard. 2nd DIMACS workshop on DNA based computers, Princeton, 1996

Distributed Diagnosis in Dynamic Fault Environments using HeartBeat Algorithm

Dr.K.Nageswara Rao, Professor & Hod, B.Srinivasa Rao, Sr.Asst.Prof, Sreenivasa Raju V, Department of Computer Science, PVPSIT,Kanuru,Vijayawada

Abstract—The problem of distributed diagnosis in the presence of dynamic failures and repairs is considered. To address this problem, the notion of bounded correctness is defined. Bounded correctness is made up of three properties: bounded diagnostic latency, which ensures that information about state changes of nodes in the system reaches working nodes with a bounded delay, bounded start-up time, which guarantees that working nodes determine valid states for every other node in the system within bounded time after their recovery, and accuracy, which ensures that no spurious events are recorded by working nodes. It is shown that, in order to achieve bounded correctness, the rate at which nodes fail and are repaired must be limited. This requirement is quantified by defining a minimum state holding time in the system. Algorithm Heartbeat Complete is presented and it is proven that this algorithm achieves bounded correctness in fully-connected systems while simultaneously minimizing diagnostic latency, start-up time, and state holding time.

I. INTRODUCTION

An important problem in distributed systems that are subject to component failures is the distributed diagnosis problem. In distributed diagnosis, each working node must maintain correct information about the status (working or failed) of each component in the system. In this paper, we consider the problem of achieving diagnosis despite dynamic failures and repairs. In static fault situation statuses of nodes remain fixed for as long as it takes an algorithm to completely diagnose the system.

We present a formal model of dynamic behavior, which allows us to rigorously define what it means for a diagnosis algorithm to be correct in dynamic situations. This notion of correctness, referred to as bounded correctness, consists of three properties: bounded diagnostic latency, bounded start-up, and accuracy. For bounded diagnostic latency, all working nodes must learn about each event (node failure or repair) within a bounded time L . For bounded start-up, nodes that recover must determine a valid state for every other node within time S of entering the working state. Finally, accuracy ensures that no spurious events are recorded by any working node.

The specification of bounded correctness represents a strengthening of the properties required for a solution to the diagnosis problem. To our knowledge, no prior distributed diagnosis algorithm has been formally proven to achieve properties as strong as those of bounded correctness in the presence of truly dynamic failures and repairs. We present

herein the first algorithms for distributed diagnosis that are rigorously proven to be correct in dynamic fault situations. Furthermore, evaluation of prior algorithms shows that our algorithms achieve significantly shorter diagnostic latency while tolerating substantially higher event rates than previous ones. We show in the next section that previous algorithms, by focusing almost exclusively on minimizing the number of tests performed, have unintentionally made themselves quite vulnerable to dynamic environments.

II. RELATED WORK

The bulk of the work in system diagnosis has assumed a static fault situation [3], [9], [17], [18], [20], [23], i.e., the statuses of nodes do not change during execution of the diagnosis procedure. Some diagnosis algorithms, e.g., [5], [13], [19] allow dynamic failures and repairs to occur, but are only guaranteed to be correct when system status has become stable. One of the diagnosis algorithms in [4] assumes that nodes can fail dynamically, but cannot be repaired during execution of the diagnosis procedure. This approach is suitable in some systems, but is not a satisfactory solution in general. The diagnosis model of [14] considers dynamic failures but requires a centralized diagnosis entity.

Previous work on distributed diagnosis has focused almost exclusively on minimizing the number of tests performed. One interesting result of our work is to show that the goal of minimizing tests and the goal of effectively handling dynamic failures and repairs are directly in conflict. Prior algorithms that minimize the number of tests construct sparse testing graphs and propagate information in reverse direction of tests. The ideal testing property for which these algorithms strive is to have each node tested by exactly one other node at each testing round. With dynamic failures and repairs, the latency and the minimum time a node must remain in a particular state can be as high as $(n - 1)T$, where n is the number of nodes in the network and T is the duration of a testing round. Experiments in [5] on networked systems were conducted with parameters of $n = 60$ and $T = 30$ seconds, which yield a diagnostic latency and state holding time of about 30 minutes when dynamic environments are considered. The above result holds for all of the following algorithms: ADSD [5], BH [2], Adapt [24], and RDZ [19]. Other relevant algorithms such as Hi-ADSD and its variants [8], [9] have latencies of at least $\log_2^2 n$ rounds, which are still far greater than Algorithm HeartbeatComplete.

Since we assume testing is accomplished via heartbeat-based mechanisms which have low cost, we are not so concerned with the number of tests. Rather, we try to minimize diagnostic latency and state holding time in dynamic fault environments for completely-connected networks. Our algorithm for completely-connected networks, known as Algorithm HeartbeatComplete, has a latency of approximately one heartbeat transmission round and a state holding time of about half a round.

A problem closely related to distributed diagnosis, known as the synchronous group membership problem [6], [7], [11], [12], [15], is for each working node to maintain correct information about the group of working nodes with which it can communicate, and for all nodes in one group to agree on the membership of the group. In [12], it is shown that, under some models, these two problems are equivalent and an algorithm for one problem can be converted to an algorithm for the other. The primary difference between the two approaches is the requirement of agreement among nodes in the group membership problem, while agreement is not required in distributed diagnosis.

The work is closest to the approach taken in [6], where there is no limit on how many nodes can change state during execution of the algorithm, but there is a limit on how frequently an individual node can change state. The algorithm of [6] guarantees that working nodes make identical membership changes at the same local clock times. To achieve this strong property, the algorithm requires synchronized clocks and a form of atomic broadcast. Bounded correctness can be achieved without clock synchronization and does not require any special communication mechanisms. In addition, the work of [6] does not derive lower bounds on state holding time and latency and, therefore, does not address the limits of dynamic behavior nor the optimality of the presented algorithm.

Another related area is that of failure detection. Failure detectors are used to solve higher-level problems such as consensus and atomic broadcast in asynchronous and partially-synchronous systems. To our knowledge, the latest work that considers failures and recoveries is [1]. In [1], existence of nodes that eventually are permanently working or permanently failed is assumed. Since we do not assume existence of such nodes in our model, all nodes are unstable in the terminology of [1]. In [1], working nodes do not distinguish between unstable and failed nodes. Hence, no evaluation is done of the minimum time an unstable node needs to be in a particular state so that its status is accurately tracked. Diagnosis done by unstable nodes on other unstable or permanently failed nodes is also not considered. Last, [1] considers only asynchronous completely-connected networks.

III. SYSTEM MODEL

In this section, we present some basic definitions used throughout the paper.

1) Communication Model

Diagnosis algorithms can use either unicast or multicast communication. We assume generic parameters that could apply to either type of communication. We also assume a *synchronous* system in which the communication delay is bounded. This is an implicit assumption in all prior work on distributed diagnosis.

Definition 1. *The send initiation time, Δ_{send_init} , is the time between a node initiating a communication and the last bit of the message being injected into the network. This includes message set-up time on the node, any delay in accessing the communication medium, and the time to inject all of the message bits into the network. To simplify analysis, it is assumed that Δ_{send_init} is a constant.*

Definition 2. *The minimum and maximum message delays, Δ_{send_min} and Δ_{send_max} , are the minimum and maximum times, respectively, between the last bit of a message being injected into the network and the message being completely delivered at a working neighboring node.*

We assume that messages are encoded in such a way, e.g., using checksums, to enable incomplete messages to be detected and discarded. Hence, failures that occur on a sending node in the middle of a message transmission (prior to the last bit of the message, including the checksum, being injected into the network) appear as omissions at receiving nodes.

In a completely-connected network, there is a direct communication channel between every pair of nodes. It is not difficult to show that this is a requirement to be able to achieve bounded correctness with an arbitrary number of node failures.

2) Fault Model

We consider crash faults in nodes. The network delivers messages reliably. The crash fault assumption differs from traditional work on system-level diagnosis for which the fault model is not specified. However, the classical assumption that tests are perfect implies some class of easily-detectable faults. The crash fault assumption is necessitated by our use of heartbeat-based algorithms for diagnosis, which have been more commonly used in group membership algorithms. Hiltunen [12] shows how heartbeat-based algorithms can be transformed into test-based algorithms and vice versa. Using this transformation, our algorithms could be easily converted to ones that use explicit testing and the crash fault assumption could then be loosened.

Nodes can alternate between working correctly and being crashed in our model. Hence, the status of a node is modeled by a state machine with two states, *failed* and *working*.

Failed nodes do not send messages nor do they perform any computation. Working nodes execute faithfully the diagnosis procedure.

Definition 3. *The state holding time is the minimum time that a node remains in one state before transitioning to the other state.* It is important to note that, in completely-connected networks, Definition 3 is our model's only restriction on the timing of node failures and repairs. Since node failures and repairs are independent in this model, there are no restrictions on the number of nodes that are in the failed state at any one time nor on the number that can fail (or recover) at the same instant. This model is, therefore, considerably less restrictive than many of the models used. Since we will show later that a nonzero state holding time is required to solve the problems of interest and this is the only assumption on failure timing in our model, it is the least restrictive dynamic model possible for this problem.

3) Time and Clock Models

Since we are interested in dynamic failure situations in which failure and recovery timing is critical, it is imperative that the notion of time be well defined.

Definition 4. *Time that is measured in an assumed Newtonian time frame (which is not directly observable) is referred to as real time.*

Definition 5. *Time that is directly observable on a node's clock is referred to as clock time. The clock time of node X at real time t is denoted by $T_X(t)$.*

Definition 6. *While a node is in the working state, its clock experiences bounded drift. This means that if a node X is in the working state continuously during a real-time interval $[t_1, t_2]$, then for all real-time intervals $[u_1, u_2]$ subset $[t_1, t_2]$ $|[T_X(u_2) - T_X(u_1)] - (u_2 - u_1)| \leq p(u_2 - u_1)$, where $p \ll 1$ is the maximum drift rate of a clock.*

4) Algorithm Assumptions

We assume algorithms work by use of heartbeat messages, i.e., each node periodically initiates a round of message transmissions to other nodes in order to indicate that it is working.

Definition 7 *Assume an arbitrary node X initiates a round of heartbeat transmissions at real time t and remains in the working state indefinitely afterward. X will initiate another round of heartbeat transmissions no later than real time $t + (1 + p)\pi$, where π is the heartbeat period.*

We do not restrict algorithms to exchange status information only by heartbeat messages. For example, a node could send heartbeat messages to a subset of other nodes and then rely on those nodes to relay the information that it is working to the remaining nodes via ordinary (nonheartbeat) messages. We do assume, however, that heartbeats are the basic mechanism for a node to notify other nodes that it is working.

After a node recovers, it could initiate a round of heartbeats immediately after entering the working state or it could wait before doing so. If the node waits, however, it should not wait more than π in local clock time in order to maintain the heartbeat period. This leads to the following definition.

Definition 8 *The recovery wait time for an algorithm, denoted by $W \leq \pi$, is the local clock time for which the algorithm waits after entering the working state before initiating a round of heartbeat transmissions.*

5) Bounded Correctness

In a system that dynamically experiences failures and repairs and has nonzero communication delay, the view that any node has of the system at any time is, inevitably, out of date. To examine the limits of diagnosis algorithms, we consider what we believe are the weakest properties that any such algorithm should guarantee. Each working node should have timely information about the status of every other node, either working or failed, in the system. Any transition between the two states on a node is referred to as an *event*. The goal is for working nodes to learn about every event in the system as quickly as possible, to have their views of other nodes be out of date by only a bounded amount, and to not detect any spurious events.

Formally, we represent this goal by three properties which we collectively refer to as *bounded correctness*. Specification of one of these properties requires the following definition.

Definition 9. *A state held by a working node X for another node Y at time t is said to be T -valid if node Y was in the indicated state at some point during the interval $[t - T, t]$.*

Property 1: Bounded Diagnostic Latency. *Consider any event in the system that occurs at an arbitrary real time t . Any node that is in the working state continuously during the interval $[t, t + L]$ learns about the event and records the new state of the node that experienced the event by time $t + L$, where L is an algorithm-dependent bounded time referred to as the diagnostic latency of the algorithm.*

Property 2: Bounded Start-Up. *Consider the recovery of an arbitrary node X at real time t . If X remains in the working state continuously during the interval $[t, t + S]$, then at time $t + S$, X holds L -valid states for every other node in the system, where $S \geq L$ is an algorithm-dependent bounded time referred to as the start-up time of the algorithm.*

Property 3: Accuracy. *Consider an arbitrary working node X after its start-up time. Every state transition (working to failed or failed to working) recorded by X for an arbitrary node Y corresponds to an actual event that occurred on Y and no single event on Y causes multiple state transitions to be recorded on X .*

Taken together, these properties ensure that after a node recovers (or starts up for the first time), it will determine

valid state information about every other node in the system within bounded time and from that time on it will maintain a faithful record of events that occur on all nodes. Bounded Diagnostic Latency ensures that no events are missed, while Accuracy guarantees that no spurious events are recorded.

IV. COMPLETELY CONNECTED NETWORKS

(1) Limits on Algorithm Performance

In this section, we derive lower bounds on the diagnostic latency, start-up time, and state holding time achievable by any heartbeat-based diagnosis algorithm in completely-connected networks. The maximum time between two consecutive heartbeats arriving from a continuously working node at any other node in the system sets a limit on how early failed nodes can be identified by the absence of a heartbeat. This is specified in the following lemma.

Lemma 1. *Assume an arbitrary node Y is in the working state continuously for sufficiently long to send two consecutive heartbeats to another arbitrary node X . The maximum time between the two heartbeats arriving at X , $\Delta_{\text{heartbeat}}$, is $(1 + p)\pi + \Delta_{\text{send}_{\text{max}}} - \Delta_{\text{send}_{\text{min}}}$*

Proof. Suppose Y initiates the first heartbeat at time t . That heartbeat will arrive at X at time $t + \Delta_{\text{send}_{\text{init}}} + \Delta_{\text{send}_{\text{min}}}$ at the earliest. By Definition 8, Y will initiate its next heartbeat at time $t + (1 + p)\pi$ at the latest. This heartbeat will arrive at X at time $t + (1 + p)\pi + \Delta_{\text{send}_{\text{init}}} + \Delta_{\text{send}_{\text{min}}}$ at the latest. Subtracting the earliest and latest arrival times on X yields the maximum heartbeat interarrival time stated in the lemma.

Lemmas 2 and 3 provide the desired performance limits for any algorithm.

Lemma 2. *The diagnostic latency and start-up time of any heartbeat algorithm that achieves bounded correctness are both at least $(1 + 3\rho)\pi + 2(1 + \rho)\Delta_{\text{send}_{\text{max}}} - (1 + 2\rho)\Delta_{\text{send}_{\text{min}}}$*

Proof. The worst-case latency for the detection of node Y 's failure by node X occurs if Y fails immediately after initiating a heartbeat to X . If t represents the heartbeat initiation time, Y will fail at time $t + \Delta_{\text{send}_{\text{init}}}$ and the heartbeat will be received by X at time $t + \Delta_{\text{send}_{\text{init}}} + \Delta_{\text{send}_{\text{max}}}$ at the latest. Factoring in clock drift, X must then wait $(1 + \rho)\Delta_{\text{heartbeat}}$ time on its local clock without receiving a heartbeat before concluding that Y has failed. Subtracting the failure time from the latest detection time and simplifying yields a maximum failure detection latency of $(1 + 3\rho)\pi + 2(1 + \rho)\Delta_{\text{send}_{\text{max}}} - (1 + 2\rho)\Delta_{\text{send}_{\text{min}}}$.

We now analyze the latency in detecting a node's recovery. We assume that nodes initiate heartbeat transmission immediately after making a transition from the failed state to

the working state because this produces the shortest possible latency for recovery detection. The maximum delay before a working node receives a heartbeat from a recovered node is, therefore, $\Delta_{\text{send}_{\text{init}}} + \Delta_{\text{send}_{\text{max}}}$. Since $\Delta_{\text{send}_{\text{init}}} < \pi$, this is shorter than the failure detection latency derived above. Hence, the latency is equal to the failure detection latency.

For start-up time, we also need to consider the amount of time it takes after a node X returns to the working state before it determines an initial status for each other node in the system. The question here is, how long must node X wait without receiving a heartbeat from node Y before concluding that node Y is failed? The worst-case occurs if a heartbeat arrived from node Y just prior to node X 's recovery. In this case, Y must wait for clock time $(1 + \rho)\Delta_{\text{heartbeat}}$ before it is safe to conclude that node Y failed. In real time, this could take as long as $(1 + \rho)(1 + p)\Delta_{\text{heartbeat}}$. Since this is less than the minimum diagnostic latency and start-up time cannot be less than latency, the minimum start-up time is equal to the minimum latency derived above.

Lemma 3. *The state holding time for any heartbeat algorithm to achieve bounded correctness is at least $(1 + 4\rho)\pi/2 + (1 + 2\rho)(\Delta_{\text{send}_{\text{max}}} - \Delta_{\text{send}_{\text{min}}}) - \rho\Delta_{\text{send}_{\text{init}}}$*

Proof. We analyze the minimum state holding time as a function of the recovery wait time W and then minimize the function with respect to W to determine the minimum for any algorithm. As in the proof of Lemma 2, we ignore ρ^2 terms. Denote the minimum possible state holding time by SHT_{min} .

After a node Y recovers from the failed state, it must send a single heartbeat so that other nodes detect the recovery event. Hence,

$$\text{SHT}_{\text{min}} > (1 + \rho)W + \Delta_{\text{send}_{\text{init}}} \quad (1)$$

This is the minimum time a node must remain in the working state after making a transition into that state. Now, consider the minimum time a node must remain in the failed state in order for the transition into that state to be detected. Such a transition cannot be detected if the node returns to the working state and sends a new round of heartbeats as early as if it had never left the working state. The worst-case is if a node fails at time $t + \Delta_{\text{send}_{\text{init}}}$ immediately after successfully initiating a heartbeat to another node X and returns to the working state in time SHT_{min} . In this situation, the first heartbeat arrives at X at time $t + \Delta_{\text{send}_{\text{init}}} + \Delta_{\text{send}_{\text{max}}}$ at the latest. The second heartbeat will arrive at X at time $t + \Delta_{\text{send}_{\text{init}}} + \text{SHT}_{\text{min}} + (1 - \rho)W + \Delta_{\text{send}_{\text{init}}}\Delta_{\text{send}_{\text{min}}}$ at the earliest. If the difference between these arrival times is no greater than $(1 + 2\rho)\Delta_{\text{heartbeat}}$, then node X cannot distinguish this situation from one in which node Y remained in the working state for the entire interval. This yields

$$SHT_{\min} > (1 + 3\rho)\pi + 2(1 + \rho)(\Delta_{\text{send_max}} - \Delta_{\text{send_min}}) - \Delta_{\text{send_init}} - (1 - \rho)W \quad (2)$$

From (1) and (2), we have

$$SHT_{\min} > \max((1 + \rho)W + \Delta_{\text{send_init}}, (1+3\rho)\pi + 2(1+\rho) \times (\Delta_{\text{send_max}} - \Delta_{\text{send_min}}) - \Delta_{\text{send_init}} - (1-\rho)W). \quad (3)$$

As W increases, the right-hand side of (1) monotonically increases and the right-hand side of (2) monotonically decreases. Furthermore, since $\Delta_{\text{send_init}} \ll \pi$ in practice, the right-hand side of (1) is less than the right-hand side of (2) when $W = 0$. This means that the two expressions cross for some $W > 0$. Thus, the right-hand side of (3) is minimized when the right-hand sides of (1) and (2) are equal. Setting these two expressions to be equal and solving for W yields

$$W = (1 + 3\rho)\pi/2 + (1 + \rho)(\Delta_{\text{send_max}} - \Delta_{\text{send_min}}) - \Delta_{\text{send_init}} - (4)$$

and the lemma follows.

Note that it is theoretically possible for the right-hand side of (4) to be greater than π , which is outside the allowable range for W . However, in practice, all the parameters in (4) are small compared to π and the value for W that produces the smallest possible state holding time is approximately $\pi/2$. Even if the right-hand side of (4) is greater than π then from (3), SHT_{\min} in the range $0 \leq W \leq \pi$ will be greater than its absolute minimum and the lemma will still hold.

If we assume that ρ , $\Delta_{\text{send_max}}$, $\Delta_{\text{send_min}}$, and $\Delta_{\text{send_init}}$ are all much smaller than π , which is likely to be true in practice, then Lemma 3 gives the minimum state holding time as approximately $\pi/2$. This is somewhat counterintuitive in that it would seem that a node should remain in the failed state for at least π time before recovering in order to guarantee that its failure is observed by all working nodes. However, this analysis allows for the possibility that an algorithm forces nodes that recover to delay sending their heartbeats in order to extend the inter arrival times of their heartbeats on working nodes by enough to allow those nodes to observe the failure.

(2) An Optimal Algorithm for Bounded Correctness in Completely-Connected Networks

In this section, we present a new heartbeat-based algorithm, referred to as Algorithm HeartbeatComplete, for distributed diagnosis that provably minimizes diagnostic latency, start-up time, and state holding time in completely-connected networks.

4.2.1 Description of Algorithm HeartbeatComplete

The analysis of Section 4.1 shows that, if heartbeat messages are sent periodically by each working node to all other nodes with a period of π , then it is safe to declare a node to be failed when no heartbeat is received from it within a clock time of $(1 + \rho)\Delta_{\text{heartbeat}}$. The pseudocode for Algorithm HeartbeatComplete, which makes use of this fact is shown in Fig. 1.

Each node executing Algorithm HeartbeatComplete uses a broadcast mechanism to send heartbeat messages to all of its

neighbors in a single message requiring only one send initiation. Due to the assumption that faults are restricted to nodes, if a node successfully initiates a heartbeat, it will be

Fig. 1. Pseudocode for Algorithm HeartbeatComplete Upon entering the working state, node X executes:

```
Status[X] ← working;
SetSendHeartbeatTimer(W);
for Y = 0 to n - 1 do
  if Y ≠ X then
    Status [Y] = unknown;
    SetReceiveHeartbeatTimerY((1+ρ)Δheartbeat);
  endif;
endfor;
Upon receiving a heartbeat message from node Y, node X
executes:
Status[Y] ← working;
SetReceiveHeartbeatTimerY((1 + Δheartbeat);
Handlers execute when their corresponding timers expire:
Procedure SendHeartbeatTimerHandler
  broadcast heartbeat message to all neighbors;
  SetSendHeartbeatTimer(Π);
Procedure ReceiveHeartbeatTimerHandlerY
  Status [Y] ← failed;
```

received by all of its neighbors within time $\Delta_{\text{send_max}}$. However, it should be emphasized that there are no ordering requirements in this broadcast, e.g., it is neither causal nor atomic, so that broadcasts sent by two different nodes can be received in different orders on different nodes.

4.2.2 Algorithm Analysis

The following theorem states that Algorithm HeartbeatComplete achieves bounded correctness and characterizes its diagnostic latency, start-up time, and state holding time. The proof can be found in the appendix.

Theorem 1. *With $W \leq \pi$ and a state holding time of $\max((1 + \rho)W + \Delta_{\text{send_init}}, (1 + 3\rho)\pi + 2(1 + \rho)(\Delta_{\text{send_max}} - \Delta_{\text{send_min}}) - \Delta_{\text{send_init}} - (1 - \rho)W)$,*

Algorithm HeartbeatComplete achieves bounded correctness with diagnostic latency and start-up time equal to $\max((1 + 3\rho)\pi + 2(1 + \rho)\Delta_{\text{send_max}} - (1 + 2\rho)\Delta_{\text{send_min}}(1 + \rho)W + \Delta_{\text{send_init}} + \Delta_{\text{send_max}})$

Corollary 1. With

$$W = \min(\pi, (1 + 3\rho)\pi/2 + (1 + \rho)(\Delta_{\text{send_max}} - \Delta_{\text{send_min}}) - \Delta_{\text{send_init}})$$

Algorithm HeartbeatComplete achieves bounded correctness with minimum diagnostic latency and start-up time while requiring minimum state holding time.

Proof. With W as specified in the corollary, the value of the diagnostic latency and start-up time given by Theorem 1 becomes $(1 + 3\rho)\pi + 2(1 + \rho)\Delta_{\text{send_max}} - (1 + 2\rho)\Delta_{\text{send_min}}$, which, according to Lemma 2, is the minimum possible.

The expression for state holding time given in Theorem 1 is exactly the one derived in the proof of Lemma 3. This expression was shown to be minimized by the value of W specified in the corollary.

Note that Algorithm HeartbeatComplete does not specify any value for W . The above corollary gives the value of W that minimizes the diagnostic latency and start-up time while requiring minimum state holding time. If minimizing the minimum time a node must spend in the working state is critical, then W should be set to zero.

4.2.3 Message Cost of Algorithm HeartbeatComplete

Many systems that are logically completely-connected actually employ a bus or redundant bus structure. In such systems, HeartbeatComplete sends one heartbeat message per node per round, which is the minimum possible message cost. Even if buses are not used, a single broadcast message per node per round is generated. In networks with efficient broadcast support, e.g., Ethernet, this cost can still be low. In a complete network made entirely of point-to-point links, the algorithm would generate $n(n - 1)$ messages per round, which is higher than the best known algorithms for completely-connected networks; $2n$ ([5]) and $2n \log n$ ([8] and [9]). However, the cost of HeartbeatComplete still represents only one message per link per node per round in this case. Note that, to satisfy our model, any type of logical completely-connected network must reliably deliver all messages, and so redundant buses, redundant links, or reliable broadcast support must be provided.

Complexity Analysis

Algorithm	Message Count (Msgs)	Diagnosis Latency (Testing Round)
ADSD	n	n
HeartBeat	$n(n-1)$	≤ 1

V. CONCLUSION

Heartbeat Algorithms works over a different approach where diagnostic latency is given higher priority than no of tests performed, and it has achieved the diagnostic latency of approximately half-testing round. This algorithm can be used to diagnose a system with faulty environment i.e. network fault detection & monitoring

VI. FUTURE ENHANCEMENTS

1. This work can be extended by considering different types of faults

2. Fault detection of nodes from network can be considered.

REFERENCES

- [1] Preparata F., Metze G., and Chien R., "On the connection assignment problem of diagnosable systems", IEEE Trans. Elect. Comput. EC-16, 6 (Dec.), pp. 848-854, 1967.
- [2] Kuhl J. and Reddy S., "Distributed fault-tolerance for large multiprocessor systems", In Proceedings of the 7th Annual Symposium on Computer Architecture, pp. 23-30, 1980
- [3] R. Bianchini and R. Buskens, "Implementation of On-Line Distributed System-Level Diagnosis Theory", IEEE Trans. Computers, vol. 41, pp. 616-626, May 1992.
- [4] E.P. Duarte Jr. and T. Nanya, "A Hierarchical Adaptive Distributed System-Level Diagnosis Algorithm", IEEE Trans. Computers, vol. 47, pp. 34-45, Jan. 1998.
- [5] E.P. Duarte Jr., A. Brawerman, and L.C.P. Albini, "An Algorithm for Distributed Hierarchical Diagnosis of Dynamic Fault and Repair Events", Proc. Seventh Int'l Conf. Parallel and Distributed Systems, pp. 299-306, 2000.
- [6] Subbiah A., and Douglas M., "Distributed Diagnosis in Dynamic Fault Environments", IEEE Transactions On Parallel And Distributed Systems, Vol.15, pp. 453-467, May 2004
- [7] Subbiah A., "Design and Evaluation of a Distributed Diagnosis Algorithm for Arbitrary Network Topologies in Dynamic Fault Environments", MS thesis, Georgia Inst. of Technology, http://www.ece.gatech.edu/~arun/ms_thesis.pdf, 2001.
- [8] M.K. Aguilera, W. Chen, and S. Toueg, "Failure Detection and Consensus in the Crash-Recovery Model," Distributed Computing, vol. 13, no. 2, pp. 99-125, 2000.
- Bagchi and S.L. Hakimi, "An Optimal Algorithm for Distributed System Level Diagnosis," Proc. Digest of the 21st Int'l Symp. Fault Tolerant Computing, pp. 214-221, 1991.
- [9] M. Barborak, M. Malek, and A.T. Dahbura, "The Consensus Problem in Fault Tolerant Computing," ACM Computing Surveys, vol. 25, pp. 171-220, June 1993.
- [10] D. Blough and H. Brown, "The Broadcast Comparison Model for On-Line Fault Diagnosis in Multicomputer Systems: Theory and Implementation," IEEE Trans. Computers, vol. 48, pp. 470-493, May 1999.
- [11] Andrew S. Tanenbaum, "Computer Networks" 3rd Edition, Prentice-Hall November 2002
- [12] R.P. Bianchini and R. Buskens, "An Adaptive Distributed System- Level Diagnosis Algorithm

- and Its Implementation,” *Proc. FTCS-21*, pp. 222-229, 1991.
- [13] F. Preparata, G. Metze, and R.T. Chien, “On The Connection Assignment Problem of Diagnosable Systems,” *IEEE Trans. Electronic Computers*, vol. 16, pp. 848-854, 1968.
- [14] S.L. Hakimi and A.T. Amin, “Characterization of Connection Assignments of Diagnosable Systems,” *IEEE Trans. Computers*, vol. 23, pp. 86-88, 1974.
- [15] S.L. Hakimi and K. Nakajima, “On Adaptive System Diagnosis” *IEEE Trans. Computers*, vol. 33, pp. 234-240, 1984.
- [16] J.G. Kuhl, and S.M. Reddy, “Distributed Fault-Tolerance for Large Multiprocessor Systems,” *Proc. Seventh Ann. Symp. Computer Architecture*, pp. 23-30, 1980.
- [17] J.G. Kuhl and S.M. Reddy, “Fault-Diagnosis in Fully Distributed Systems,” *Proc. FTCS-11*, pp. 100-105, 1981.
- [18] S.H. Hosseini, J.G. Kuhl, and S.M. Reddy, “A Diagnosis Algorithm for Distributed Computing Systems with Failure and Repair,” *IEEE Trans. Computers*, vol. 33, pp. 223-233, 1984.
- [19] R.P. Bianchini, K. Goodwin, and D.S. Nydick, “Practical Application and Implementation of System-Level Diagnosis Theory,” *Proc. FTCS-20*, pp. 332-339, 1990.
- [20] C.-L. Yang and G.M. Masson, “Hybrid Fault-Diagnosability with Unreliable Communication Links,” *Proc. FTCS-16*, pp. 226-231, 1986.
- [21] S.Rangarajan, A.T. Dahbura, and E.A. Ziegler, “A Distributed System-Level Diagnosis Algorithm for Arbitrary Network Topologies,” *IEEE Trans. Computers*, vol. 44, pp. 312-333, 1995.
- [22] M. Stahl, R. Buskens, and R. Bianchini, “Simulation of the Adapt On-Line Diagnosis Algorithm for General Topology Networks,” *Proc. IEEE 11th Symp. Reliable Distributed Systems*, Oct. 1992.
- [23] G. Masson, D. Blough, and G. Sullivan, “System Diagnosis,” *Fault-Tolerant Computer System Design*, D.K. Pradhan, ed. Prentice Hall, 1996.
- [24] E.P. Duarte Jr. and T. Nanya, “Multi-Cluster Adaptive Distributed System-Level Diagnosis Algorithms,” *IEICE Technical Report FTS 95-73*, 1995.
- [25] M. Malek, and J. Maeng, “Partitioning of Large Multicomputer Systems for Efficient Fault Diagnosis,” *Proc. F7CS-12*, pp. 341-348, 1982.
- [26] E.P. Duarte Jr., E Mansfield, T. Nanya, and S Noguchi, “Non- Broadcast Network Fault-Monitoring Based on System-Level Diagnosis,” *Proc. IFIP/IEEE IM’97*, pp. 597

Temperature Variation on Rough Actor-Critic Algorithm

P.K.Pandey, D.Tiwari

Department of Computer Science, Jaypee Institute of Engineering Technology, Guna

Email: { pandey02_bit ,deepshikha1213 }@rediffmail.com

Abstract- The problem considered in this paper is how to guide Rough actor-critic learning based on temperature variation. The solution to this problem stems from a modification of the action-preference model that includes a temperature adjustment factor. We consider modification of the action-preference model using average rough coverage derived from approximation spaces. Approximation spaces provide a ground for deriving pattern-based behaviors as well as information granules that can be used to influence the policy structure of an actor in a beneficial way. This paper includes the results of a recent study of swarm behavior by collections of biologically-inspired bots carried out in the context of an artificial ecosystem. This ecosystem has an ethological basis that makes it possible to observe and explain the behavior of biological organisms that carries over into the study of actor-critic learning by interacting robotic devices. The proposed approach results in new forms of Rough actor-critic learning (RACL), i.e., rough coverage temperature variation ACL. The contribution of this article is a framework for actor-critic learning defined in the context of temperature variation and approximation spaces.

Keywords: Actor-critic learning, approximation space, rough sets, temperature variation.

I. INTRODUCTION

Swarms learn by evaluating their actions. In reinforcement learning, the choice of an action is based on estimates of the value of a state and/or the value of an action in the current state using some form of an update rule (see, e.g. [7,8,10]). A swarm learns the best action to take in each state by maximizing a reward signal obtained from the environment. Two different forms of Rough Actor-Critic method are considered in this article as a part of study of reinforcement learning in real-time by a swarm. First, a conventional Rough Actor-Critic method is considered, where a critic evaluates whether things have gotten better or worse than expected as a result of an action selection in the previous state. A temporal difference (TD) error term δ is computed by the critic to evaluate an action previously selected. An estimated action preference in the current state

is than determined using δ . Swarm actions are generated using the Gibbs softmax method [8]. In the study of swarm behavior of multiagent systems such as systems of cooperating bots, it is help to consider ethological methods

[1, 3], Where each proximate cause (stimulus) usually has more than one possible response. Swarm actions with lower TD error tend to be favored. The second form of Rough Actor-Critic method is defined in context of environmental factor.

In this new form of Rough Actor-Critic method we have include temperature adjustment Factor (TAF). The contribution of this article is to introduce a Temperature Rough Actor-Critic method (TRAC) to adjust the preference of selecting the particular action. This form of actor critic method utilizes what is known as reference reward, which is action specific and the next action not only depends on previous average reward but also on some environmental factor like temperature. This paper has the following organization. A brief introduction about the swarm-bots is given in section II. Section III represents a basic of Rough sets. Section IV represents the Approximation spaces. Effect of temperature on swarm-bots is given in section V. Actor critic reinforcement learning by a Swarm is considered using a conventional approach in section VI and the temperature rough actor critic method in section VI (B).

II. WHAT IS SWARM-BOT?

Swarm-bots is self-assembling and self-organizing robot colony composed of number (30-35) of smaller devices, called s-bots [12]. Each s-bot is a fully autonomous mobile robot capable of performing basic tasks such as autonomous navigation, perception of environment and grasping of objects. In addition to these features, one s-bot is able to communicate with other s-bots and physically connect to them in flexible ways, thus forming a so-called swarm-bot. Such a robotic entity is able to perform tasks in which a single s-bot has major problems, such as exploration, navigation, and transportation of heavy objects on very rough terrain (this hardware structure is combined with a distributed adaptive control architecture loosely inspired upon ant colony behaviors).



Fig.1: Swarm-bot

III. ROUGH SET

The rough set approach introduced by Zdzislaw Pawlak provides a ground for concluding to what degree a set of equivalent behaviors is a part of a set of behaviors representing a standard. For computational reasons, a syntactic representation of knowledge is provided by rough sets in the form of data tables. Informally, a data table is represented as a collection of rows each labeled with some form of input and each column is labeled with the name of an attribute (feature) that computes a value using the row input. Formally, a data (information) table IS is represented by a pair (U, A) , where U is a non-empty, finite set of elements and A is a non-empty, finite set of attributes (features), where $a: U \rightarrow V_a$ for every a belongs to A . For each B is subset of A , there is associated an equivalence relation $Ind_{IS}(B)$ such that $Ind_{IS}(B) = \{(x, x_0) \in U^2 | a \in B, a(x) = a(x_0)\}$. Let $U/Ind_{IS}(B)$ denote a partition of U , and let $B(x)$ denote a set of B -indiscernible elements containing x . $B(x)$ is called a block, which is in the partition $U/Ind_{IS}(B)$. For X is subset of U , the sample X can be approximated from information contained in B by constructing a B -lower and B -upper approximation denoted by B_*X and B^*X , respectively, where $B_*X = \{x \in U | B(x) \text{ is subset of } X\}$ and $B^*X = \{x \in U | B(x) \cap X \neq \emptyset\}$. The B -lower approximation B_*X is a collection of sample elements that can be classified with full certainty as members of X using the knowledge represented by attributes in B . By contrast, the B -upper approximation B^*X is a collection of sample elements representing both certain and possible uncertain knowledge about X . Whenever B_*X is a proper subset of B^*X , i.e., $B_*X \subset B^*X$, the sample X has been classified imperfectly, and is considered a rough set.

IV. APPROXIMATION SPACES

The primary notions of the theory of rough sets are the approximation space and lower and upper approximations of a set. The approximation space is a classification of the domain of interest into disjoint categories. The classification formally represents our knowledge about the domain, i.e. the knowledge is understood here as an ability to characterize all classes of the classification, for example, in terms of

features of objects belonging to the domain. Objects belonging to the same category are not distinguishable, which means that their membership status with respect to an arbitrary subset of the domain may not always be clearly definable. This fact leads to the definition of a set in terms of lower and upper approximations. The lower approximation is a description of the domain objects which are known with certainty to belong to the subset of interest, whereas the upper approximation is a description of the objects which possibly belong to the subset. Any subset defined through its lower and upper approximations is called a rough set. It must be emphasized that the concept of rough set should not be confused with the idea of fuzzy set as they are fundamentally different, although in some sense complementary notions. An equivalence relation induces a partitioning of the universe. These partitions can be used to build new subsets of the universe. Subsets that are most often of interest have the same value of the outcome attribute.

A approximation space is a system of $GAS = (U, I, \nu)$ Where

- U is a non-empty set of objects, and $P(U)$ is the power set of U .
- $I: U \rightarrow P(U)$ is an uncertainty function.
- $\nu: P(U) \times P(U) \rightarrow [0, 1]$ denotes rough inclusion

The uncertainty function I defines a neighborhood of every sample element x belonging to the universe U . The rough inclusion function ν computes the degree of overlap between two subsets of U . Let $P(U)$ denote the power set of U . In general, rough inclusion $\nu: P(U) \times P(U) \rightarrow [0, 1]$ can be defined in terms of the relationship between two sets where

- $\nu(X, Y) = \frac{|X \cap Y|}{|Y|}$ if $Y \neq \emptyset$
- $\nu(X, Y) = 1$, otherwise
-

For any X, Y is subset of U . In the case where X is subset of Y , then $\nu(X, Y) = 1$. The minimum inclusion value $\nu(X, Y) = 0$ is obtained when $X \cap Y = \emptyset$ (i.e., X and Y have no elements in common). In a hierarchical model of an intelligent system, one or more approximation spaces would be associated with each layer.

V. EFFECT OF TEMPERATURE ON SWARM-BOT

Temperature is the major environmental factor that causes the problem in taking the particular action by swarm-bot. When the temperature crosses from a certain boundary value swarm-bot functioning is affected. Swarm-bots are functioning well in between certain desirable temperature. When these swarm-bot have to work on high temperature area like equator or in some very low temperature like poles than its time to take a particular action increases. Temperature has direct effect on its processor and sensors. Temperature reduces processor life and degrades the performance of sensor due to the thermal effect. Swarm-bots are trained by various algorithms like greedy method, actor

critic method etc. Here we are using actor-critic method to operate the swarm-bot. In the previous actor-critic method the environmental factor is not included but now we are including the temperature in preference formula (which is used in deciding a particular action) and show its effect on action taking priority. To reduce the impact causes by temperature we are using temperature adjustment factor (TAF).

Ambient temperature affects Swarm-bot temperature. With an ordinary heat sink a normal swarm-bot cannot able to reduce such effect that is generated by the excess temperature. Swarm-bot life and temperature are inversely related - the higher the temperature, the lower the swarm-bot Life. Temperature affects the processor of swarm-bot. This holds true for all integrated circuits that is used in swarm-bot - *heat is the enemy!*

$$\text{Swarm-bot processors Life} = \text{Normal Life Hours} / [((273 + \text{New Temp}) / (273 + \text{Normal Temp}))^t] \quad (1)$$

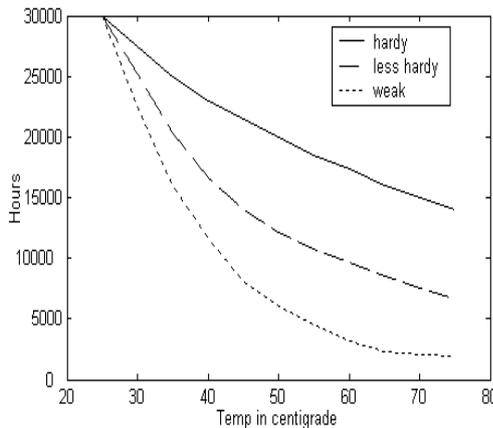


Fig.2: Graph Shows the Life of Swarm-bots Processor

Here Normal Life Hours means that the swarm-bots has some expected life at Normal Temp, say 30,000 hours. If the swarm-bot is run at a higher temperature, its processor Life is degraded by the ratio of the New Temp to Normal Temp raised to the power of "t". The number 273 is a constant in the formula. "t" is determined by real life temperature tests - The Swarm processor is run at a constant 60 C, then 70 C, and the resultant decrease in CPU life determines t. Let's plug in some numbers and see what we get. [13]

The Graph shows the relationship for the three cases outlined above, with "Hardy" on the top line and "weak" the bottom line. As you can see, depending on the bot's "hardiness". Heat does degrade swarm-bot life in a measurable way.

VI. ACTOR-CRITIC METHOD

Actor-critic methods are temporal difference (TD) methods with a separate memory structure to explicitly represent the

policy independent of the value function [7,8] used. The policy structure is known as the *actor*, because it is used to select actions, and the estimated value function is known as the *critic*, because it criticizes the actions made by the actor. Learning is always on-policy: the critic must learn about and critique the actor is currently following. The critique takes the form of a TD error. This scalar signal is the sole output of the critic and drives all learning in both actor and critic, as suggested by Figure 4 Actor-critic methods are the natural extension of the idea of reinforcement comparison methods to TD learning and to the full reinforcement-learning problem. Typically, the critic is a state-value function. After each action selection, the critic evaluates the new state to determine whether things have gone better or worse than expected. That evaluation is the TD error δ .

$$\delta_t = r_{t+1} + \gamma V(s_{t+1}) - V(s_t) \quad (2)$$

Where V does the critic implement the current value function. This TD error can be used to evaluate the action just selected, the action a_t taken in states s_t . If the TD error is positive, it suggests that the tendency to select a_t should be strengthened

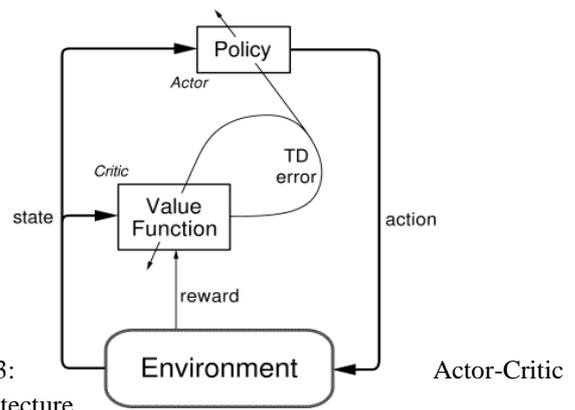


Fig.3: Actor-Critic Architecture

for the future, whereas if the TD error is negative, it suggests the tendency should be weakened.

$$\pi_t(s, a) = \Pr\{a_t = a \mid s_t = s\} = e^{P(s, a)} / \sum_b e^{P(s, b)}$$

Where the $P(s, a)$ are the values at time t of the modifiable policy parameters of the actor, indicating the tendency to select (preference for) each action a when in each state s . Then the strengthening or weakening described above can be implemented by increasing or decreasing $P(s_t, a_t)$, for instance, by

$$P(s_t, a_t) \leftarrow P(s_t, a_t) + \beta * \delta \quad (4)$$

Where β is another positive step size parameter.

A. Rough Actor-Critic Algorithms:

This algorithm is used for the learning process in swarm-bot. The main rough actor-critic method is described here in which we are not considering the temperature as an environmental factor.

Initialize, for all $s \in S, a \in A(s)$:
 $p(s, a) \leftarrow 0$
 $\pi(s, a) \leftarrow e^{p(s, a)} / [\sum_{b=1}^{A(s)} e^{p(s, a)}]$
 $\text{Count}(s) \leftarrow 0$
 Repeat forever:
 Initialize s
 Repeat (for each step of episode):
 Choose a from s using $\pi(s, a)$
 Take action a , observe r, s'
 $\text{Count}(s) \leftarrow \text{Count}(s) + 1$
 $V(s) \leftarrow V(s) + 1/\text{Count}(s) [r - V(s)]$
 $\delta = r + \gamma V(s') - V(s)$
 $p(s, a) \leftarrow p(s, a) + \beta(\delta - r')$
 $\pi(s, a) \leftarrow e^{p(s, a)} / [\sum_{b=1}^{A(s)} e^{p(s, a)}]$
 $s \leftarrow s'$
 Until s is terminal
 Calculate r' as follows (At the end of each episode):
 Start with ethogram table $\text{DTsbot} = (\text{Ubeh}, A, d)$.
 Discretize feature values in DTsbot .
 Construct approximation space where
 B is subset of A
 $D = \{x \mid d(x) = 1\}$
 $B_a(x) = \{y \in \text{Ubeh} \mid a(x) = a(y) a \in B\}$
 Lower Approximation: $B * D = \{x \mid B_a(x) \text{ is subset of } D\}$
 if $B_a(x)$ is subset of $B * D$, then
 Compute rough inclusion value for each $B_a(x)$
 $r' = \sum [v(B_a(x), B * D)] / n$

B. Temperature Rough Actor-Critic Method (TAC)

This section introduces the Temperature Rough Actor-Critic Method. In fact, common variation includes additional factors varying the amount of credit assigned to action taken. The other factor made affects on the calculation of preference. We adjusted the preference formula by applying the TAF on it. In Temperature Rough Actor-Critic method the preference can be calculated as follows

$$p(s, a) \leftarrow p(s, a) + [[\beta(\delta - r')]/t] * \text{TAF}$$

Where r' is reminiscent of the idea of a reference reward used during reinforcement comparison, t is the temperature and TAF is the Temperature adjustment factor. In Temperature Rough actor-critic method (TRAC) preference of taking a particular action depend upon temperature. To reduce the effect of temperature we are using the temperature adjustment factor. It may be the value which is taken by a particular sensor device. This TAF minimizes the effects on probability, which is generated by temperature. The new TRAC Algorithm is as follows.

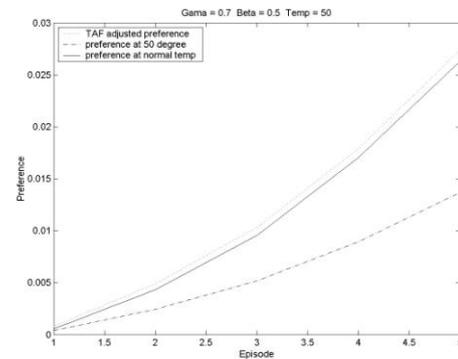
Initialize, for all $s \in S, a \in A(s)$:
 $p(s, a) \leftarrow 0$
 $\pi(s, a) \leftarrow e^{p(s, a)} / [\sum_{b=1}^{A(s)} e^{p(s, a)}]$
 $\text{Count}(s) \leftarrow 0$
 Repeat forever:
 Initialize s
 Repeat (for each step of episode):
 Choose a from s using $\pi(s, a)$

Take action a , observe r, s'
 $\text{Count}(s) \leftarrow \text{Count}(s) + 1$
 $V(s) \leftarrow V(s) + 1/\text{Count}(s) [r - V(s)]$
 $\delta = r + \gamma V(s') - V(s)$
 $p(s, a) \leftarrow p(s, a) + [[\beta(\delta - r')]/t] * \text{TAF}$
 $\pi(s, a) \leftarrow e^{p(s, a)} / [\sum_{b=1}^{A(s)} e^{p(s, a)}]$
 $s \leftarrow s'$
 Until s is terminal
 Calculate r' as follows (At the end of each episode):
 Start with ethogram table $\text{DTsbot} = (\text{Ubeh}, A, d)$.
 Discretize feature values in DTsbot .
 Construct approximation space where
 B is subset of A
 $D = \{x \mid d(x) = 1\}$
 $B_a(x) = \{y \in \text{Ubeh} \mid a(x) = a(y) a \in B\}$
 Lower Approximation: $B * D = \{x \mid B_a(x) \text{ is subset of } D\}$
 if $B_a(x)$ is subset of $B * D$, then
 Compute rough inclusion value for each $B_a(x)$
 $r' = \sum [v(B_a(x), B * D)] / n$

VII. CONCLUSION

This paper presents an ethological approach to observing reinforcement learning by swarms of cooperating agents in an ecosystem testbed that is being used to design line-crawling bots that forms swarms to carry out inspection of various power system structures. The result reported in this paper shows the effect of temperature on swarm that learn. This effect is minimized by the temperature adjustment factor that is included in the rough actor-critic algorithm.

The main idea behind this paper to represent the actor critic algorithm included with temperature adjustment factor (TAF) in the preference field of algorithm. The result of this paper shown by graph in which solid line shows the preference of general rough actor-critic algorithm at normal temp, dashed line shows the preference at 40 or 50 degree temperature and dotted line represent the adjusted preference with TAF. To measure the temperature we need some sensors according to that we can change the value of TAF



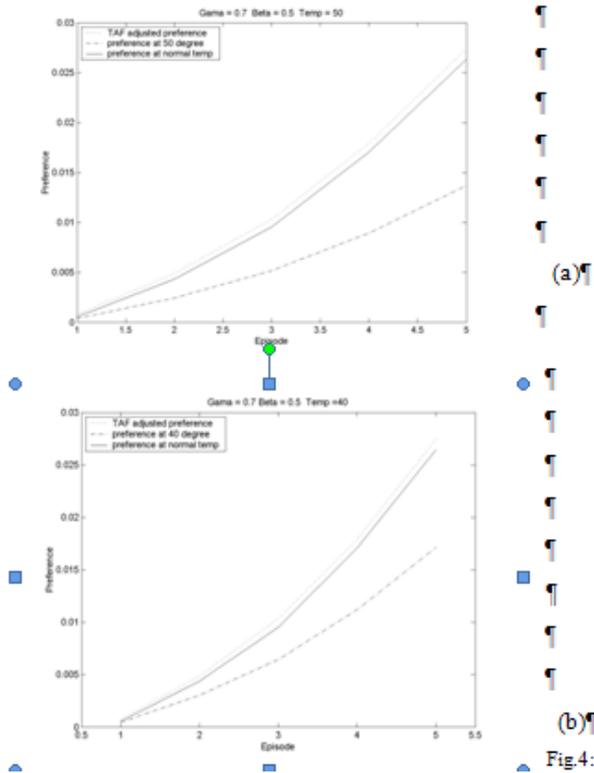


Fig.4: Graph shows the preference

(a) at 50 degree temp

(b) at 40 degree temp

Ultimately, it is important to consider ways to evaluate the behavior of an intelligent system as it unfolds. Behavior decision table constantly change during the life of an intelligent system because of changing temperature and changing rewards of corresponding action responses. As a result there is a need for a co-operating system of agents to gain measure and share knowledge about changing behavior patterns. Part of future work may be included for the other environmental factor like Wind, Snowfall and Rainy season. According to that we can make changes in the actor-critic algorithms.

REFERENCES

- [1] J.F.Peters. Rough Ethology: Towards a biologically inspired study of collective behavior in Intelligent System with Approximation Spaces, Transactions on Rough Sets- III, LNCS 3400, 153-174, 2005, Springer-Verlag Berlin Heidelberg 2005.
- [2] J.F.Peters and Christopher Henry. "Reinforcement Learning in Swarms that Learn". In proceeding of the 2005/IEEE/WIC/ACM International Conference on Intelligent Agent Technology, edited by A.Skowron, Compiegne University of Technology, France 19-22 sept 2005, 400-456.
- [3] J.F. Peters, C. Henry, S. Ramanna, Rough Ethograms: Study of Intelligent System Behavior. In: M.A.Klopotek, S. Wierchori, K.Trojanowski(Eds), New Trends in Intelligent Information Processing and Web Mining (IIS05), Gdańsk, Poland, June 13-16 (2005), 117-126.
- [4] J.F. Peters, C. Henry, S. Ramanna, Reinforcement learning with pattern-based rewards. in proceeding of forth International IASTED Conference. Computational Intelligence(CI 2005) Calgary, Alberta, Canada, 4-6 July 2005, 267-272
- [5] J.F. Peters, T.C. Ahn, M. Borkowski, V. Degtyaryov, S. Ramanna, Line-crawling robot navigation: A rough neurocomputing approach. In: C. Zhou, D. Maravall, D. Ruan (Eds.), Autonomous Robotic Systems. Studies in Fuzziness and Soft Computing 116 (Heidelberg: Springer-Verlag, 2003) 141-164
- [6] J.F. Peters, Approximation spaces for hierarchical intelligent behavioral system models. In: B.D. Keplicz, A. Jankowski, A. Skowron, M. Szczuka (Eds.), Monitoring, Security and Rescue Techniques in Multiagent Systems, Advances in Soft Computing, (Heidelberg: Physica-Verlag, 2004) 13-30
- [7] L.P. Kaelbling, M.L. Littman, A.W. Moore, Reinforcement learning: A survey Journal of Artificial Intelligence Research, 4, 1996, 237-285.
- [8] R.S. Sutton, A.G. Barto, and Reinforcement Learning: An Introduction (Cambridge, MA: The MIT Press, 1998).
- [9] E. Bonabeau, M. Dorigo, G. Theraulaz, Swarm Intelligence. From Natural to Artificial Systems, (UK: Oxford University Press, 1999).
- [10] C. Gaskett, Q-Learning for Robot Control. Ph.D.Thesis, Supervisor: A.Zelinsky, Department of Systems Engineering, The Australian National University, 2002.
- [11] R. Gross, M. Dorigo, Cooperative transport of objects of different shapes and sizes. In M. Dorigo, M. Birattari, C. Blum, L.M. Gambardella, F. Mondada, T. Stutzle(Eds.), Ant Colony Optimization and Swarm Intelligence Lecture Notes in Computer Science, 3172, 106-117, 2004.
- [12] Francesco Mondada, Giovanni C. Pettinaro, Andre Guignard, Swarm bot: A New Distributed Robotic Concept, Autonomus System Lab(LSA), EPFL-STI-I2S, Lausanne, Switzerland
- [13] Joe Citarella, Overclocking's Impact on CPU Life (www.overclockers.com)

Evaluation of Efficient Web Caching and Prefetching Technique for Improving the Proxy Server Performance

¹G.N.K.Suresh Babu and ²S.K.Srivatsa

1 Apollo Engineering College, Chennai, Tamil Nadu, India

2 St.Joseph College of Engineering, Chennai, Tamil Nadu, India

Abstract- Web caching is a temporary storage of web objects for later retrieval. This paper describes the improvement of web cache to develop a utility to share internet from single connection to a large network. This paper differs from the existing one as the former uses the data structures and databases for the storage and retrieval of the web pages. In this paper the usage of Randomized algorithms is implemented. Using this algorithm we replace the document in web cache in a effective manner. The randomized algorithms are used to clear the local folder, which has all the web pages saved in "cache" extension. Based on our analysis, we proposed a new algorithm which takes recently, frequency, perfect-history, and document size into account for web cache optimization. Considering all the above mentioned parameters the algorithm is proven to be efficient than its predecessors. This paper tries to resolve the problems in the existing system and provides improved algorithm, by which the performance of the web cache is improved.

Keywords:

Web cache, Randomized algorithm, Page replacement, LRU, proxy cache, Bandwidth.

I. INTRODUCTION

"The World Wide Web is the universe of network-accessible information, an embodiment of human knowledge." The recent increase in popularity of the World Wide Web has led to a considerable increase in the amount of traffic over the Internet. As a result, the web has now become one of the primary bottlenecks to network performance. When a user requests objects, which are connected to a server on a slow network link, there is considerable latency, which can be noticed at the client end. Transferring the object over the network lead to increase in the level of traffic. Increase in traffic will reduce the bandwidth for competing requests and increase latencies for other users. In order to reduce access latencies, it is desirable to store copies of popular objects closer to the user. Consequently, Web Caching has become an increasingly important topic. After a significant amount of research to reduce the noticeable response time perceived by users, it is found that Web caching and Web Prefetching are two important

techniques to this end. This paper provides an environment containing a number of ready-made options like cache, log file, error checking, connection pooling, etc

II. WEB CACHING

Web caching is the emerging technology in web. In web caching if the client is requesting a page from server it will fetch from the server and will give response to the server. A web cache sits between web server and a client and watches request for web pages. It caches web documents for serving previously retrieved pages when it receives a request for them. According to the locations where objects are cached, web caching technology can be classified into three categories, i.e., client's browser caching, client-side proxy caching, and server-side proxy caching.

In client's browser caching, web objects are cached in the client's local disk. If the user accesses the same object more than once in a short time, the browser can fetch the object directly from the local disk, eliminating the repeated network latency. However, users are likely to access many sites, each for a short period of time. Thus, the hit ratios of per-user caches tend to be low.

In client side proxy caching, objects are cached in the proxy near the clients to avoid repeated round-trip delays between the clients and the origin Web servers. To effectively utilize the limited capacity of the proxy cache, several cache replacement algorithms are proposed to maximize the delay savings obtained from cache hits. Such advanced caching algorithms differ from the conventional ones (e.g., LRU or LFU algorithms) in their consideration of size, fetching delay, reference rate, invalidation cost, and invalidation frequency of a Web object. Incorporating these parameters into their designs, these cache replacement algorithms show significant performance improvement over the conventional ones. In addition, cooperative caching architectures, proposed in enable the participating proxies to share their cache content with one another. Since each participating proxy can seek for a remote cache hit from other participating proxy's cache, the overall hit ratio can be further improved.

In server-side Web caching and content distribution networks (CDN) are recently attracting, an increasing amount of attention. It is noted that, as the Web traffic grows exponentially, overloaded Web servers become the sources of the prolonged response time. Server-side Web

caching, which distributes routes the user requests to the proper server-side proxies, it is able to release the Web server's load. Server side proxy caching will shorten the user perceived response time.

III. PROXY CACHING

Caching can be implemented at various points in the network. The best method among this is to have a cache in the Web server itself. Further, it is increasingly common for a university or corporation to implement specialized servers in the network called Caching Proxies. Such proxies act as agents on behalf of the client in order to locate a cached copy of an object if possible. There are different types of proxy server based on FTP, HTTP, and SMTP and so on. They are FTP Proxy Server which relays and caches FTP Traffic. HTTP Proxy Server which has one way request to retrieve Web Pages and Socks Proxy Server is the newer protocol to allow relaying of far more different types of data, whether TCP or UDP. NAT Proxy Server which works differently from other servers, it allows the redirection of all packets without a program having to support a Proxy Server. SSL Proxy Server which is an extension to the HTTP Proxy Server which allows relaying of TCP data similar to a Socks Proxy Server.

Furthermore, a Proxy Server can be split into another two Categories:

- Anonymous
- Transparent.

1) *Anonymous*

An Anonymous Proxy Server blocks the remote computer from knowing the identity of the computer using the Proxy Server to make requests. Anonymous Proxy Servers can further be broken down into two more categories, Elite and Disguised. An Elite Proxy Server is not identifiable to the remote computer as a Proxy in any way.

2) *Transparent*

A Transparent Proxy Server tells the remote computer the IP Address of the Computer. This provides no privacy. A Disguised Proxy Server gives the remote computer enough information to let it know that it is a Proxy, however it still does not give away the IP of the computer it is relaying information for.

IV. WEB PREFETCHING

In Web prefetching scheme the proxy itself will give the response to the clients if the web page requested is present in the proxy itself. Several algorithms based on Markov models and web mining techniques are proposed to derive prefetching rules from the server's access log.

Solution for the problem

In this paper an innovative cache replacement algorithm (i.e.) randomized algorithm is proposed. Randomized

algorithm combines the benefit of both utility, based schemes and RR (Round Robin) schemes and it avoids the need for data structures. The utility function assigns to each page a value based on recentness of use and frequency of use, size of page, cost of fetching and RR scheme would replace the least recently used web documents. These data will be evacuated only when it crosses the expiry time

To reduce the latency time and to increase the memory capacity and processing power the proxy server is designed in which data and images are stored separately. The proxy server can be connected to number of clients.

V. WEB CACHING WITH PROXIES

After a serious research in caching technologies it was found that Web caching with proxies is the efficient technology. Web proxy will be between the server and client and will serve for web page request.

Cache deployment options

There are three main cache deployment choices:

- Near the content provider (provider-oriented)
- Near the content consumer (consumer-oriented)

At strategic points in the network, based on user access patterns and network topology.

Provider-oriented deployment

In provider oriented deployment method caches positioned near or maintained by the content provider, as in reverse proxy and push caching, improve access to a logical set of content. This type of cache deployment can be critical to delay-sensitive content such as audio or video. Positioning caches near or on behalf of the content provider allows the provider to improve the scalability and availability of content, but is obviously only useful for that specific provider. Any other content provider must do the same thing.

Consumer-oriented deployment

Positioning caches near the client, as in client side proxy caching has the advantage of leveraging one or more caches to a user community. If those users tend to access the same kind of content, this placement strategy improves response time by being able to serve requests locally. Thus this technology is widely used in recent trends and used in this paper model

Strategic point oriented deployment

In strategic point oriented deployment method the dynamic deployment of caches at network choke points, is a strategy embraced by the adaptive caching approach. Although it would seem to provide the most flexible type of cache coverage, it is still a work in progress and, to the best of the authors' knowledge, there have not been any performance studies demonstrating its benefits. The dynamic deployment technique also raises important questions about the administrative control of these caches, such as what impact network boundaries would have on cache mesh formation.

VI. COMBINING CACHING AND PREFETCHING

Prefetching and caching are two known approaches for improving the performance of file systems. Although they have been studied extensively, most studies on prefetching have been conducted in the absence of caching or for a fixed caching strategy. After the invention of complication in individual prefetching technology (i.e.) prefetching file blocks into a cache can be harmful even if the blocks will be accessed in the near future. This is because a cache block needs to be reserved for the block being prefetched at the time the Prefetch is initiated. The reservation of a cache block requires performing a cache block replacement earlier than it would otherwise have been done. Making the decision earlier may hurt performance because new and possibly better replacement opportunities open up as the program proceeds and hence combining caching and prefetching is essential.

VII. CACHE REPLACEMENT POLICIES

One of the key complications in implementing cache replacement policies for Web objects is that the objects to be cached are not necessarily of homogeneous size. For example, if two objects are accessed with equal frequency, the hit ratio is maximized when the replacement policy is biased towards the smaller object. This is because it is possible to store a larger number of objects of smaller size.

In addition to non homogeneous object sizes, there are several other special features of the Web, which need to be considered. First, the hit ratio may not be the best possible measure for evaluating the quality of a Web caching algorithm. For example, the transfer time cost for transferring a large object is more than that for a small object, though the relationship is typically not straightforward. It will depend on the distance of the object from the Web server. Furthermore, Web objects will typically have expiration times. So, when considering which objects to replace when a new object enters a Web cache.

We must consider not only the relative frequency, but also factors such as object sizes, transfer time savings, and expiration times. It may not always be favorable to insert an object into the cache, because it may lower the probability of a hit to the cache.

However, maximizing the cache hit ratio alone does not guarantee the best client response time in the Web environment. In addition to maximizing the cache hit ratio, a cache replacement algorithm for Web documents should also minimize the cost of cache misses, i.e., the delays caused by fetching documents not found in the cache. Clearly, the documents, which took a long time to fetch, should be preferentially retained in the cache. For example, consider a proxy cache at Northwestern University. The cache replacement algorithm at the proxy found two possible candidates for replacement. Both documents have the same size and are referenced with the same rate, but one document originates from the University of Chicago while the other is from Seoul National University. The cache

replacement algorithm should select for replacement the document from the University of Chicago and retain the document from Seoul National University because upon a cache miss the former can be fetched much faster than the latter.

VIII. RELATED REPLACEMENT ALGORITHMS

1) *Least Recently-Used (LRU)*

In the standard least recently used (LRU) caching algorithm for equal sized objects we maintain a list of the objects in the cache, which is ordered, based on the time of last access. In particular, the most recently accessed object is at the top of the list, while the least recently accessed object is at the bottom. When a new object comes in and the cache is full, one object in the cache must be pruned in order to make room for the newly accessed object. The object chosen is the one which was least recently used. Clearly the LRU policy needs to be extended to handle objects of varying sizes.

$$MI_{LRU} = \frac{1 - \%FirstTimers - \%LRUHitRatio}{\%LRUHitRatio}$$

LRU treats all documents equally, without considering the document size, type or network distance. It also ignores frequency information, thus an often-requested document will not be kept if it is not requested for a short period so LRU does not perform well in web caches. A scan, stream of multiple documents accessed only once, can force all the popular documents out of an LRU-based cache.

2) *Least-Frequently-Used (LFU)*

In least-frequently used method evicts the document, which is accessed least frequently. Least Frequently Used also has some disadvantages; the important problem with this method is cache pollution which means that a document that was formerly popular, but no longer is, will stay in the cache until new documents become more popular than the old one which was used. This can take a long time, and during this time, part of the cache is wasted. It assumes that probabilities are constant, but in practical it is not so. it also includes problem with implementation. Ideally, an implementation of the algorithm would keep a frequency counter for documents not in the cache as well as those present. On the scale of the web, this is prohibitive. However, if this is not done, the performance of the algorithm diminishes. Even if only the counters for the documents in the cache are kept, counters have to be updated continuously, even when no document needs to be replaced, incurring considerable overhead. It is also necessary to keep the documents sorted by frequency to be able to make rapid decisions at replacement time. This method has no notion of document size or cost of retrieval.

IX. PROPOSED SYSTEM MODEL

The proxy is located near the Web clients in order to avoid repeated round-trip delays between the clients and the origin Web servers. The origin Web server in this model is

an enhanced Web server which employs a prediction engine and this will derive prefetching rules from the server's access log periodically. These derived rules are assumed to be frequent. That is, only rules with supports larger than the minimum support are derived and provided by Web servers. The derived prefetching rules are stored in the prefetching rule depository of the Web server.

As shown in Fig.1, the proxy serves the requests sent from the Web clients. In the case that a cache miss occurs, the proxy will forward the request to the origin Web server for resolution. Upon receiving the request, the origin server will log this request into record, fetch the requested object from the Web object depository, and check the prefetching rule depository at the same time. If this request triggers some prefetching rules in the prefetching rule depository, the objects implied by these prefetching rules and their corresponding confidences will be piggybacked to the responding message as hints and returned to the proxy. After the proxy receives the response with the hints piggybacked from the origin Web server, the proxy will first send the requested object back to the client and then determine whether it is worth caching the piggybacked implied objects in the proxy. Here the cache replacement algorithm is devised for the integration of Web caching and Web prefetching techniques. If cache hit is found (i.e., the client's request can be satisfied directly with the proxy's local cache), we assume that the proxy will still communicate with the origin Web server to obtain the prefetching hints related to that request after the proxy has sent the response to the client. As such, we are able to investigate each request the prefetching hints from the origin Web server to ensure that the discovered prefetching hints are always up-to date.

A typical system model is shown below

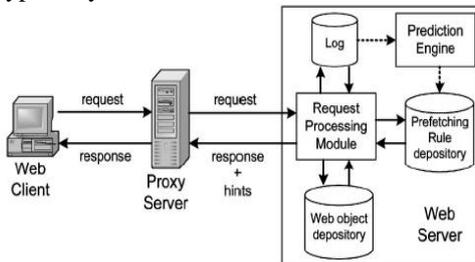
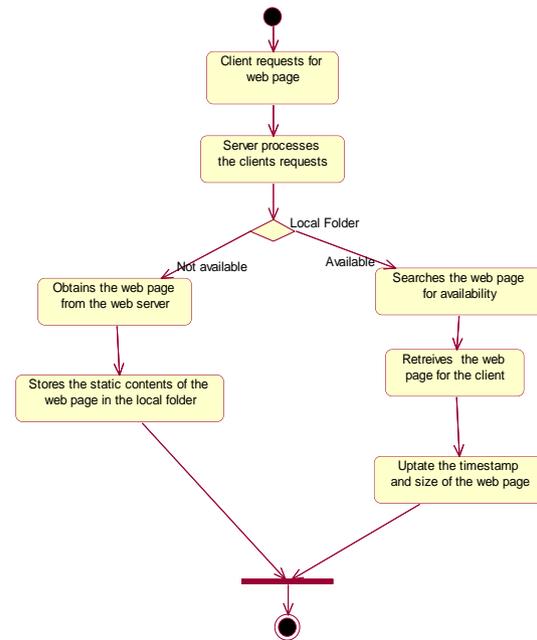


Fig. 1 Module for integrating web caching and prefetching



2 Activity flow of cache

Fig.

In fig2 the flow of the data is described that is any pages from web can be accessed by client after it send some request and wait for the response. The server processes the client request and analyze that it was in cache folder or it was get from the web server, if it retrieves from the local cache folder itself it simply update the times amp and size of the web page and forward to the client, else if get from the web server, it saves a copy in local folder then forward to client.

X. CREATION OF PROXY SERVER CONFIGURATION

In this module proxy server is configured in which the images are stored separately. This is done by initializing the Boolean function for images. The port number and maximum number of connection possible with the proxy server are initialized and during the running mode of the server, the server is started and we check whether the ip address is present. If so the server socket is created with port number and maximum number of connection and if ip address is not present then we get the new ip address for it and this process is done by setting the timeout period.

Graphical user interface design is done in this phase. This includes the menu items like start server and stop server and viewing cache and the graphical user interface contains help menu item and client can also view that administrator is currently connected to which person.

Connection of proxy server with Internet

The proxy server is connected with the internet by creating HTTP configuration and proxy cache pool. Proxy ip address

and port number is connected with the network and if the cache is enabled then that cache is used.

Replacement algorithm design

N: Total number of Documents

M: Next Least useful document

Eviction: Retrieval the requesting document

Sample: Method for selecting the retrieval option

If (eviction)

```

{
    if(first_iteration)
    {
        sample(N);
        evict_least_useful;

        keep_least_useful(M);
    }
else
    {
        sample(N-M);

        evict_least_useful;

        keep_least_useful(M);
    }
}
    
```

Replacement algorithm designing includes two iteration models. in the first iteration step the N-documents is randomly picked from the cache and among that N-documents the least useful document is evicted and then next M least useful document is retained and in subsequent iteration the N-M documents is randomly picked from cache And it is Appended to the M previously retained documents and among the N-samples the least useful document is evicted and M next least useful document are retained.

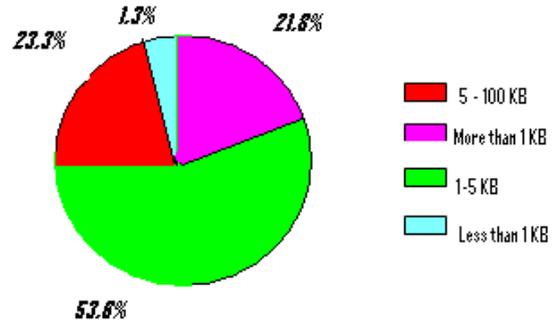


Fig. 3 Pie chart Notification

The pie chart identifies the percentage notification of requests 20% most popular objects. In fig 3 the chart defines the percentage of the requests of the file for a sample organization. In that it shows that less than 1 kilo byte size files are requested is only 1.3%, more than 1 kilo byte size files are requested 21.8%, 5- 100 kilo byte size files are requested 23.3% and 1 – 5 kilo byte size files are requested 53.6%, when compare to all the final identification which was green in color is higher percentage which was requested by the client.

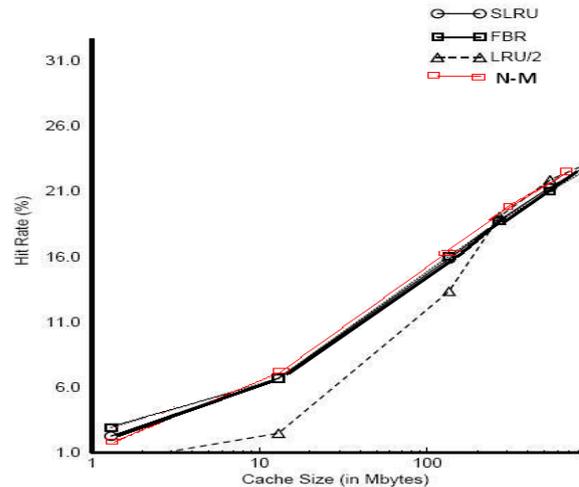


Fig. 4 Cache Ratio Notification

In fig 4 The Hit ratio for our algorithm is implemented, here we compare our algorithm with SLRU (second Least recently used), FBR and LRU/2. The ratio of our algorithm has slight difference from the other algorithms retrieval. In the above figure we describe the hit ratio along with our cache size.

XI. CONCLUSION

The proxy server is designed, doesn't use the data structure to store the data in the cache and hence the server is capable of accessing to number of clients and the administrator can view the data's present in the cache. The organization in the

proxy server is a simple local folder which holds the ".cache" files and the images in separate folder. The work to be proposed in the next phase is to connect this proxy server where in we connect the server to a LAN. The server provides the internet connection to the LAN. Later the cache replacement algorithms are implemented to reduce the response time of the user. The existing algorithms are being studied and the improvements in the algorithms are being implemented. Improvements are being done to the algorithms on various then the improved proxy server is created. So we conclude that the performances of the replacement algorithms are improved.

BIBLIOGRAPHY

- [1] C. Aggarwal, J.L. Wolf, and P.-S. Yu, "Caching on the World Wide Web," IEEE Trans. Knowledge and Data Eng., vol. 11, no. 1, pp. 94- 107, Jan. /Feb. 1999.
- [2] G. Barish and K. Obraczka, "World Wide Web Caching: Trends and Techniques," IEEE Comm. Magazine, Internet Technology Series, pp. 178-185, 2000.
- [3] Proxy Cache Replacement Algorithms:
A History-Based Approach Department of Informatics,
Aristotle University, Thessaloniki 54006, Greece VTEX
(CG) PIPS No:404532 artty:ra (Kluwer BO v.2001/10/30)
WJ404532.tex; 22/02/2002;
- [4] An Efficient Web Caching Algorithm based on LFU-K
replacement policy, Proceedings of the Spring Young

Researcher's Collo-quium on Database and Information
Systems SYR-CoDIS, St.-Petersburg, Russia, 2004

[5] Naizheng Bian, Hao Chen on "A Least Grade Page
Replacement Algorithm for Web Cache Optimization" IEEE
Computer society 2008 Workshop on Knowledge Discovery
and Data Mining

[6] Shenggang Wan Qiang Cao Xubin He Changsheng
Xie Chentao Wu on "An Adaptive Cache
Management Using Dual LRU Stacks to Improve Buffer
Cache Performance "IEEE Computing and Communication
conference, Publication Date: 7-9 Dec. 2008 On page(s): 43-
50

[7] Shih-Hao Hung Chien-Cheng Wu Chia-Heng Tu on
"Optimizing the Embedded Caching and Prefetching
Software on a Network-Attached Storage System". IEEE/
IFIP Publication Date:17-20Dec.2008 Volume:
1, On page(s): 152-161.

Corresponding Author:

G.N.K.Suresh Babu

Apollo Engineering College,

Chennai, Tamil Nadu, India

91-9841586522

Wireless LAN Security System

Qasim Siddique
Foundation university Islamabad

Abstract- In just the past few years wireless LAN has come to occupy a significant niche in the local area network market. Increasingly, organization are finding that WLAN are an indispensable adjunct to traditional wired LAN to satisfy requirement for mobility relocation ,ad hoc networking and coverage of location difficult to wired

Wireless technologies have become increasingly popular in our everyday business and personal lives. Today, wireless communication technology has become very important part of our daily life. The use of wireless communication technology is increasing day by day but there is a danger of black hole associated with these types of networks communications (wireless). This research paper will provide an overview of security risks, threats and vulnerabilities (weaknesses in the design and implementation) with wireless network systems, referencing IEEE 802.11. This research paper will not cover the topic of Bluetooth wireless security. To combat these risks, some protocols and mechanisms will be needed to secure this wireless communication and increase the use of wireless based systems.

I. INTRODUCTION

WLANs allow greater flexibility and portability than do traditional wired local area networks (LAN). Unlike a traditional LAN, which requires a wire to connect a user's computer to the network, a WLAN connects computers and other components to the network using an access point device. An access point communicates with devices equipped with wireless network adaptors it connects to a wired Ethernet LAN via an RJ-45 port. Access point devices Ethernet Card typically have coverage areas of up to approximately 100 meters (300 feet). This coverage area is called a range or cell. Users move freely within the cell with their laptop or other network device. Access point cells can be linked together to allow users to even "roam" within a building or between buildings.

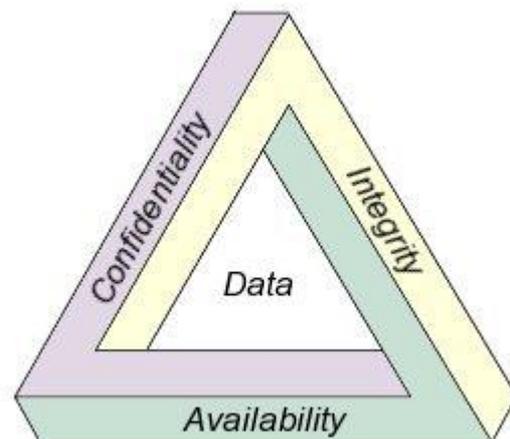
802.11 is the original wireless local area network standard, designed for 1 Mbps to 2 Mbps for communication wireless transmissions. In 1999 by 802.11a, established a high-speed wireless local area network standard for the 5 GHz band and supported 54 Mbps data rate. Also completed in 1999 was the 802.11b standard, which operates in the 2.4- 2.48GHz band and supports 11 Mbps. The 802.11b standard is currently the dominant/Small business/organization standard for wireless local area networks, providing sufficient speeds for most used application in today's world. Because the 802.11b standard has been so widely used in today's world, the security weaknesses in the standard have also been exposed. Another standard, 802.11g, still in draft, operates in the 2.4 GHz waveband, where current wireless local area network products based on the 802.11b standard operate.

"As wireless based system is developed and implemented and used the complexity of the types of attacks will increase, but these appear the standard main methods used to break and Attack wireless systems. These attacks may be very similar against other wireless type Technologies and is not unique to 802.11b. Understanding these risks and how to develop Security solution for 802.11b will provide a strong foundation for integrating a good secure solution to any wireless solution". (**Maxim Pollino 2002**).

II. WIRELESS TECHNOLOGIES STANDARDS

Wireless technologies conform to a variety of standards and offer varying levels of security features. The principal advantages of standards are to encourage mass production and to allow products from multiple vendors to interoperate 802.11 And the Bluetooth standard. WLANs follow the IEEE 802.11 standards. Ad hoc networks follow techniques or are based on the Bluetooth standard, which was developed by a consortium of commercial companies making up the Bluetooth Special Interest Group (SIG). These standards are followed below.

- IEEE 802.11
- Bluetooth



III. SECURITY FEATURES OF 802.11

The three basic security services defined by IEEE for the WLAN environment are as follows

1) Authentication

A primary goal of WEP was to provide a security service to verify the identity of communicating client stations. This provides access control to the network by denying access to client stations that cannot authenticate properly. This service

addresses the question, “Are only authorized persons allowed to gain access to my network?”

2) Confidentiality

Confidentiality, or privacy, was a second goal of WEP. It was developed to provide “privacy achieved by a wired network.” The intent was to prevent information compromise from casual eavesdropping (passive attack). This service, in general, addresses the question, “Are only authorized persons allowed to view my data?”

3) Integrity

Another goal of WEP was a security service developed to ensure that messages are not modified in transit between the wireless clients and the access point in an active attack. This service addresses the question, “Is the data coming into or exiting the network trustworthy has it been tampered with?”

IV. PROBLEMS IEEE 802.11 STANDARD SECURITY

1) Security features in vendor products are frequently not enabled

Security features, albeit poor in some cases are not enabled when shipped, and users do not enable when installed. Bad security is generally better than no security.

2) IVs are short (or static)

24-bit IVs cause the generated key stream to repeat. Repetition allows easy decryption of data for a moderately sophisticated adversary

3) Cryptographic keys are short

40-bit keys are inadequate for any system. It is generally accepted that key sizes should be greater than 80 bits in length. The longer the key, the less likely a compromise is possible from a brute-force attack.

4) Cryptographic keys are shared

number of people sharing the key grows, the security risks also grow. A fundamental tenant of cryptography is that the security of a system is largely dependent on the secrecy of the keys.

5) Cryptographic keys cannot be updated automatically and frequently

Cryptographic keys should be changed often to prevent brute-force attacks.

6) RC4 has a weak key schedule and is inappropriately used in WEP

7) *The combination of revealing 24 key bits in the IV and a weakness in the initial few bytes of the RC4 key stream leads to an efficient attack that recovers the key. Most other applications of RC4 do not expose the weaknesses of RC4 because they do not reveal key bits and*

do not restart the key schedule for every packet. This attack is available to moderately sophisticated adversaries

8) Packet integrity is poor

CRC32 and other linear block codes are inadequate for providing cryptographic integrity. Message modification is possible. Linear codes are inadequate for the protection against advertent attacks on data integrity. Cryptographic protection is required to prevent deliberate attacks. Use of non cryptographic protocols often facilitates attacks against the cryptography.

9) No user authentication Occurs

Only the device is authenticated. A device that is stolen can access the network.

10) Authentication is not enabled; only simple SSID identification occurs

Identity-based systems are highly vulnerable particularly in a wireless system because signals can be more easily intercepted.

11) Device authentication is simple shared-key challenge-response

One-way challenge-response authentication is subject to “man in the middle” attacks. Mutual authentication is required to provide verification that users and the network are legitimate.

12) The client does not authenticate the Access Point

The client needs to authenticate the Access Point to ensure that it is legitimate and prevent the introduction of rogue Access Point.

V. SECURING WIRELESS LOCAL AREA NETWORK

Wireless local area network operates in the same fashion as a wired local area network except that data is transported through a wireless medium rather than cables. The following sections describe common threats that local area network faced with and some countermeasures that can be employed to protect against such threats. Some of the threats are as followed:

Threats

➤ Eavesdropping

The main threat posed to a Wireless local area network is the potential for unauthorized persons to eavesdrop on radio signals transferred between a wireless station and an access point which compromises the privacy of sensitive information and data (**Barken 2004**).

Eavesdropping is considered to be a **passive attack**.

Example

When a radio operator sends a message over a radio path, other users who are equipped with a compatible receiver within the range of the transmission are able to listen. Also, because an eavesdropper has the ability to listen to a message without modifying the data, the sender and intended receiver of the message are unaware that there has been an intrusion

➤ Unauthorized Access

Another threat to Wireless local area network is when an intruder enters a Wireless local area network disguised as an authorized user. When the intruder has gained access, he can violate the confidentiality and integrity of the network traffic by sending, receiving, altering, or forging messages (Nichols, Lekkas 2002).

Unauthorized access is considered as an active attack and can be executed using a wireless adapter, which is compatible with the wireless network

➤ Authentication Mechanisms Try

One of the best protections against this type of unauthorized access is to deploy authentication mechanisms to ensure that only users who are authorized can gain access to the network. One of the hardest tasks for wireless LAN is to detect unauthorized access when they occur. This is because unsuccessful attacks might be misinterpreted as merely unsuccessful logon attempts caused by high bit error rate.

➤ Capture Secret keys and Passwords

An attacker can lure a station onto his network in order to capture secret keys and passwords. Another way to accomplish this is that the attacker rejects the logon attempts but record the messages transmitted during the logon process.

The first attack described is very hard to execute because the attacker must have specific details in order to deceive the station into believing that it has accessed its home network.

The second attack mentioned is easier to implement because in this case all that is required by the attacker is a receiver and an antenna that is compatible with the stations.

In addition to this, the attack is more difficult to detect. This is because the unsuccessful logons are common in WLAN environments. The best method to protect against these types of attacks is to employ an efficient mechanism that allows wireless stations to authenticate to access points without disclosing confidential keys or passwords.

➤ Interference and Jamming

A third threat to wireless LAN security is radio interference which can deteriorate bandwidth. In most cases the interference is accidental. Since WLANs use unlicensed radio waves, other electromagnetic devices can coincide with WLAN traffic (Barken 2004)

Sources of interference can include high power amateur, military, and industrial, scientific, and military transmitters.

➤ Denial of Service Attack

Interference may also be intentional. If an attacker has a powerful transmission, he can produce a radio signal strong enough to overwhelm weaker signals which can disrupt communications. This is known as jamming and is a denial of service attack.

VI. TYPES OF JAMMERS

There are two types of jammers which are as followed and can be utilized against wireless LAN

- i) Traffic
- ii) High Power Pulsed and lower power partial-band jammers.

Jamming equipment is available to consumers or can be created by attackers. These types of attacks can be mounted remotely from the targeted network.

VII. PHYSICAL THREATS

The physical structure of a wireless Local area network can be impacted if it is damaged. Similar to a wired LAN, a wireless Local area network operating in infrastructure mode is dependant upon a number of physical components.

Physical Components

Some of the physical components are as followed

- ✓ Access points APs,
- ✓ Cables,
- ✓ Antennas,
- ✓ Wireless adapter,
- ✓ Software.

Harm to any of these could significantly reduce the strength of the signal, limit coverage area, or reduce bandwidth.

Infrastructure components are also vulnerable to the conditions of its environment, especially if outdoors. APs can be affected by snow and ice. Antennas which are placed on poles or buildings have the risk of being knocked down by winds, rain, or ice which can change the beam width for transmitting signals. Finally, Physical components can be attacked.

Example

An attacker could cut the cabling that connects an AP to the wired network, isolating affected microcells and disrupting power to the receiver. Another potential attack could involve stealing or compromising wireless station or adapter and see it to try and intercept WLAN traffic or to gain unauthorized access to the network. Accidents and improper handling can harm wireless adapters and wireless stations.(Can cause damage to whole network and make it unusable)

VIII. OTHER SECURITY RISKS

With the prevalence of wireless devices, more users are seeking ways to connect remotely to their own organization's networks. One such method is the use of UN trusted third-party networks. Conference centers, for example, commonly provide wireless networks for users to connect to the Internet and subsequently to their own organizations while at the conference. Airports, hotels, and even some coffee franchises are beginning to deploy 802.11 based publicly accessible wireless networks for their customers, even offering VPN capabilities for added security.

IX. UN TRUSTED PUBLIC NETWORKS PRIMARY RISKS

These un-trusted public networks introduce three primary risks which are as followed

- 1) They are public; they are accessible by anyone, even malicious users;
- 2) They serve as a bridge to a user's own network, thus potentially allowing anyone on the public network to attack or gain access to the bridged network
- 3) They use high-gain antennas to improve reception and increase coverage area, thus allowing malicious users to eavesdrop more readily on their signals.

By connecting to their own networks via an untrusted network, users may create vulnerabilities for their company networks and systems unless their organizations take steps to protect their users and themselves. Users typically need to access resources that their organizations deem as either public or private. Agencies may want to consider protecting their public resources using an application layer security protocol such as Transport Layer Security (TLS), the Internet Engineering Task Force standardized version of Secure Sockets Layer (SSL). However, in most agencies, this is unnecessary since the information is indeed public already. For private resources, agencies should consider using a VPN solution to secure their connections because this will help prevent eavesdropping and unauthorized access to private resources.*

X. COUNTERMEASURES

Some of the countermeasures are as followed

1) *Management Countermeasures*

Management countermeasures for securing wireless networks begin with a comprehensive security policy. A security policy and compliance therewith, is the foundation on which other countermeasures the operational and technical are rationalized and implemented.

2) *Security Policy*

A Wireless local area network security policy should be able to do the following jobs

- Identify who may use Wireless local area network technology in an agency
- Identify Internet access is required or not
- Describe who can install access points and other wireless devices
- Provide limitations on the location of and physical security(Guards) for access points
- Describe the nature of information that may be sent over wireless links
- Describe conditions and requirement under which wireless devices are allowed
- Define security settings for access points
- Describe limitations on how the wireless device may be used
- Describe the configuration (H/W and S/W) of all wireless devices
- Provide guidelines on reporting losses of wireless devices and security incidents
- Provide guidelines for the protection of wireless clients to overcome theft
- Provide guidelines on the use of encryption and key management tools

XI. OPERATIONAL COUNTERMEASURES

Physical security is the most fundamental step for ensuring that only authorized users have access to wireless computer equipment. Physical security combines such measures as access controls, personnel identification, and external boundary protection. As with facilities housing wired networks, facilities supporting wireless networks need physical access controls.

Example

Photo identification, card badge readers, or biometric devices can be used to minimize the risk of improper penetration of facilities. Biometric systems for physical access control include palm scans, hand geometry, iris scans, retina scans, fingerprint, voice pattern, signature dynamics, or facial recognition. External boundary protection can include locking doors and installing video cameras for surveillance around the perimeter of a site to discourage unauthorized access to wireless networking components such as wireless APs. It is important to consider the range of the AP when deciding where to place an AP in a wireless system environment. If the range extends beyond the physical boundaries of the office building walls, the extension creates security vulnerability. An individual outside of the building, perhaps "war driving," could eavesdrop on network communications by using a wireless device that picks up the RF emanations. A similar consideration applies to the implementation of building-to-building bridges. Ideally, the APs should be placed strategically within a building so that the range does not

exceed the physical perimeter of the building and allow unauthorized personnel to eavesdrop near the perimeter. Agencies should use site survey tools to measure the range of AP devices, both inside and outside of the building where the wireless network is located. In addition, agencies should use wireless security assessment tools (e.g., vulnerability assessment) and regularly conduct scheduled security audits.

XII. TECHNICAL COUNTERMEASURES

Technical countermeasures involve the use of hardware and software solutions to help secure the wireless local area network systems

1) Software countermeasures

Software countermeasures include proper AP configurations (i.e., the operational and environment. security settings on an AP), software patches and upgrades, authentication, intrusion detection systems (IDS), and encryption.

2) Hardware countermeasures

Hardware solutions include smart cards, VPNs, public key infrastructure (PKI)

XIII. SMART CARDS

Smart cards may add another level of protection, although they also add another layer of complexity. Agencies can use smart cards in conjunction with username or password or by themselves. They can use smart cards in two-factor authentication. Agencies can also combine smart cards with biometrics.

In wireless networks, smart cards provide the added feature of authentication. Smart cards are beneficial in environments requiring authentication beyond simple username and password. User certificate and other information are stored on the cards themselves and generally require the user only to remember a PIN number. Smart cards are also portable; consequently users can securely access their networks from various locations. As with an authentication software solution, these tamper-resistant devices may be integrated into a WLAN solution to enhance the security of the system. Again, users should be careful to fully understand the security provided by the smart card solution.

XIV. VIRTUAL PRIVATE NETWORKS

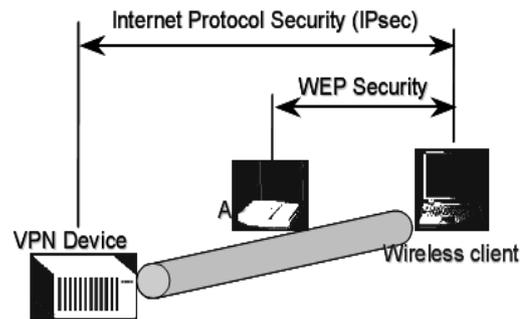
VPN technology is a rapidly growing technology that provides secure data transmission across public network infrastructures. VPNs have in recent years allowed corporations to harness the power of the Internet for remote access. VPNs are typically used in three different scenarios:

- i) Remote user access,
- ii) LAN-to-LAN (site-to-site) connectivity
- iii) Extranet

Most VPNs in use today make use of the IPsec protocol suite. IPsec, developed by the Internet Engineering Task

Force (IETF), is a framework of open standards for ensuring private communications over IP networks. It provides the following types of robust protection

- Confidentiality
- Integrity
- Data origin authentication
- Traffic analysis protection.



Infrared

IR is the third radio technology specified in the original 802.11 standard. This Technology transmits data at high frequencies just below visible light on the electromagnetic system. IR signals are susceptible to interception, interference, and jamming. Therefore, IR systems are typically utilized for high-security applications in enclosed facilities (**Barken 2004**).

Drawback

It is also more expensive than the Spread-Spectrum technologies mentioned above in addition to its data rate being low 1 – 2 Mbps.

Narrowband

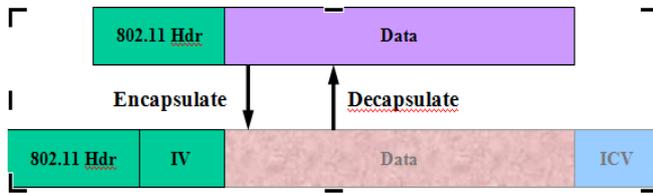
Narrowband transmits and receives radio signals on a specific frequency. This keeps the radio signal as narrow as possible. This method prevents cross-talk among radio channels by coordinating different channel frequencies.

Drawback

A drawback of narrowband is that eavesdroppers can easily detect transmitted signals. It also requires a license from the FCC for each site that it is used at.

XV. WIRED EQUIVALENT PRIVACY (WEP)

WEP (Wired Equivalent Privacy) is implemented in the 802.11 specification to provide basic levels of authentication and data encryption. 802.11b utilizes WEP. It is a crucial element for securing confidentiality and integrity of on WLAN systems in addition to providing access control through authentication.



XVI. ENCRYPTION

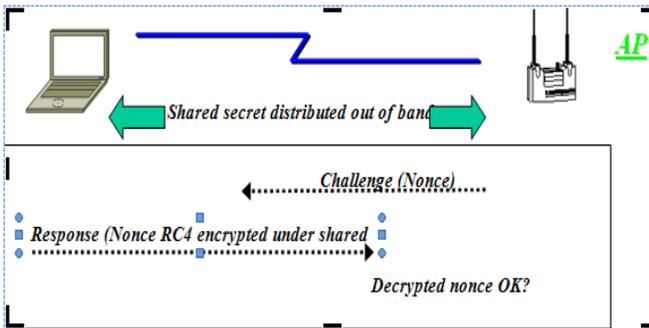
WEP uses a shared secret key between a wireless stations and an access point. The data Sent and received between the station and AP can be encrypted using the shared key. WEP provides data encryption with a 128-bit secret key and a RC4 Pseudo Random Generator.

Processes

There are two processes that are applied to plaintext data; one encrypts the plaintext and the other protects it from unauthorized modification during transition. After the secret key has encrypted the text, it returns the encrypted text back to the AP. If the text matches the text that was sent then the client is authorized and granted access.

Problem

A problem that this method has is that the key distribution. Most WLANs share one key across all stations and Access Points in the network. It's not likely that a key shared among several users will remain secret forever. Some network administrators address this issue by configuring wireless stations with the secret key as opposed to allowing users to execute this task.



XVII. AUTHENTICATION

There are two types of authentication that WEP provides

- Default Open System (all users are permitted to access a WLAN)
- Shared key authentication (controls access to WLAN and prevents unauthorized network access).

1) Shared key authentication

Shared key authentication is the more secure mode. It employs a secret key that is shared among all stations and Access points in a WLAN. Shared key works only if WEP encryption is enabled.

2) Default Open System mode

The system will default to Open System mode which will permit most any station within range of an AP to access the network. This will permit an intruder to enter the system where he can interfere with your messages. It is important to ensure that WEP is enabled whenever secure authentication is required.

In many WLAN systems, the key utilized for authentication is the same key used for encryption. This presents a weakness which strengthens the problems mentioned above. If the attacker has control of the shared key he can access the network in addition to decrypt the messages. The solution is to distribute separate keys throughout the system one for authentication and one for encryption.

XVIII. CONCLUSION

The main purpose of this research paper is to provide an overview of security risks, threats and vulnerabilities (weaknesses in the design and implementation) with wireless network systems, referencing IEEE 802.11. After our research we conclude that the wireless network are increasing day by day we must implemented all types of security policies in our wireless local area network security system. And bring many technical countermeasures in our wireless network. To protect our network from the security risk A better solution is to assign a unique key to each station and to change keys frequently in Wired Equivalent Privacy.

Solution of problems with Existing 802.11 Wireless LAN Security

Cryptographic keys are short

The main problems of the IEEE 802.11 WLAN is that the Cryptographic keys is short its length is only 40 bits brute force attacked can easily be applied on it

$2^{40} = 1099511627776$ by applying this much combination the key can be broken

Time required breaking 40 bit key = 10 hours

$2^{40} = 1099511627776 = 10$ hours

But if we increase the key bit length up to 128 then

$2^{128} = 3.4 * 10^{38} = 5.4 * 10^{18}$ Years

So we can solve this problem by increasing the key bit length and it also required more memory space and more computing time

Cryptographic keys are shared

Sharing the cryptographic key with the user we must authenticate the user and its device and then share the key after the user is authenticated

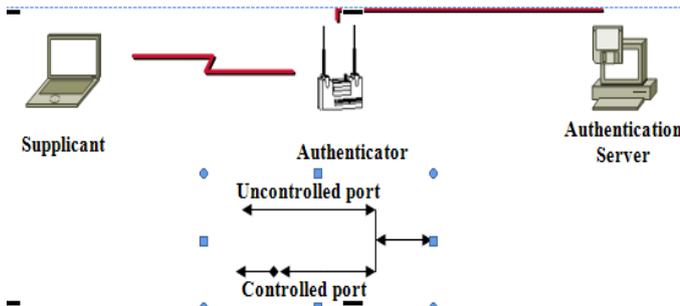
Cryptographic keys cannot be updated automatically and frequently

By placing some type of technical mechanism in the cryptographic key we can update them automatically and can change them whenever we required. But if we increased

the bit of the key we may not required them to update frequently.

No user authentication occurs

In wireless network only device is authenticated we can prevent these problems by giving this user a specific type of digital certificate and update that certificate whenever user logon on the network.



Maxim, Merrit and Daivd Pollino. Wireless Security. McGraw-Hill/Osborne, 2002.

Barken, Lee. How Secure Is Your Wireless Network? Saddle River, NJ: Prentice Hall

PTR, 2007

Nichols, Randall K. and Panos C. Lekkas. Wireless Security: Models, Threats, and

Solutions. McGraw-Hill, 2004

Mallick, Martyn. Mobile & Wireless Design Essentials. Wiley Publishing, Inc: Indianapolis, Indiana, 2003

Dubendorf, Vern A. Wireless Data Technologies. West Sussex, England: John Wiley & Sons Ltd, 2006

The client does not authenticate the Access Point

Client must be authenticated on the access point before they logon the WLAN.

RECOMMENDATION

- Develop an agency security policy that addresses the use of wireless technology, including 802.11.
- Ensure that users on the network are fully trained in computer security awareness and the risks associated with wireless technology
- Locate APs on the interior of buildings instead of near exterior walls and windows
- Ensure that wireless networks are not used until they comply with the agency's security policy.
- Disable all insecure and non essential management protocols on the Access point.
- Ensure that encryption key sizes are at least 128 bits or as large as possible Install a properly configured firewall between the wired infrastructure and the wireless network (AP or hub to APs).
- Install antivirus software on all wireless clients.
 1. Understand and make sure that all default parameters are changed

XIX. BIBLIOGRAPHY

Sara Nasre Wireless LAN Security Research Paper [Most of the part of this Research paper is from Sara Nasre Research Paper]

IEEE 802.11 http://en.wikipedia.org/wiki/IEEE_802.11

Tom Kerygiannis Les Owens Wireless network security 802.11,Bluetooth and Handheld Devices

A Trust-Based Secured Routing Protocol for Mobile Ad hoc Networks

K.Seshadri Ramana

DR. A.A. Chari

Prof. N.Kasiviswanath

Abstract- In wireless adhoc networks, all nodes are mobile and can be connected dynamically in an arbitrary manner for packet type communications. All nodes behave as routers and take part in discovery and maintenance of routes to other nodes in the network. In this paper, we propose a routing protocol that is based on securing the routing information from unauthorized users. Even though routing protocols of this category are already proposed, they are not efficient, in the sense that, they use the same kind of encryption algorithms for every bit of routing information they pass from one intermediate node to another in the routing path. This consumes lot of energy or power as well as time. Our routing algorithm basically behaves depending upon the trust one node has on its neighbor. The trust factor and the level of security assigned to the information flow decide what level of encryption is applied to the current routing information at a source or intermediate node. In other words, above a certain level of trust level, there is no need for the source or intermediate node to perform high level encryption on the routing information as it completely trusts the neighboring node. So based on level of trust factor, the routing information will be low-level, medium level, high level encrypted, the low-level being normal AODV. This not only saves the node's power by avoiding unnecessary encoding, but also in terms of time, which is very much valuable in cases of emergencies where the information is as valuable as the time.

KEYWORDS:

Ad-hoc Routing Protocol, AODV, encryption, Decryption, trust factor, security level, Trust.

^{1.} K.Seshadri Ramana, Associate Professor, G.Pulla Reddy Engineering College, Kurnool, A.P., India. (Email: ramana.kothapalli@gmail.com.)

^{2.} Dr.A.A.Chari, Professor Department of OR&C, Rayalaseema University, Kurnool, A.P., India.

^{3.} Prof. N.Kasiviswanath, Professor & Head of Computer & Science Engineering, G.Pulla Reddy Engineering College, Kurnool, A.P., India. (Email: nkasiviswanath@yahoo.com)

I. INTRODUCTION

Mobile host and wireless networking hardware are becoming widely available, and extensive work has been done in the recent years in integrating these elements into traditional networks such as internet. They can be often used in scenarios in which no infrastructure exists, or in which the existing infrastructure does not meet application requirements for reasons of security or cost. Ad hoc routing protocols are challenging to design and secure ones are even more so. Prior research has generally studied the routing problem in a non-adversarial setting, assuming a trusted environment [7]. These may be sufficient for normal day-to-day applications but for applications such as military exercises and disaster relief, a secure and a more reliable communication is a prerequisite.

Our main focus is on on-demand routing protocols [7], in which a node attempts to discover a route to some destination, if and only if has a packet to send to that destination. The source must wait until a route has been discovered, but the traffic overhead is less than Table-driven algorithms [7] where many of the updates are for the unused paths. This reduced overhead affects bandwidth utilization, throughput as well as power usage. No prior advertisement is done, which makes the on-demand routing protocols covert in nature. However, this property is alone not enough to stop a malicious user to access the routing information and initiate directed attacks at the source, destination or any other intermediate node in the network, thus effectively disrupting or even bring down the network.

In applications involving secure and covert operations, information security is one thing that can never be compromised. These operations would rather go for a dependable and unbreakable communication than for a cheap, insecure and fast communication. The idea is instead of going for a path, which involves unknown, not trustable enough nodes, it's better to go with the established path with known and trusted nodes. Routing protocols are very vulnerable since they can reveal topology information. Listening to few

DSR messages in promiscuous mode gives valuable information. A GPS based routing algorithm may give exact node locations. Typically, an attacker can playback routing information and easily collapse the network in different ways.

The remainder of this paper is organized as follows: Section 2 summarizes the basic operation of the Ad-hoc On-demand Distance Vector Routing [8] [13] on which we base the design of our secure routing protocol including why we chose it. In Section 3, we present the design of our new secure ad-hoc network routing protocol. Section 4 presents our simulation based performance evaluation of a basic form of our protocol. Finally, section 5 present concluding remarks.

II. AODV

The Ad hoc On Demand Distance Vector (AODV) routing algorithm is a routing protocol designed for ad hoc mobile networks. AODV is capable of both Unicast and multicast routing. It is an on demand algorithm, meaning that it builds routes between nodes only as desired by source nodes. It maintains these routes as long as they are needed by the sources. AODV uses sequence numbers to ensure the freshness of routes. It is loop-free, self-starting, and scales to large numbers of mobile nodes.

AODV builds routes using a route request / route reply query cycle. When a source node desires a route to a destination for which it does not already have a route, it broadcasts a route request (RREQ) packet across the network. Nodes receiving this Packet update their information for the source node and set up backwards pointers to the source node in the route tables. In addition to the source node's IP address, current sequence number, and broadcast ID, the RREQ also contains the most recent sequence number for the destination of which the source node is aware. A node receiving the RREQ may send a route reply (RREP) if it is either the destination or if it has a route to the destination with corresponding sequence number greater than or equal to that contained in the RREQ. If this is the case, it unicasts a RREP back to the source. Otherwise, it rebroadcasts the RREQ. Nodes keep track of the RREQ's source IP address and broadcast ID. If they receive a RREQ which they have already processed, they discard the RREQ and do not forward it.

As the RREP propagates back to the source, nodes set up forward pointers to the destination. Once the source node receives the RREP, it may begin to forward data packets to the destination. If the source later receives a RREP containing a greater sequence number or contains the same sequence number with a smaller hop-count, it may

update its routing information for that destination and begin using the better route. As long as the route remains active, it will continue to be maintained.

A route is considered active as long as there are data packets periodically traveling from the source to the destination along that path. Once the source stops sending data packets, the links will time out and eventually be deleted from the intermediate node routing tables. If a link break occurs while the route is active, the node upstream of the break propagates a route error (RERR) message to the source node to inform it of the now unreachable destination(s). After receiving the RERR, if the source node still desires the route, it can reinitiate route discovery.

AODV is chosen because of the inherent security in the protocol. Notice that one of the differences between AODV and DSR is that, DSR requires every packet to carry the routing information, whereas, in AODV, once the route is established, the data packets just carry the flow-ID. So, in DSR, we've to encrypt the routing information in every single data packet which is, not impossible, but not desired.

III. TRUST BASED ADAPTIVE ON DEMAND AD HOC ROUTING PROTOCOL

1) Design Goals:

The main aim is to mask the route path between the source and destination from all the other nodes, so as to avoid any kind of directed attacks. In fact, most of the routing disruption attacks are caused by malicious injection or altering of routing data. So, we feel that there is a need to prevent these attacks by totally hiding the routing information from unauthorized nodes.

2) Protocol Description:

In this protocol, routing information is shielded from every other node except the source and the destination. A few other routing protocols already exist implementing this idea by encrypting the routing information. This also involves in keeping the source node anonymous. It is to be noted that encryption is a very tedious process which involves consuming lot of nodes' time and energy. So, if this process is implemented at all intermediate nodes, it's very difficult to design a scalable, viable and efficient routing protocol design. Here is where the trust factor and security level of the application are implemented.

So, instead of masking from all the nodes, both time and energy can be saved by masking this information only from the un-trusted nodes. This also depends on the level of security that the application demands. The application demands and the trust levels can be classified as follows:

Security level : {high, medium, low}

Trust factor : { 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 }

Encryption : {high, medium, low}

	HIGH	MIDIUM	LOW
9,	medium	low	no
10	encryption	encryption	encryption
6,7,	high	medium	no
8	encryption	encryption	encryption
2,3,	high	high	no
4,5	encryption	encryption	encryption
0,1	-	-	-

Table 1: Security level description

The numbers in the table 1 correspond to the trust factor. The top column of high, medium, low relate to the security level of application. Suppose, if there is a neighboring node whose trust factor with the source node falls between 6 and 8, and the security level is set to “high”, then the routing information is highly encrypted. This doesn’t necessarily mean that this kind of encryption is going to take place all the way to the destination. Depending on the trust factor one node has on its neighbors and the level of security assigned to the application, the level of encryption varies. If the trust factor of a node falls below 2, then that node will not be included in any routing path.

Even though all the nodes in the routing path do encrypt their routing information, the difference lies in the keys they use. For ensuring the high level security 128-bit key will be used, where as for a low-level encryption a 32-bit key will be used. This ensures that instead of applying 128-bit key for every bit of routing information between every two nodes in the routing path, which involves spending considerable amount of time and node’s energy, we can actually fluctuate between these keys and save on the above mentioned parameters.

3) Route Discovery:

Route discovery allows any host in the ad hoc network to dynamically discover a route to any other host in the ad hoc network, whether directly reachable within wireless transmission range or reachable through one or more intermediate network hops through other hosts. A host initiating a route request first broadcasts a *route request* packet, which is received by those hosts within the wireless transmission range of it. An additional field, Security Level, has been added to the original RREQ. This is where the application will set the level of security it requires. Since, we are trying to keep the source

anonymous from other nodes and also take the trust factor of the neighboring node into consideration; we first look up the source nodes’ trust table and depending on the trust factor and the level of security for the application, we encrypt the Source ID with the public key of the destination. Now, the source broadcasts this message to its neighboring nodes.

Source -> broadcast: {RREQ, seqnum, PbD[Sid], D_{id}, SL }
 where seqnum is the sequence number, PbD[Sid] is the encrypted Source ID with the destination’s (D) public key, D_{id} is the Destination ID and SL is the security level set by the application. This is to make sure that only the destination can unlock the information and know who the source is.

When the neighboring node, node B, receives the RREQ packet, it looks into the packet and checks whether the RREQ is destined to it or not.

It then looks up it’s trust table for each of it’s neighboring node and then encodes its own information first with it’s private key, appends it to the source information and then encodes the whole with the public key of destination and locally broadcasts the RREQ packet.

Intermediate node -> broadcast: {RREQ, seqnum, PbD[P_vB[B_{id}], PbD[Sid]], D_{id}, SLq}
 where PbD[P_vB[B_{id}]] is the encrypted intermediate nodes’ ID (B).

Here, one might argue that the since the destination is open to everyone, then this RREQ might not be propagated all the way down. This might be true in cases where a malicious node is bent on disrupting the network. It’s not possible to eliminate the bad node altogether, so the best way is to avoid it. But here, it is not possible to initiate any directed attacks towards a particular route between a particular source and a particular destination. Since the source is not known, it is impossible for the passive malicious node to get information about the source. And if it still initiates its attack directed towards the destination, then it can be easily identified using ARIADNE, LHAP, ANODR etc., [12] [10] [11] [14] and listed as bad node and be avoided in further route discoveries.

In this way, the RREQ is propagated along the network and finally reaches the destination. The destination checks that this RREQ is destined to itself, then applies its private key and then public keys of the intermediate nodes in the order they were encoded. This helps in authenticating that the intermediate node themselves encoded their information and prevent any kind of misrepresentation by any malicious node.

The destination node then checks to see if there are any designated bad nodes (trust factor less than 2) in the

intermediate node list. It compares each node with its list of known bad nodes. If it finds any known bad nodes, it simply discards the RREQ and wait for the next RREQ to arrive. If every intermediate node is not in its bad node list then the destination node generates a flow-id and encodes it with the public keys of intermediate nodes in the order they would receive. Once this is done, the destination node locally broadcasts the RREP packet.

D -> broadcast: {RREP, P_{bC}[F_{id},P_{bB}[F_{id},P_{bS}[P_{vD}[F_{id}]]]]}]

Where P_{bC}, P_{bB}, are the public keys of the intermediate nodes in the order they were encoded. P_{bS} refers to the public key of the source and P_{vD} is the private key of the destination.

When the neighboring nodes receive the RREP packet, they would try to decode it using their public key. If they fail, they just discard the RREP, but if they are successful then they will update their corresponding route table path with the local source and destination along with the flow-id. And then, they will remove their part from the packet and locally broadcast it.

C -> broadcast: {RREP, P_{bB}[F_{id},P_{bS}[P_{vD}[F_{id}]]]}]

When the source receives the RREP, it first applies its private key and then the public key of destination. This authenticates the destination, and prevents misrepresentation of the destination by any malicious node. Now, the source gets the flow-ID generated by the destination which completes the process of route discovery. Now the source just uses the flow-ID in the header of the data packet to identify the route. All the intermediate nodes also use the flow-ID to identify the packet and forward them accordingly.

S -> B: {F_{id}, Data}

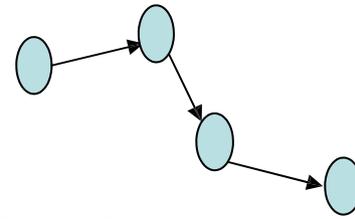


Figure 1: Example Scenario

4) *Route Maintenance:*

All nodes maintain tables which contain the information about the routes. Route disruption can occur due to various reasons. One of the important reasons is that since the nodes are mobile, it happens that some times they might move out of each other's transmission range.

Once the route is broken, a node cannot forward the packet to its neighbor. In this case, the node generates a route error packet, with the flow-id as the header and transmits it to the node up in the hierarchy. The error packet will be propagated all the way up to the source, which then issues a new route request. This is similar to normal AODV operation except for the local repair.

IV. PERFORMANCE EVALUATION

The Simulation platform used for evaluating the proposed approach is GlomoSim[1], a discrete – event , detailed simulator for wireless adhoc networks. It is based on the C-based parallel simulation language PARSEC[2]. The AODV protocol of GlomoSim was modified with cryptographically delay.

4.1 *Simulation Environment:*

To simulate the effects of encoding and decoding, we introduced delay when the nodes are issuing a RREQ, RREP, RERR, forwarding etc., Table 2 shows the performance (encryption and decryption bit-rate) of different cryptosystems.

Table 2. Processing Over head of Various Cryptosystems (on iPAQ3670 pocket PC with Intel StrongARM 206MHz CPU)[15]

Cryptosystem	decryption	encryption
ECAES (160-bit key)	42ms	160ms
RSA (1024-bit key)	900ms	30ms
El Gamal (1024-bit key)	80ms	100ms

AES/Rijndael (128-bit key & block)	29.2Mbps	29.1Mbps
RC6 (128-bit key & block)	53.8Mbps	49.2Mbps
Mars (128-bit key & block)	36.8Mbps	36.8Mbps
Serpent (128-bit key & block)	15.2Mbps	17.2Mbps
TwoFish (128-bit key & block)	30.9Mbps	30.8Mbps

A unique property of ad hoc networks is the dynamicity of the topology. The velocity of nodes is the main component of the network dynamicity. Ad hoc routing algorithms are designed to cope up with this property, thus, we choose a fast maximum time ranging from 0 to 200 seconds.

The random mobility generator based on the random way point algorithm [17] is used for the node movement pattern in the networks. The node movement is restricted to a flat terrain without any obstacles. The node starts moving toward a point independently and randomly chosen at speeds ranging between 0 and 20m/sec. It pauses for a predefined amount of time on arriving at the point.

For the communication pattern, we used constant bit rate (CBR) traffic model. The number of sources of CBR in the simulation is 30. Each source sends out 8 packets/sec using 64 byte packets. We run the simulation using twenty random scenarios at each pause time. The simulation lasts for 100 seconds. Table 3 summarizes the parameters chosen for this simulation environment.

Network Size	500 x 500
Number of Nodes	50
Initial Transmission	100
Movement Speed	20 m/sec
Pause Time (second)	0, 10, 20, 60,100, 200
Packet Size	60 byte
Transmission Rate	8 packets/sec
Number of Scenarios	20 / pause time
Simulation Time	100 secs

Table 3. Simulation Parameter Values

We implemented simple AODV (low level security), and then modified with the new trust based algorithm for medium level and high level security.

4.2 Average End-End Delay:

The average end-end delay at the given traffic load is shown in the Figure 2. Note that a link becomes unreliable when it is broken and/or saturated with heavy traffic. When link is unreliable the node fails delays.

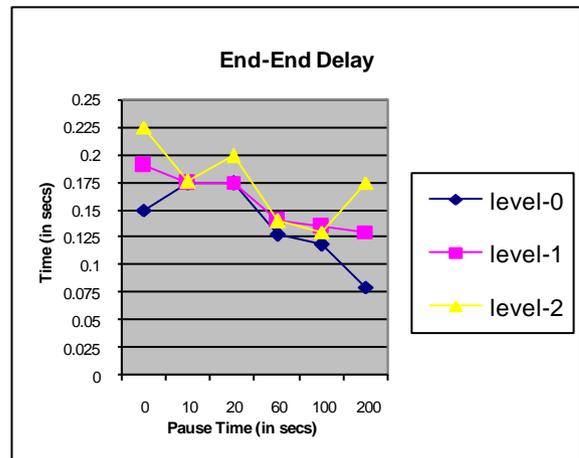


Figure 2: Average End-End Delay

From the figure, it can be seen that low security level has the lowest average end-end delay whereas at high level, which requires higher level of encryption, has the highest end-end delay. This is the price which has to be paid for information security and network reliability.

4.3 Packet Delivery Ratio:

Figure 3 shows how many packets are successfully received at the destination in the 500 x 500 networks. It shows that at high level security we have the lowest percentage of packet delivery because of obvious delay occurring due to high level cryptographic operations at the nodes.



Figure 3: Packet Delivery Fraction

It can be seen that the Packet Delivery Fraction gradually increases from around 25% at pause time 0 sec (high mobility) to 50% at pause time 200 sec (low mobility).

V. CONCLUSIONS

In this work, we proposed a solution for the application to choose the level of security it needs. Based on this level of security the application needs and the level of trust a particular node has on its neighbors, the nodes encrypt the information. So, instead of using the same kind of encryption for all the information exchanged, this protocol provides a way to limit this kind of high level of encryption to only the applications which really need them. This saves a lot of time as shown in our study. No protocol can effectively solve all existing security problems. Our proposed protocol can be easily combined with other routing protocols to detect a malicious node [5] [15] [16], and can be implemented in normal networks to high level security networks.

REFERENCES

- [1] Lokesh Bajaj, Mineo Takai, Rajat Ahuja, Rajive Bagrodia, and Mario Gerla. Glomosim: A scalable network simulation environment. Technical Report, 12, 1999.
- [2] R. Bagrodia and R. Meyer. Parsec: A Parallel simulation environment for complex system. Technical report, 1998
- [3] P. Papadimitratos and Z. Haas. Secure Routing for Mobile Ad Hoc Networks. In: SCS Communication Networks and Distributed Systems Modeling and Simulation Conference (CNDS 2002), 2002.
- [4] D. L. Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms, Communications of the ACM, 24(2):84-88, 1981
- [5] S. Yi, P. Naldurg, R. Kavets. Security-Aware Ad- Hoc Routing for Wireless Networks. Technical Reprt, UIUCDCS-R-2001-2241, UILU-ENG-2001-1748.
- [6] Onion Routing, <http://www.onion-router.net/>
- [7] S.J. Lee, M. Gerla and C.K Toh. A Simulation Study of Table-Driven and On-Demand Routing Protocols for Mobile Ad Hoc Networks, IEEE Network, Jul. 1999
- [8] M. K. Marina and S. R. Das. Ad Hoc On-demand Multipath Distance Vector Routing. In ICNP, Pages 14-23, 2001.
- [9] Marina Dupcinov, Srdjan Krco. Routing in ad- hoc Networks, Technical Report, Applied Research Lab, EEI, Ericsson, Ireland, 2002.
- [10] Y. Zhang and W. Lee. Intrusion Detection in Wireless Ad-Hoc Networks, MOBICOM 2000.
- [11] S. Zhu, S. Xu, S. Setia, S. Jajodia. LHAP: A Lightweight Hop-by-Hop Authentication Protocol for Ad-Hoc Networks. George Mason University and University of California at Irvine.
- [12] L. Zhou and Z. Hass. Securing Ad Hoc Networks. IEEE Network Magazine, 13(6), November, December 1999.
- [13] David B. Johnson, David A. Maltz. Dynamic Source Routing in Ad Hoc Wireless Networks, Mobile Computing, 1996.
- [14] Yin-Chun Hu, Adrian Perrig, David B. Johnson. Ariadne: A Secure On-Demand Routing Protocol for AHoc Networks, MOBICOM 2002.
- [15] J. Kong, Xiaoyan Hong. ANODR: Anonymous On Demand Routing with Untraceable Routes for Mobile Ad-Hoc Networks. MOBIHOC, June 2003.
- [16] Tom Goff, B. Nael, Abu-Ghazaleh, Dhananjay. S. Phatak and Ridvan Kahvecioglu, Preemptive Routing in Ad Hoc Networks. ACM SIGMOBILE, July, 2001.
- [17] J. Broch, D. A. Maltz, D. B. Johnson, Y-C. Hu, J. Jetcheva, A Performance Comparison of Mult-Hop Wireless Ad Hoc Routing Protocols, In the Proceedings of the 4th International Conference on Mobile Computing and Networking, ACM MOBICOM '98, pp. 85-97, Oct 1998.

Generation of Fractal Music with Mandelbrot Set

S.SUKUMARAN

S.G. Lecturer in Computer Science
Erode Arts College (Autonomous)
Erode – 638 009 Tamil Nadu
E-mail: prof_sukumar@yahoo.co.in

G.DHEEPA

Research Scholar
Erode Arts College (Autonomous)
Erode – 638 009 Tamil Nadu
E-mail: dheep_csc@yahoo.co.in

Abstract: Fractal is an irregular and fragmented geometric shape that can be subdivided in parts, where each part appears to be the same in all range of scale. Fractals play a central role in the realistic rendering and modeling natural phenomena in computer graphics. Fractals have infinite details. One can magnify a fractal and observe fascinating details. Many mathematical structures are fractals. In recent years many relations have been discovered between fractals and music. Fractal music is a result of a recursive process where an algorithm is applied multiple times to process its previous output. This Paper will cover some of the research that has been done on these relations. It will show how artists are currently using fractals to generate the basic melodies in their composition; the computer program that generates these melodies will be discussed.

Keywords:

Fractal, Mandelbrot set, Fractal Music, MIDI, etc...

I. INTRODUCTION

Benoit Mandelbrot invented the word fractal. Latin adjective - fractus verb – frangere means ‘to break’ to create irregular fragments ^[1]. Fractals generated by dynamical systems are called Algebraic fractals, Ex: Mandelbrot & Julia set.

Just as there are graphical representations of fractals, there are also musical representations of fractals. Music and Mathematics always had a close relationship since the time of Pythagoras and his discoveries of harmony and scales. He formulated a scientific approach to music, expressing music intervals as numeric proportions.

The most rigorous mathematical study of music in more recent years would be the system formulated by Joseph Schillinger in the 1920’s and 1930’s. Mathematics and Music are closely connected in many ways. For example, rhythm can be easily described using fractions. Pitches can be represented by real numbers. Chords can be represented by the addition of integers. In the 20th century, musical composition using mathematics by the help of computer programming began to evolve. These types of compositions are called Algorithmic Compositions which are also known popularly as "Fractal Music". Fractal Music is a musical

piece composed using fractal mathematics by means of computer programming. ^[5]

1) Notes and Numbers

Consider an elementary mapping for mathematical quantities and musical notes; match up each note in the musical chromatic scale with an ordinary number.

c1	c#1	d1	d#1	e1	f1	f#1	g1
1	2	3	4	5	6	7	8

These numbers could represent actual notes played on a synthesizer, an output onto magnetic media, or just a theoretical set.

2) 1/f noise and music

In the mid – 1970’s, an even more general mathematical study of music was performed by Richard F.Voss and John Clarke at the University of California. This time, rather than studying the structure of the music as it is written, the researchers decided to study the actual audio physical sound of the music as it is played. This was accomplished by analyzing the audio signal which, in a stereo system, would correspond to the voltage used to drive the speakers. The signal was fed through a PDP-11 computer, which then measured a quality called a spectral density.

Spectral density is often used in the analysis of random signals or noise, and is a useful characterization of the average behavior of any quantity varying in time ^[3]. Another quality, called the autocorrelation function, measures how the fluctuations in the signal are related to previous fluctuations.

The concepts of spectral density and autocorrelation are a bit difficult to grasp mathematically, but can be understood intuitively; Benoit Mandelbrot explains them in the following manner. If one takes a tape recorder and records a sound, then play it faster or slower than normal, the character of the sound changes considerably. Some sounds, however will sound exactly the same as before if they are

played at a different speed; one only has to adjust the volume to make it sound the same. These sounds are called “scaling sounds”.

II. CLASSIFICATION

The simplest example of a scaling sound is white noise, which is commonly encountered as static on a radio. This is caused by the thermal noise produced by random motions of electrons through an electrical resistance. The autocorrelation function of white noise is zero, since the fluctuations at one moment are unrelated to previous fluctuations. If white noise is recorded and played back at a different speed, it sounds pretty much the same: like a “colorless” hiss. In terms of spectral density, white noise has a spectral density of $1/f^0$.^[6]

Another type of scaling sound is sometimes called Brownian noise because it is characteristic of Brownian motion, the random motion of small particles suspended in a liquid and set into motion by the thermal agitation of molecules. Brownian motion resembles a random walk in three dimensions. Since where the particle goes next depends on its current position, Brownian motion is random but still highly correlated. Brownian noise has a spectral density of $1/f^2$.

Voss and Clerk analyzed several recordings of music and speech. They first analyzed the spectral density of the audio signal itself. This consisted of a series of sharp peaks between 100 Hz and 2 kHz, which was far from the kind of results they were seeking. Since they wanted to measure quantities that varied more slowly, they then examined a quantity they called the audio power of the music. This was proportional to the power delivered to the speakers rather than the voltage. The audio power seemed to show $1/f$ behavior, which is midway between white noise ($1/f^0$) and Brownian noise ($1/f^2$).^[3]

After Voss and Clarke found $1/f$ behavior in music, they decided to try applying these results in composing music using white, Brownian, and $1/f$ noises and compare the results. This composition technique was done in the following manner. A physical noise source was first used to provide a fluctuating voltage with the desired spectrum. This was done by using various electronic methods which could produce the desired noise. The voltages were then sampled, digitized, and stored in a computer as a series of numbers with a spectral density the same as the noise source. These numbers were then rounded and scaled and matched to notes over two octaves of a musical scale, matching the higher numbers to the notes with higher frequencies and the lower numbers to notes of lower frequencies. The process was then repeated, this time interpreting the numbers produced as duration of notes. They then turned these data into musical scores.

The three types of music were then played for several listeners, who made comments about the pieces. Most of them said that the white music was too random and the

Brownian music was too correlated. The $1/f$ music, however, seemed to sound the most like regular music.

$1/f$ noise (“one-over- f -noise”, occasionally called “flicker noise” or “pink noise”) has a power spectra $p(f)$ as a function of the frequency f behaves like ; $p(f) = 1/f^a$, where the exponent a is very close to 1 (hence the name “ $1/f$ ” noise). $1/f$ noise appears in nature all over the places, including traffic flow, radioactive decay, chemical systems, granular flow, ecological systems, human speech and music. The music derived from the $1/f$ noise is the most pleasing to the human ear.

III. MANDELBROT SET

Fractals generated from dynamical system are of algebraic type. Algebraic fractals are generated by simple iterated transformation function like $f(z) = z^n + c$ where z and c are complex numbers^[9]. The Mandelbrot set is one of the algebraic fractal types.

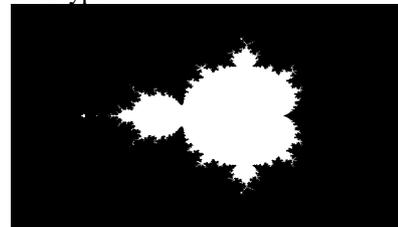


Figure 1. Mandelbrot Set
(White region – belongs to Mandel fractal set
Black region – belongs to outside the set)

(1) Algorithm - Mandelbrot Set

Step 1:

Input x_{min} , x_{max} , y_{min} , y_{max} , M and R
Where M is the maximum number of iterations
 R is radius of the circle.
Choose a complex function ($Z \rightarrow Z^2 + C$)
 $Z = x + iy$ and $C = p + iq$
 $x \in [x_{min}, x_{max}]$, $y \in [y_{min}, y_{max}]$
 $dx = (x_{max} - x_{min})/a$;
 $dy = (y_{max} - y_{min})/b$;
where a is the maximum number of columns
and b is the maximum number of rows
of the display screen.

Step 2 :

For all points (n_x, n_y) of the screen
 $(n_x = 0, 1, 2, \dots, a-1)$,
 $(n_y = 0, 1, 2, \dots, b-1)$
Go through the following routine:

Step 3 :

Set $k=0$
 $p_0 = x_{min} + n_x * dx$
 $q_0 = y_{min} + n_y * dy$
 $x_0 = y_0 = 0$

Step 4:

Calculate (x_{k+1}, y_{k+1}) , where x_{k+1} is the real part of the given complex function and y_{k+1} is the imaginary part of the complex function.

Step 5:

Calculate $r = x_k^2 + y_k^2$

- i) if $r > R$ then set black to color and go to step6
- ii) if $k = M$ then set white to color and go to step6
- iii) $r \leq R$ and $k < M$, repeat step 4

Step 6:

Plot the point $(nx, ny, color)$ and go to the next point (step2).

IV. PROPOSED METHOD

This work uses the language Visual Basic which comes with libraries and methods that are useful for creating graphical displays and music.

1) Graphing the Mandelbrot Set

The Mandelbrot set is a set of complex numbers, so we graph it on the complex number plane. First we need a test to determine if a given number is inside the set or outside the set. The test is based on the equation $Z = Z^2 + C$.

C represents a constant number, meaning that it does not change during the testing process. C is the number we are testing. Z starts out as zero, but it changes as we repeatedly iterate this equation. With each iteration we create a new Z that is equal to the old Z squared plus the constant C . So the number Z keeps changing throughout the test. We are not really interested in the actual value of Z as it changes; we just look at its magnitude.

The magnitude of a number is its distance from zero. To calculate the magnitude of a complex number, we add the square of the number's distance from the x-axis (the horizontal real axis) to the square of the number's distance from the y-axis (the imaginary vertical y axis) and take the square root. In this illustration a is the distance from the y-axis, b is the distance from x-axis, and d is the magnitude, distance from zero.

As we iterate our equation, Z changes and the magnitude of Z also changes. The magnitude of Z will do one of the two things. It will either stay equal to or below 2 forever, or it will eventually surpass 2. Once the magnitude of Z surpasses 2, it will increase forever. In the first case, where the magnitude of Z stays small, the number we are testing is part of the Mandelbrot set. If the magnitude of Z eventually surpasses 2, the number is not part of the Mandelbrot set. As we test many complex numbers we can graph the ones that are part of the Mandelbrot set on the complex number plane. If we plot thousands of points, an image of the set will appear.

We can also add color to the image. The colors are added to the points that are not inside the set, according to how much iteration was required before the magnitude of Z surpassed 2. Not only do colors enhance the image aesthetically, they help to highlight parts of the Mandelbrot set that are too small to show up in the graph.

To make exciting images of tiny parts of the Mandelbrot set, we just select an area and magnify it. Notice that each image is a detail of the center of the image preceding it. We can trace on the outline of the image and click a point on the outline to generate music.

2) Mapping to Musical notes in Mandelbrot set

Each current point (x, y) created using the algorithm is mapped to MIDI notes. MIDI is the digital note for music. The midi notes are played in real time. Mandelbrot music produces music following the sinuosities of the Mandelbrot "mountains" along straight lines. The music is recorded using sound recorder and saved for future use.

The "Draw entire area" button creates the full view of Mandelbrot graph. The "Show outline" button traces the outline of the Mandelbrot graph. The "Zoom area" button is used to magnify a certain portion of the graph. The "Generate Music" button generates music based on the point clicked on the outline. The "Stop music" button stops the music.

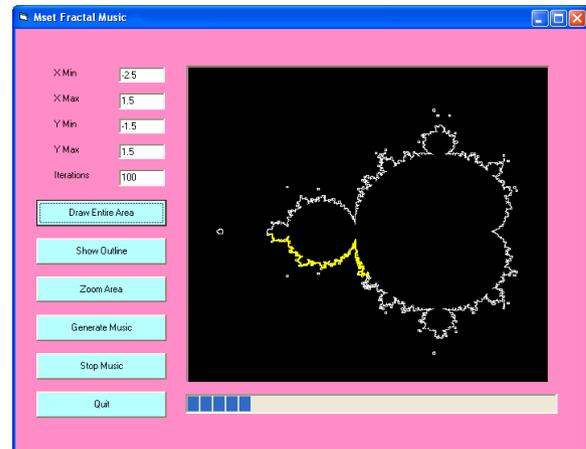


Figure 2. Mset Fractal Music



Figure 3. Mandel music sound recorder

V. RESULTS

This paper demonstrates that it is possible to generate music from fractal sets. For generating more interesting pictures, the region of the complex plane with x-boundaries -2.25 and 0.75 and y-boundaries -1.5 and 1.5 is enough. The following music files are generated.

X Min	X Max	Y min	Y max	Music
-0.75	0.5	1	1.25	M1.wav
0	0.25	.25	.75	M2.wav
-2.25	-0.5	-1.5	-0.75	M3.wav

VI. CONCLUSION

In the future, this method of generating music using fractals can be used by artists for composing basic melodies and tunes in their composition, thereby saving time. Here in this paper the music is generated using the notes of piano. In the same way, notes of different musical instruments like drums, guitar, violin etc... can be used to create different music. The method discussed in this paper can be also used to generate different fascinating fractal images and music using various fractal algorithms.

REFERENCES

- [1] Falconer, Kenneth, Techniques in Fractal Geometry, John Wiley and sons, 1997
- [2] Alice Kelley, Layering techniques in fractal art, Computer and Graphics, vol. 24, pp 611-616, 2000.
- [3] Voss Richard F and John Clarke. "1/f noise in Music: Music From 1/f Noise." J. Acoust. Soc. Am. 63(1)(1978)
- [4] Olsan, Harry F. (1967) "Music Physics and Engineering (2nd ed. (Dover Publications, New York, NY.)
- [5] Shillinger, Joseph. Schillinger System of Music Composition, Da Capo, New York.
- [6] Gardner, Martin. "Mathematical Games." Scientific American April 1978
- [7] Barnsley, R. L. Devaney, B.B. Mandelbrot, H. Peitgen, D. Saupe, P. F. Vos, "The Science of fractal Images", Springer-Verlag, 1988.
- [8] Casey. D. Stephen, Nicholas F. Reingold, "Self-similar Fractal set: Theory and Procedure", IEEE Computer Graphics and Applications, pp: 73-82, 1994.
- [9] Mandelbrot. B. B. "The Fractal Geometry of Nature", W. H. Freeman and Company, 1982.
- [10] Peitgen H-O, Richter PH, "The Beauty of Fractals", Springer, 1986.
- [11] Pohlmann, Ken C. (1991) Advanced Digital Audio. Howard Sams, CA Arms, In
- [12] Schroeder, Manfred R. (1991) Fractals, Chaos, and power Laws. W.H. Freeman and co..
- [13] Chaos and Fractals: New Frontiers of science by H.O. Peitgen, H. Jurgens, D. Saupe.
- [14] Robert L. Devaney and Linda Keen, editors, "Chaos and fractals: The mathematics behind the computer graphics", American Mathematical Society, 1989.
- [15] Robert L. Devaney, "Chaos Rules", Research Article, Sep 2003.
- [16] Kindermann, Lars. Musinum- The Music in the Numbers. Available on-line at <http://www.forwiss.uni-erlangen.de/~kinderma/musinum.html>.
- [17] Rochon D, A generalized Mandelbrot set for bicomplex numbers. Fractals, pp. 355-368, 2000.

Performance Analysis & QoS Guarantee in ATM Networks

Parag Jain¹, Sandip Vijay^{!!}, S. C. Gupta^{!!!}

¹Doctoral Candidate, Bhagwant Univ. & Professor, MCA Dept., Roorkee Institute of Technology, Roorkee, India.

^{!!}Wireless Computing Research Lab, Department of Electronics & Computer, I.I.T. Roorkee, India.

^{!!!}Professor Emeritus, Department of Electronics & Computer, I.I.T. Roorkee, India

Abstract- The performance of different ATM switch buffer queuing schemes for round-robin and weighted round-robin didn't have too much difference. This paper presents the discrete-event simulation provides detailed, accurate network simulation results and it observed a wide variety of network statistics. The software simulation package, OPNET, which specializes in discrete-event simulation of communication systems, has many attractive features and can simulate large communication networks with detailed protocol modeling and performance analysis.

I. INTRODUCTION

1) Introduction to ATM

The ATM (Asynchronous Transmission Mode) Technology is the merged result of Packet Switching (Packet switching is a store and forward data transmission technique in which a message is broken up into small parts each called packet.) and TDM (Time Division Multiplexing is a method of putting multiple data streams in a single signal by separating the signal into many segments, e.g. having a very short duration) These techniques are clearly described by Prycker et al [1]. The first 5 bytes contain cell-header information, and the remaining 48 bytes contain the "payload" (user information) [2].

2) Service Categories in ATM Networks

ATM Network is designed to carry different type of traffic at the same type. Traffic could be voice, video or IP traffic. Internally all different traffic is carried as 53 byte cells. However, handling of traffic depends on the characteristics and requirement of the traffic. There are four categories of Service; the QoS Parameters for those categories are [3] as follows:

Constant Bit Rate (CBR)

Variable Bit Rate (VBR)

- Real Time VBR and Non Real Time VBR
- Real Time Variable Bit Rate (Rt-VBR)
- Non Real time Variable Bit Rate (Nrt-VBR)
- Available Bit Rate (ABR)
- Unspecified Bit Rate (UBR)

3) Quality of Service (QoS) Parameters in ATM Networks

Primary objectives of ATM are to provide QoS Guarantees while transferring cells across the network. There are mainly three QoS parameters specified for ATM and they are indicators of the performance of the network.

Cell Transfer Delay (CTD)

The Delay experienced by a cell between the first bits of the cell is transmitted by the source and the last bit of the cell is received by the destination. This includes propagation delay, processing delay and queuing delay at switches. Maximum cell transfer delay (Max CTD) and Mean cell Transfer Delay (Mean CTD) are used.

Peak to peak Cell Delay Variation (CDV)

The difference of the maximum and minimum CTD experienced during connection. Peak to Peak CDV and instantaneous CDV are used.

Cell loss Ratio (CLR)

The percentage of cells lost in the network due to error or congestion that is not received by the destination. CLR value is negotiated between user and network during call set up process and is usually in the range of 10^{-1} to 10^{-15} .

4) Traffic Management

A key advantage of ATM is that it can carry traffic of different type like voice, video, data etc. different type of traffic necessitates a mechanism that can fairly manage the traffic coming on different virtual connection of different type. Traffic management in ATM does this by appropriately providing QoS for different type of traffic. By doing so traffic management has following Components

Negotiations of a Contract Between end System and the Network

To make the QoS job easier for the Network, ATM forum define 5 different QoS classes.

Five different classes are Class 0, Class 1, Class 2, Class 3, Class 4. which corresponds to best effort applications, CBR circuit emulation applications, VBR video and audio applications, connection-oriented data, and connectionless

data respectively. For each Specified QoS class, the network specifies an objective value for each QoS parameters.

Connection Admission Control

ATM network uses Connection admission control to reserve the bandwidth for each virtual connection on atm network. Every time a new connection is made, the network checks to see if it can fulfill the QoS requirements and the traffic characteristics of the incoming connection.

- **Peak Cell Rate(PCR)**

Define an upper bound on the traffic that can be submitted by the source into the ATM network. PCR is defined in terms of T, where T is the minimum inters cell spacing in seconds. This is needed for CBR traffic.

- **Sustainable Cell rate(SCR)**

It is the upper bound on the average rate that could be sent over a period on an ATM connection. SCR is basically measure of bursty traffic. SCR is needed for VBR services as it enables the network to allocated resources efficiently.

- **Maximum Burst Size(MBS)**

MBS is the maximum burst size that can be sent continuously at PCR. If the cells are presented to the network at MBS interspersed by idle time period, then at no time overall rate should exceed the SCR.MBS is also specified for VBR sources.

- **Minimum cell rate**

MCR species the minimum rate that should be allocated to an ABR source by the network. MCR make sure that ABR source never have to transmit at rate lower than MCR.

Traffic Policing

The Incoming traffic on a virtual connection is measured by traffic policing component and it discards the traffic that exceeds the negotiated parameters specified in the contract. Traffic policing employs Generic cell rate Algorithm (GCRA) ,which is also commonly known as leaky bucket algorithm. Leaky bucket algorithm checks the rate at which traffic arrives on a virtual connection. And if the arrival doesn't conform to the contract then it either marks them as potential candidates for discard during congestion or if arrival rate too high, it immediately drops them. Cells could be marked as potential candidate for discard by setting the CLP bit in the cell. CLP=1 makes them likely candidates to be dropped in case of congestion. Policing is usually used for VBR traffic where source is allowed to send burst of traffic over a period of time.

Traffic Shaping

Traffic shaping shapes the traffic coming on ATM interface that doesn't conform to the traffic contract and then it ensures by adjusting the incoming rate that traffic reaches the destination without getting discarded. Traffic shaping does this by buffering the traffic and sending it into the network at some later time.

5) *Performance Parameters*

Following parameters characterize the performance of ATM systems

Throughput

This can be defined as the rate at which the cells depart the switch measured in the number of cell departures per unit time. It mainly depends on the technology and dimensioning of the ATM switch. By choosing a proper topology of the switch, the throughput can be increased.

Connection Blocking Probability

Since ATM is connection oriented, there will be a logical connection between the logical inlet and outlet during the connection set up phase. Now the connection blocking probability is defined as the probability that there are not enough resource between inlet and outlet of the switch to assure the quality of all existing as well as new connection.

Cell Loss Probability

In ATM switches when more cells than a queue in the switch can handle will compete for this queue, cell will be lost. This cell loss probability has to be kept within limits to ensure high reliability of the switch. In Internally Non-Blocking switches, cell can only be lost at their inlets/outlets. There is also possibility that ATM cell may be internally misrouted and they reach erroneously on another logical channel. This is called Insertion Probability.

Switch Delay

This is the time to switch an atm cell through the switch. The typical values of switching delay range between 10 and 100 µsecs. This delay has two parts .Fixed switching delay and queuing delay fixed switching delay is because of internal cell transfer through the hardware.

Jitter on the Delay

This is also called Cell delay variation(CDV) and this is denoted as the probability that the delay of the switch will exceed a certain value. This is called a quantile and for example, a jitter of 100 µsecs at a 10exp-9 quantile means the probability that the delay in the switch is larger than 100 Microsecs is smaller than 10exp-9.

Asynchronous Transfer Mode (ATM) is a connection-oriented packet switching technique that is universally accepted as the transfer mode of choice for Broadband Integrated Services Digital Network. This report describes key features of the ATM network and some relative simulation work we have done by OPNET.

Since the project is emphasize in simulation work by OPNET, we'll just simply introduce the basic principles of ATM:

- ATM is considered as a specific packet oriented transfer mode based on fixed length cells. Each cell

consists of a 48bytes of information field and a 5bytes of header, which is mainly used to determine the virtual channel and to perform the appropriate routing. Cell sequence integrity is preserved per virtual channel.

- ATM is connection-oriented. The header values are assigned to each section of a connection for the complete duration of the connection. Signaling and user information are carried on separate virtual channels.
- The information field of ATM cells is carried transparently through the network. No processing like error control is performed on it inside the network.
- All services (voice, video, data,) can be transported via ATM, including connectionless services. To accommodate various services an adaptation function is provided to fit information of all services into ATM cells and to provide service specific functions (e.g. clock recovery, cell loss recovery ...).

II. SIMULATION BY OPNET

1) Introduction:

OPNET (Optimized Network Engineering Tool) provides a comprehensive development environment for the specification, simulation and performance analysis of communication networks.

OPNET provides four tools called editors to develop a representation of a system being modeled. These editors, the Network, Node, Process and Parameter Editors, are organized in a hierarchical fashion, which supports the concept of model level reuse. Models developed at one layer can be used by another model at a higher layer.

2) ATM Model Features:

Signaling Support : Signaling is provided for point-to-point, full-duplex, Switched Virtual Circuit (SVC), Soft-Permanent Virtual Circuit (SPVC) and Soft-Permanent Virtual Path (SPVP).

Traffic Control: Traffic control includes Call Admission Control (CAC) and Usage Parameter Control (UPC). Traffic Control is based on specific service category, traffic parameters (PCR, SCR, MCR, MBS) and QoS parameters (ppCDV, maxCTD, CLR).

Buffering: Buffers can be configured at each switch for various QoS levels. A QoS level is made up of the QoS category (CBR, rt-VBR, nrt-VBR, ABR, UBR), the QoS parameters, (ppCDV, max CTD, CLR), and the traffic parameters.

3) ATM Node Models:

The ATM model suite contains several client and server node models, which can be subdivided into the following categories: workstations and servers, uni-clients and uni-servers, uni-sources and uni-destinations, and intermediate switching elements (such as clouds and switches). The ATM nodes can be found in the object palettes with an "atm" prefix: atm, atm_advanced, atm_lane, and atm_lane_advanced. In this simulation, we choose atm_uni_client for applications that run directly over ATM.

Uni-clients and uniservers: atm_uni_client, atm_uni_server

These client node models feature an application layer that resides directly over the ATM layer. Unlike the ATM workstation node model, the ATM uni-client model establishes a separate ATM connection for each application task. Examples of application tasks are sending an e-mail, downloading a .je, and making a voice call. ATM uni-clients can be used only with ATM uni-servers, which are capable of supporting all of the application services. The ATM uni-client and uni-server node models are located in the atm_advanced object palette.

4) ATM Model attributes:

The intermediate and advanced ATM nodes have several attributes that can be used to specify ATM configuration details. Some of the important ATM model attributes we concern in our simulation are:

(1) Traffic Contract:

This attribute specifies the traffic contract used by the application layer when it sends traffic over an ATM stack. Although the application layer includes data traffic, signaling traffic and IP/ATM routing traffic, only data traffic has a configurable traffic contract. The Traffic Contract attribute has 3 parts: the Category, the Requested Traffic Contract, and the Requested QoS.

- **Category:** This attribute specifies the service category used by the application. OPNET supports all five categories specified by the ATM Forum Traffic Management Specification 4.0: CBR, rt-VBR, nrt-VBR, ABR, and UBR. For a call to be admitted by call admission control, there should be at least one path to the destination where all nodes support the requested service category.

- **Requested Traffic Contract:** This attribute specifies the traffic parameter settings for the connection. The Requested Traffic Contract allows you to specify the peak cell rate (PCR), minimum cell rate (MCR), sustainable cell rate (SCR), and mean burst duration (MBS) in the incoming and outgoing directions. During call admission control, these requested values are compared to the supported parameters on all intermediate nodes.

- **Requested QoS:** This attribute specifies the application's requested Quality of Service, which includes the peak-to-peak cell delay variation (ppCDV), the maximum cell transfer delay (maxCTD), and the cell loss ratio (CLR). During call admission control, these requested values will be compared to the supported parameters on all intermediate nodes.

(2) Port Buffer Configuration:

This attribute is used to specify supported parameters and to configure buffers on each port of a node. The configuration specified in this attribute applies to all ports of the node.

- **Queue Number:** This attribute specifies the queue index. To automatically assign indices to the queues, you can use the Per VC setting. Alternatively, you can assign each queue a unique queue number. The queue number is used to identify the queue being monitored for certain statistics (such as queue length).

- **Queue Parameters:** This attribute allows you to specify the amount of bandwidth that is allocated to a specific queue.

- **Max_Avail_BW (% Link BW):** This is the maximum bandwidth available to this queue. It is calculated as a percentage of the link bandwidth. For CBR calls, this attribute regulates the maximum bandwidth reserved and hence guarantees this bandwidth as well.

- **Min_Guaran_BW (% Link BW):** This is the minimum guaranteed bandwidth expressed as a percentage of link bandwidth. For non-CBR calls, this attribute defines the bandwidth reserved. For example, for a rt-VBR call, SCR is the minimum guaranteed bandwidth. The value specified for minimum guaranteed bandwidth is equal to the weight of this queue when the ATM QoS Priority Scheme attribute is set to weighted round-robin.

- **Size:** This attribute determines the number of cells in the queue.

- **Traffic and Qos Parameters:** These attributes are used when selecting a buffer for a call that is waiting for admission through a port in a node. A buffer is selected if both of the following requirements are met:

- The traffic parameters (PCR, SCR, MCR, and MBS) of the incoming call are less than or equal to the value specified in the Traffic Parameters attribute.

- The QoS parameters (maxCTD, ppCDV, CLR) of the incoming request are greater than or equal to the values specified in the QoS Parameters attribute.

(3) ATM Switching Speed:

This attribute specifies how fast a cell is switched through the core switching fabric. This speed is specified in cells/sec.

(4) ATM QoS Priority Scheme:

This attribute specifies the servicing scheme for the queues. Two types of queuing schemes are available: round-robin and weighted round-robin.

- **Weighted round-robin scheme:** queues are serviced depending on the weights assigned to them. Weights are determined according to the Minimum Guaranteed Bandwidth attribute (in ATM Port Buffer Configuration >Queue Parameters attribute) of each queue parameter. This scheme ensures that the guaranteed bandwidth is reserved.

- **Round-robin scheme:** all queues have the same priority and therefore have the same chance of being serviced. The link's bandwidth is equally divided amongst the queues being serviced.

(5) Routing attributes:

Switches in the ATM model suite use a dynamic routing protocol, ATM Distance Vector Routing, which is implemented as a distributed, asynchronous adaptation of the Bellman-Ford shortest path algorithm. When the ATM signaling layer receives a call setup request, the source node finds a route to the call's destination. The following attributes are used to configure this routing protocol:

ATM Routing Update Interval. This attribute specifies the time between regular routing updates. Routing tables are periodically updated to ensure that all nodes are aware of the latest topology changes.

ATM Active and Passive Failure Detection Modes. Failures and recoveries in the network must be detected by nodes adjacent to the failure or recovery point. These (adjacent) nodes must then inform other nodes in the network of the failure or recovery.

- **active failure detection mode:** the routing process detects a neighbor node or link failure/recovery and updates its routing tables immediately. The node sends out route advertisements that reflect its updated routing table.

- **passive failure detection mode:** failure is detected implicitly when no route costs have been received in two or more route update periods.

5) Simulation results:

The fig1 is the ATM network we constructed by OPNET. This ATM network we are simulating consists of several ATM switches, server and clients(the reason why we didn't

use more ATM switches is the simulation time increase beyond our computer can afford), we use OC3 link to connect the network for supporting maximum 155Mbps traffic.

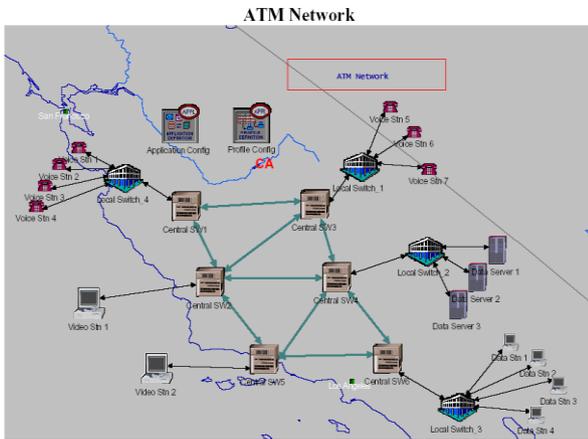
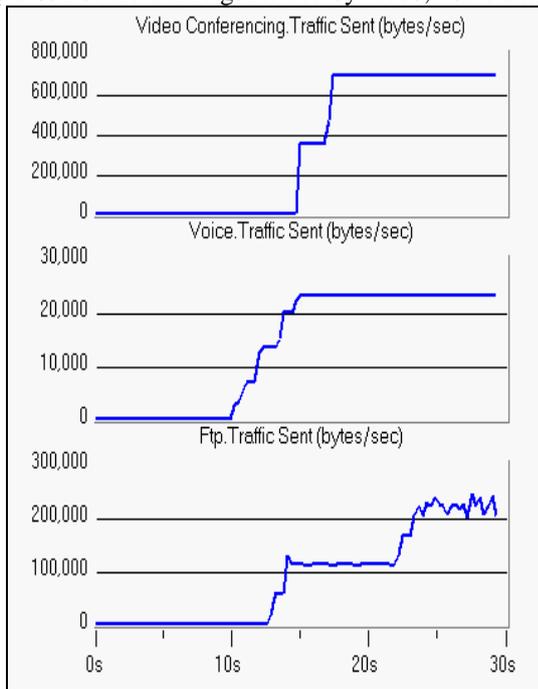


Fig1. ATM network structure

We can simply adjust the traffic load by changing the *Traffic Scaling Factor* in *Configure Simulation* menu before we run the simulation every time. The three traffic components: video, voice and data are generated; we use *rt_VBR* for video, *CBR* for the voice, and *ABR* for data traffic. However, the ratio of the three kinds of traffic is difficult to set to exactly 30%, 40% and 30%. Fig2 is the comparison of the traffic generated by video, voice and data.



The following tables are the statistic results of data, video and voice services.

Statistic	Average	Maximum	Minimum
Ftp Download Response Time (sec)	0.072	0.109	0.059
Ftp Traffic Received (bytes/sec)	84,309	285,440	0
Ftp Traffic Received (packets/sec)	30.7	96.7	0.0
Ftp Traffic Sent (bytes/sec)	84,814	242,267	0
Ftp Traffic Sent (packets/sec)	30.8	93.3	0.0
Ftp Upload Response Time (sec)	0.070	0.107	0.059

Statistic	Average	Max.	Min.
Video Conferencing Packet End-to-End Delay (sec)	0.0708	0.0709	0.0708
Video Conferencing Traffic Received (bytes/sec)	313,018	691,200	0
Video Conferencing Traffic Received (packets/sec)	18.1	40.0	0.0
Video Conferencing Traffic Sent (bytes/sec)	315,350	691,200	0
Video Conferencing Traffic Sent (packets/sec)	18.3	40.0	0.0

The shortest path routing algorithm we are using in OPNET is a dynamic routing protocol, ATM Distance Vector Routing. Since the OPNET only support two types of queuing schemes: *round-robin* and *weighted round-robin*, and the simulation is limited by the time constriction, we can only compare the two queuing schemes running for 30

seconds processing time(the actual time for the simulation is much longer, may take half an hour to several hours). The following two tables are the results for running at two different queuing schemes.

Statistic	Average	Maximum
ATM Call Blocking Ratio (%)	0	0
ATM cell Delay (sec)	0.00370	0.00755
ATM throughput (bits/sec)	1,390,592	3,064,107

Table1. Statistic results in *round-robin*

Statistic	Average	Maximum
ATM Call Blocking Ratio (%)	0	0
ATM cell Delay (sec)	0.00387	0.00910
ATM throughput (bits/sec)	1,390,239	3,064,453

Table2. Statistic results in *weighted round-robin*

III. CONCLUSION

The software simulation package, OPNET, which specializes in discrete-event simulation of communication systems, has many attractive features and can simulate large communication networks with detailed protocol modeling and performance analysis.

In our study, the performance of different ATM switch buffer queuing schemes for *round-robin* and *weighted round-robin* didn't have too much difference.

In conclusion, discrete-event simulation provides detailed, accurate network simulation results and can observe a wide variety of network statistics. However, this method of simulation generally requires ample significant time and memory.

REFERENCES:

- [1]Prycker,M.de, Peschi,R., and Landegem,T.Van,"B-ISDN and the OSI Protocol reference Model",IEEE Network Magazine,Vol 7,no.2.pp.10-18, March 1993.
- [2] Prycker,M.de,(1993),"Asynchronous Transfer Mode solution for Broadband ISDN",IInd Edition.Chichester England:Ellis Horwood. Pp. 112-18,157,159-68,289,323.
- [3]The ATM Forum,Technical Committee,(March 1999),"Traffic Management ,Version 4.1," pp. 43-45,af-tm-021.000.
- [4]http://www.microsoft.com/windows2000/en/server/help/default.asp?url=/windows2000/en/server/help/ATM_und_atmqos.htm
- [5]<http://cne.gmu.edu/modules/atm/xtt.html>
- [6]Network simulation with OPNET: <http://www.informs-cs.org/wsc99papers/043.PDF>

Survey of Forest Fire Simulation

Qasim Siddique
Foundation University
Islamabad, Pakistan
Qasim_1987@hotmail.com

Abstract- Fire modeling is used to understand and to predict possible fire behavior without getting burned. Fire models are used in different aspect of fire management. The increase in the number of forest fires in the last few years has forced governments to take precautions. Beside prevention, early intervention is also very important in fire fighting. If the fire fighters know where the fire will be in sometimes it would be easier for them to stop the fire. Therefore a big need for simulating the fire behavior exists. In this paper we present the survey of various forest fire simulations. The main goal is to determine how forest fires develop in different manners under specific conditions.

I. INTRODUCTION

Forest fires are considered a potential hazard having physical, biological, ecological and environmental consequence. Almost 6-7 million Km² of forests have been lost in less than 200 years due to wildfires. Fire disturbance has important ecological effect in many forest landscapes. Today, virtually all forest fires are man made intentional or accidental. Computer based tools entail a breakthrough advance in the forest fire simulation. Since the incipient attempts to calculate the fire ignition and estimate the fire behaviors over complex terrain and non-homogenous forest fuel pattern, the computer application has served well to forest fire defense services planners and managers in the decision making process about what, where, when and why to use fire prevention resources and fire fighting forces.

The paper is organized as follows: Section 2 presents a detailed description of the classification of the forest fires models. Section 3 presents the detail of current forest fire simulation system Section 4 present the discussion of computer aided decision support system are used to support wild land fire management decision. Finally conclusions are given in section 5.

II. CLASSIFICATION OF FOREST FIRE MODELS

Numerous fire spread model have been proposed. They can be grouped into

A. Empirical Model

These models are predicting more portable fire behavior from average condition and accumulating knowledge obtained from laboratory and outdoor, experimental fires or historical fires.

B. Semi-Empirical Model

These models are based on a global energy balance and on the assumption that the energy is transferred to the unburned fuel are proportional to the energy released by the combustion of the fuel.

C. Physical Model

These model are based on physical principle, have the potential to accurately predict the parameter of interest over a broader range of input variable then empirically based model

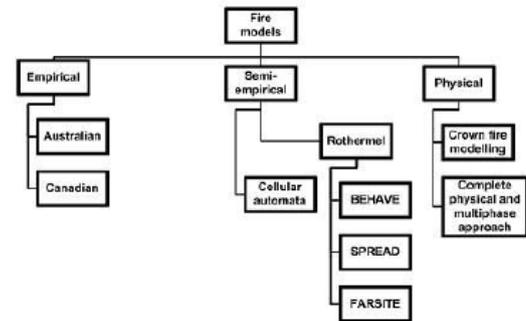


Figure 1 Classification of Forest Fire Model

D. Cellular automata Approach

Cellular automata paradigm is widely used for modeling and simulating complex dynamical system whose evolution depends exclusively on the local interactions of their constituent parts. A cellular automaton involves a regular division of the space in cells, each one characterized by a state that represents its actual condition. The state change according to a transition rule which depends on the state of neighbor cells and of the cell itself. At the starting time cells are in the states describing initial condition and subsequently the cellular automata evolves changing the state of all the cells simultaneously at discrete steps according to the transition rule.

E. Parallelization Approach

The parallelization of simulation model approach can be done with data parallelism the same algorithm can be applied to several set of data independently. This entails that the same problem has very regular structure and the same operation can be done on the different part of the problem. This mean in the case of fire simulation. The movements of each line of fireline can be calculated independently.

III. FOREST FIRE SIMULATION SYSTEMS

The fight against the forest fires emergencies requires useful tools to predict the propagation and behavior of forest fire in order to take the best decision. It means it is necessary to know the propagation and behavior of forest fire

A. Parallelization of forest fire propagation Simulation

In forest fire propagation model Josep jorba and Tomas have used the model define by Andre/Viegas 1994[1]. The main goal of this model is to study the movement of the fire line. The operational cycle of this model consist in calculating the next position of the fire line. Considering the current fire line position. To reach this goal the model is divided in to a local fire model and global fire spread model.

The local fire spread model calculates the movement of each individual section of the fire line and then the global model calculate the total fire line applying an aggregation process. The local fire model takes in account the static and dynamic conditions. The dynamic condition must be calculated before the local model can calculate the movement of the section. The global model allows the partitioning of the fire line into a set of sections. In order to calculate the movement of the section it is necessary determine the calculation of the propagation speed.

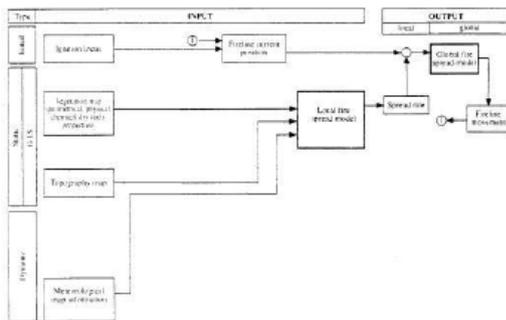


Figure 2 Simulation Components of System of Parallelization of forest fire propagation Simulation [2]

The calculation of the each section can be done parallel using data parallelization approach. In global model the fire line is composed of set of independent model. The section can be represented numerical in the form of arc. Therefore the calculation of these section can de distributed among resources of parallel machine.

B. Burn: A simulation for forest fire propagation

The burn model of the fire spread is based on the Rothermel equation [3]. The BURN has been implemented in the parallel language of the FROTRAN 90s.

The burn space of a forest fire is the area in which a fire can potentially spread the space is usually bounded with some sort of naturally bound (Mountain est.) in this particular simulation the burn space consist of $100 * 100$ matrix each cell in the matrix represent a square are of 20 feet long and 20 feet wide each cell represent 400 square feet of land.

Entire burn space consists of 10000 of this cell creating a total of 40000 square feet.

1) BRUN Input

BRUN take the input in the form of ASCII. The program accept the five input files [4]

_ A Terrains File

Contain Value of every cell of the Burn Space

_ A Moisture Data File

Contain the data of each cell which represent the Fuel Moisture of each cell

_ A Elevation Data File

_ A Wind Data File

_ Ignition Data File

The calculation of the burn model is based on the Rothermel Equation

C. 3D wildfire simulation system

In the model of 3D wildfire simulation Kijanc kose have used the using FireLib [5] using the function define in the library and some extension of these function the proposed algorithm calculate the propagation of fire in 2D. FireLib uses the algorithm defined in the BEHAVVA fire model "BEHAVWE was developed by Andrews[6] 1986 by the U.S Departments of Agriculture forest services in mid 80'S and Consist a very popular tool for predicting forest fire propagation. Beside the fuel parameter environment are very important in the determination of the fire propagation especially wind speed, humidity of the weather condition, slope and aspect of the terrain some of the information is obtained in real fire but some of the information is obtained from the local and global sources. Beside weather condition terrain is another information parameter for fire propagation, slope and aspect of the area affect the propagation speed and direction fire propagation faster in the uphill and slowly in downhill direction. Aspect determinist facing to the wind direction on the effect of the wind on the fire increased.

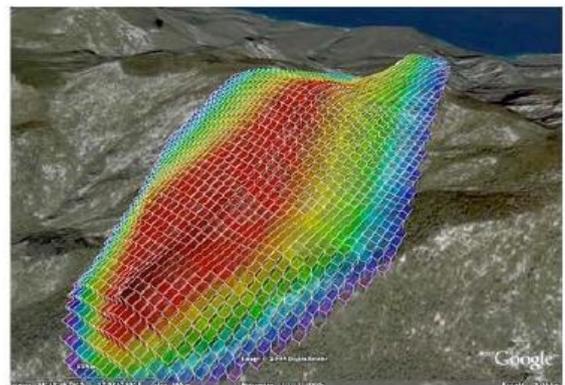


Figure 3 Ignition times using color coding red and its tone show the areas which will burn first while blue area its tones show the areas which will burn later [8]

The simulator can calculate the propagation of the time on a Landscape with varying condition. FireLib divide the area of

interest into cell. Each of cells can have its respective fuel type, aspect, slope etc. However there parameter are assume to be constant with respect to the ignition times of it neighbors cell is performed. The propagation of fire from one cell to another depends on the ignitability of the cell. This calculates yield an ignition time instant as well as an estimated flame length. However as the time increase the fire propagation further, some of the parameter in the cell may be change flame length is calculate once want the cell is ignited.

Calculation of surface parameter using height and coordinate value of point on the surface these value can be obtained for this calculation we have used the "Horn Methods[7]" (Horn 1981) due to its reduced memory and computational required.

D. Forest fire spread modeling using cellular automata approach

Ljiljana Bodrozic forest fire spread method is based on the cellular automata and is belong to semi empirical. Landscape can be represented as cellular automata. It is possible to apply cellular automata formations to a number of landscape diffusion processes such as forest fires. [9]

The most common approach for fire modeling has been simulate fire growth as a discrete process of ignition across a regularly spread landscape grids of cells. Each cell represent a fixed surface area and has a attribute that correspond to environmental feature

Computational methods are used to automata the application of fire shaped model to non uniform condition by assuming local uniformly

This Model assign a numerical value to each cell following

- _ A burning cell has a value of 3
- _ A burned cell has a value of 2
- _ A growing cell has a value of 1
- _ The state where it can be ignited has a value 0

In order to implement cellular automata model on specific area we need to obtained input data required by the model we need a matrix grid where each cell has assigned value that represents parameter that affect forest fire. Each cell represents a tree or no tree and can burn or not.

E. CARDIN 3.0

CARDIN 3.0[10] a new tool for fire fighting simulation on a PC computer is presented. The general CARDIN system uses a square scenery usually 4 to 10KM, that is on to 400 * 400 small cell., for each cell data about elevation, slope , aspect, fuel model land use and wind direction and speed are considered other parameters are average for all the scenery the fuel moisture is given for each of the 13 fuel models with out consideration of local humidity condition initial fire focuses are point sized, corresponding to one single cell to line shaped as it happened in already developed fires

The general procedure for the definition of fire shape is based on a spread law for each burring cell according to its physical data that is projected to the eight surrounding cell. The simulation process take place over a digital raster layer of information in this way slope and aspect, fuel model and wind direction and intensity is known easily by X, Y, Z the program, it self generate a new layer of information including flame length maximum rate of spread, direction of max rate of spread, fire line intensity time at which fire is expected to reach the cell and residence time of flame and ember.

The program is completed with a useful local fire projection module called FAST (Fire Analysis and Simulation Tool). It performs very fast simulations of scenery portions of 100x100 cells giving accurate results about fire line position over time.

F. Modeling Fire Spread under Environmental Influence Using a Cellular Approach

The model consists of a number of base components first it has an artificial world with 2 dimensional where bushes are placed randomly across the landscape a cell state could correspond to a number of environmental factor such as fuel and land height. A fire which is regarded as an individual in such an artificial world can be generated at a particular cell on such a landscape or many fires can be created along one side of the artificial world once started fires can then spread into their neighborhood by looking around locally. This model takes into account some of the most important influential factor contributing to the development and spread of fire. The individual in the model in this case will be having number of factor. Our goal is to determine how the fire developed in different manners from initial condition (Rapidly spread out of control, die permanently) including the factor (Bush density Flammability, Heat condition, Land Height, wind speed and it direction).

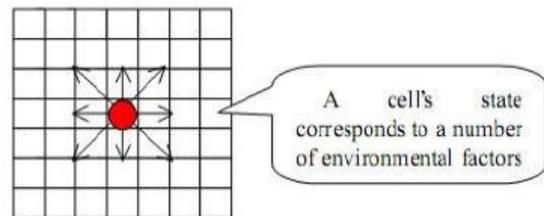


Figure 4 A fires can spread into its neighborhood on a 2-dimensional artificial world, depending on some predefined local conditions (environmental factors) in these neighboring cells [13]

This model is implemented in SWARM [12] (a general purpose simulation framework that provides a set of standard tools for simulating and analyzing complex systems exhibiting highly decentralized architecture such as a multi-agent system) a user prior to the start to simulation run can set all of the factor. The user is able to allow

variation in wind magnitude and speed during the run and can set how much variation will occur.

IV. DISCUSSION

Many computer aided decision support system are used to support wild land fire management decision. They are used for such application designing fire prescriptions, projecting the growth of a wild fire assessing the role of fire in the eco system and developing budges. Theses system have been developed over a period of 30 years by various group and individuals and targeted to meet specific need. [14] An important step in creating a simulation model is to calculate the internal simulation parameters against fire behaviors model for each fuel type present in the landscape.

V. CONCLUSION

Our study examines and classifies various model of forest fire spreading based on 2D and 3D grids of cells The simulation of forest fire propagation involves several research fields and the cooperation among researchers of these different fields is important to develop more accurate models, which reproduce the fire's behavior in a more realistic way. Moreover, the simulation of these complex models should be fast in order to predict the fire behavior in advance and use this information to decide which actions should be taken to control fire propagation. These accurate models require high performance capabilities in order to provide the results in a satisfactory time. Distributed computing provides the required computing capabilities at a relatively low cost.

REFERENCE

- [1] Andre, J.C.S, Viegas, D.X. (1994): A Strategy to Model the Average Fireline Movement of a light-to-medium Intensity Surface Forest Fire, in: Proc. of the 2nd International Conference on Forest Fire Research, Coimbra, Portugal, pp. 221-242
- [2] Josep Jorba , Tomas Margalef, Emilio Luque , Jorge Campos da Silva Andre and Domingos Xavier Viegas Parallel Approach to the Simulation of Forest Fire Propagation. In: Proceedings Environmental Communication in the Information Society. 16th International Conference "Informatics for EnvironmentalProtection", Vienna University of Technology. pp. 69-81. (2002)
- [3] Rothermel , R "A Mathematical model for predicting fire spread in wild land fuels" Res pap INT-155,Ogden, UT: U.S Department of Agriculture, Forest Service, Intermountain forest and Range Experiment Station,1972
- [4] Veach, M. S, Coddington, M., Fox, G. C. BURN: A Smulation of Forest Fire Propagation. 1994
- [5] fireLib, 2008. FireLib software implementation and documentation,<http://www.fire.org/index.php?option=content&task=category§ionid=2&id=11\&Itemid=29> (accessed 20 April 2008).
- [6] Andrews, P. L., 1986. Behave: Fire Behavior Prediction and Fuel Modeling System - BURN Subsystem Part 1, USDA Forest Service General Technical Report INT-194, 1986.
- [7] Horn B.K.P., 1981, Hill shading and the reflectance map, Proceedings of the I.E.E.E. 69, 14.
- [8] K. Kose, N. Grammalidis, E.Yilmaz,E. Cetin: "3D Wildfire Simulation System", ISPRS2008, Beijing, China, July 2008.
- [9] Antonio S. Camara, Francisco Ferreira, Spatial Simulation Modeling, Gasa, 1998
- [10] Caballero, D, J. Martinez-Millan, J. Martos, and S. Vignote. 1994. CARDIN 3.0, a model for forest fire spread and fire fighting simulation. Vol.1: 501. In proceedings of the 2nd Int. Conf. on Forest Fire Research. November 21-24, Coimbra, Portugal. Domingos Xavier Viegas, publ. 508 p.
- [11] Rothermel, R. (1983), How to predict the spread and intensity of forest and range fires, Gen. Tech. Rep. INT-143. Ogden, UT: U.S. Department of Agriculture, Forest Service, Intermountain Forest and Range Experiment Station.
- [12] Stefansson, B. (1997), Swarm: An Object Oriented Simulation Platform Applied to Markets and Organizations, Evolutionary Programming VI, Lecture Notes in Computer Science, edited by Angeline, P., Reynolds, R., and Eberhart, R.Vol.1213, Springer- Verlag, New York.
- [13] Li, X., Magill, W., 2001. Modeling fire spread under environmental influence using a cellular automaton approach. Complexity International 8, 14pp. {<http://www.complexity.org.au/ci/vol08/li01/>}.
- [14] MacGregor, Donald G.,(2004) An Inventory of Models, Tools and computer Application for wild land fire management Decembers 2004

Detecting Redundancy in Biological Databases – An Efficient Approach

Mrs.C.Sumithiradevi, PhD Scholar, R&D Department, Bharathiar University, Coimbatore, India
(Telephone: +91-997909236, E-mail: sumithradevic@yahoo.co.in)

Dr.M.Punithavalli, Director, Department of computer science, Sri Ramakrishna College of Arts and Science for Women, Coimbatore, India. (Telephone: +91-9843281552, E-mail: mpunitha_srcw@yahoo.co.in)

Abstract- Biological databases store data about molecular biological entities such as genes, proteins, diseases, etc. The main purpose of creating and maintaining such databases in commercial organizations is their importance in the process of drug discovery. As databases become more pervasive through the biological sciences, various data quality concerns are emerging. Biological databases tend to develop data quality issues regarding data redundancy. Due to the nature of this data, each of these problems is non-trivial and can cause many problems for the database. For biological data to be corrected, methods must be developed to handle the biological data. This paper discusses the biological database problems and introduces new methods to help preserve biological data quality.

Key Words:

Data Cleaning, Data Mining, Data Preparation, Data Validation, bioinformatics; biological data quality; data quality; data cleaning; Information quality;

I. INTRODUCTION

Data Mining aims at discovering knowledge out of huge data and presenting it in a form that is easily comprehensible to humans [1]. Data mining, sometimes called Knowledge Discovery in Databases (KDD), has been defined as "The nontrivial extraction of implicit, previously unknown, and potentially useful information from data" [2]. Data mining is a process that uses a variety of data analysis tools to discover patterns and relationships in data that may be used to make valid predictions [3], [5]. Other steps of the KDD process are the collection, selection, and transformation of the data and the visualization and evaluation of the extracted knowledge. Data mining employs algorithms and techniques from statistics, machine learning, artificial intelligence, databases and data warehousing and more. Some of the most popular tasks are classification, clustering, association and sequence analysis, and regression. Data mining is usually used by business intelligence organizations, and financial analysts, but is increasingly being used in the sciences and health management or medical diagnosis to extract information from the enormous data sets generated by modern experimental and observational methods [4].

The widespread exploitation of data mining is driven by technological advancements that generate voluminous data,

which can no longer be manually inspected and analyzed. For example, in the biological domain, the invention of highthroughput sequencing techniques enables the deciphering of genomes that accumulate massively into the biological databanks. Due to the large volume of biological data, i.e., databases such as GenBank [28] are often used with no consideration of the errors and defects contained within. When subject to automated data mining and analysis, these "dirty data" may produce highly misleading results, resulting in a "garbage-in garbage-out" situation. Further complication arises when some of the erroneous results are added back into the information systems, and therefore creating a chain of error proliferations. Data cleaning is particularly critical in databases with high evolutionary nature such as the biological databases and data warehouses. New data generated from the worldwide experimental labs are directly submitted into these databases on daily basis without adequate data cleaning steps and quality checks. Although data cleaning is the essential first step in the data mining process, it is often neglected conveniently because the solution towards attaining high quality data is non-obvious. Development of data cleaning techniques is at its infancy and the problem is complicated by the multiplicity as well as the complexity of data artifacts, also known as "dirty data" or data noise.

A. Data Cleaning

Data cleaning is an emerging domain that aims at improving data quality. It is a very large field that encompasses a number of research areas within database [6]. Data cleaning, also called data cleansing or scrubbing, deals with detecting and removing errors and inconsistencies from data in order to improve the quality of data. Data quality problems are present in single data collections, such as files and databases, e.g., due to misspellings during data entry, missing information or other invalid data. When multiple data sources need to be integrated, e.g., in data warehouses, federated database systems or global web-based information systems, the need for data cleaning increases significantly [7]. The process may include format checks, completeness checks, reasonableness checks, limit checks, review of the data to identify outliers (geographic, statistical, temporal or environmental) or other errors, and assessment of data by subject area experts (e.g. taxonomic specialists). These processes usually result in flagging, documenting and

subsequent checking and correction of suspect records. Validation checks may also involve checking for compliance against applicable standards, rules, and conventions [8]. As information is defined as data and method for its interpretation, it is only as good as the underlying data. Therefore, it is essential to maintain data quality. High quality data means that it is “fit for use” [14] and good enough to satisfy a particular business application. The following data quality measures allow one to quantify the degree to which the data is of high quality, namely:

- **Completeness:** All the required attributes for the data record are provided.
- **Validity:** All the record attributes have values from the predefined domain.
- **Consistency:** The record attributes do not contradict one another; e.g. the ZIP code attribute should be within the range of ZIP codes for a given city.
- **Timeliness:** The record should describe the most up-to-date state of the real-world object it refers to. Moreover, the information about an object should be updated as soon as the state of the real world objects changes.
- **Accuracy:** The record accurately describes the real world object it refers to; all the important features of the object should be precisely and correctly described with the attributes of the data record.
- **Relevancy:** The database should contain only the information about the object that is necessary for the purpose they were gathered for.
- **Accessibility and Interpretability:** The metadata describing the sources of the data in the database and transformations definitions it has undergone should be available immediately when it is needed.

In most cases it is almost impossible to have only “clean” and high-quality data entered into the information system. According to the research report of The Data Warehousing Institute (TDWI), “25% of critical data within Fortune 1000 companies will continue to be inaccurate through 2007 [13]. High quality data or “clean data” are essential to almost any information system that requires accurate analysis of large amount of real-world data. In these applications, automatic data corrections are achieved through data cleaning methods and frameworks, some forming the key components of the data integration process (e.g. data warehouses) [14] and are the pre-steps of even using the data (e.g. customer or patient matching) [15].

The classical application of data cleaning is in data warehouses [10, 11]. Data warehousing emerged as the solution for “warehousing of information” in the 1990s in the business domain; a business data warehouse is defined as a subject-oriented, integrated, non-volatile, time-variant collection of data organized to support management decisions. Data warehouses are generally used to provide analytical results from multidimensional data through effective summarization and processing of segments of source data relevant to the specific analyses. Business data warehouses are basis of decision support systems (DSS) that

provide analytical results to managers so that they can analyze a situation and make important business decisions. Cleanliness and integrity of the data contributes to the accuracy and correctness of these results and hence affects the impact of any decision or conclusion drawn, with direct cost amounting to 5 million dollars for a corporate with a customer base of a million [12].

II. THE PROBLEMS IN THE BIOLOGICAL DATABASES

Biological data is rich with issues that can be addressed with data cleaning and integration methodologies. Data cleaning in biological data is an important function necessary for the analysis of biological data. It can standardize the data for further computation and improve the quality of the data for searching. The very core purpose for most biological databases is to create repository, integrating work from numerous scientists [9]. While the problem of data artifacts in biological data has been known for a long time and individual artifacts have been reported [17, 18, 19, 20, 21, 22, 23, 24], the development of data cleaning approaches in the bioinformatics domain is at its infancy. Biological data management systems usually take the form of publicly accessible biological databases. They include primary sequence databases, protein structure databases, gene expression databases, micro-array databases, databases of protein-protein interactions, and a large number of specialist databases [27]. Many reasons accounts for the presence of data artifacts in biological databases. Biological database records are primarily collected through direct submissions by the worldwide experimentalists and sequence centers, bulk submissions from high-throughput sequencing projects, or data exchanges between the databases. Erroneous data may be mistakenly submitted, especially in projects that produce voluminous data. Different molecular databases have different data formats and schemas, and nomenclature is not standardized across databases. This introduces high level of information redundancy because the same sequence may have inconsistent, overlapping, or partial information in heterogeneous representations that cannot be easily merged. Some of the major databases update one another, replicating partial or full entries from one database to another. Replication of data also happens due to the annotation of same sequences by different groups, submission of the same sequence to different databases, or even re-submission of the same sequence to the same database either by same or different authors. In addition, the primary sequence records in the databases are often enriched with additional functional and structural information through manual annotations [26].

A Replication across Biological Databases

We carried out an analysis of scorpion toxins in SCORPION, a fully referenced database of 221 scorpion toxins [31] to assess the extent of redundancy in biological data. The SCORPION records compiled from public database sources GenBank/GenPept, Swiss-Prot, EMBL,

DDBJ, TrEMBL, PIR and PDB were overlapping to various degrees (Table 1). From among the raw entries, we found 143 cases of replication across two or more databases. Nearly half of the data entries were incomplete and required enrichment with additional structural and functional annotations.

Scorpion toxin entries

Databases	No. of toxins
Genbank, Swiss-prot, EMBL, DDBJ, PDB	3
Genbank, Swiss-prot, EMBL, DDBJ, PIR	10
Genbank, Swiss-prot, EMBL, DDBJ	19
Genbank, Swiss-prot, PIR, PDB	10
Genbank, EMBL, DDBJ, TrEMBL	17
Genbank, Swiss-prot, PIR	36
Genbank, Swiss-prot, PDB	5
Genbank, EMBL, DDBJ	16
Genbank, Swiss-prot	9
Genbank, PIR	6
Genbank, PDB	2
Genbank, TrEMBL	2
Genbank, PDB	8
Total	143

Table 1: Scorpion toxin entries replicated across multiple databases.

The bioinformatics data is characterized by enormous diversity matched by high redundancy, across both individual and multiple database s. Enabling interoperability of the data from different sources requires resolution of data disparity and transformation in the common form(data integration), and the removal of redundant data, errors, and discrepancies (data cleaning). Frequently encountered data redundancy issues are:

1. fragments and partial entries of the same item(e.g. sequence) may be stored in several source record ;
2. Databases update and cross-reference one another with a negative side effect of occasionally creating duplicates, redundant entries and , proliferating errors;
3. The same sequence may be submitted to more than one database with out cross-referencing those records;
4. The “owners” of the sequence record may submit a sequence more than once to the same database. To enable the extraction of knowledge in a data warehousing environment, these are rectified by dataware house integration and data cleaning components.

B. Data cleaning Tools

Very few complete data cleaning methods for biological data exist. The BIO-AJAX tool for detecting and resolving duplicate taxonomy of organisms utilize prefix-matching strategies to integrate different terms that describe the same species [26]. A case study of handling noises in Osteogenesis Imperfecta (OI) related sequences is presented in [25]. A method for addressing the genomic nomenclature

problem by using phylogenetic tools along with the BIO-AJAX data cleaning framework is proposed in [29]. A framework for the application of data mining tools to data cleaning in the biological domain has been presented [30]. They focused on tuple constraints and functional dependencies detection in representative biological databases by means of association rule mining. By analyzing association rules they can deduce not only constraints and dependencies, which provide structural knowledge on a dataset and may be useful to perform query optimization or dimensional reduction, but also the anomalies in the system, which could be errors or interesting information to highlight to domain experts. Bioinformatics has the same demand for high quality data, but there are limited data cleaning applications available in the domain. Majority of data cleaning methods focus on the more challenging duplicate and outlier detection problems, while other approaches address database repair issues related to various types of violations, inconsistency, and errors.

III. CORRELATION ANALYSIS

Redundancies refer to data which are recorded in more than one database entries due to different data sources, varying views of the proteins (PDB protein structures versus Swiss-Prot protein annotations), or repeated submissions of the sequence by the same or different annotators. We propose a correlation-analysis method for detecting duplicates of the biological databases. An attribute may be redundant if it can be derived from another table. Inconsistencies in attribute or dimension naming can also cause redundancies in the resulting data set. Redundancies can be detected by correlation analysis.

For given two attributes, such analysis can measure how strongly one attribute implies the other, based on the available data. For the numerical attributes, we can evaluate the correlation between attributes A and B , by computing the correlation coefficient. This is Where N is the number of tuples, a_i and b_i are the respective values of A and B in tuple i , \bar{A} and \bar{B} are the respective mean values of A and B. Where n is the number of tuples, \bar{A} and \bar{B} are respective mean values of A and B, and σ_A and σ_B Are the respective standard deviations of A and B. If the resulting value of equation (1) is greater than 0, than A and B are positively correlated σ_A and σ_B are the respective standard deviation of A and B, and $\Sigma (AB)$ is the sum of the AB crossproduct. Note that $-1 \leq r_{A,B} \leq +1$. If $r_{A,B}$ is greater than 0, then A and B are positively correlated, meaning that the values of A increase as the values of B increase. The higher the value, the stronger the correlation (i.e., the more each attribute implies the other attribute decrease. Note that correlation does not imply causality. That is, if A and B are correlated, this does not necessarily imply that A causes B or that B causes A. For example, in analyzing a demographic database, we may find that attributes representing the number of hospitals and the number of car thefts in a region are correlated. This does not mean that one causes the other. Both are actually causally

linked to a third attribute, namely, population. For categorical (discrete) data, a correlation relationship between two attribute, A and B, can be discovered by χ^2 (chi-square) test. Suppose A has c distinct values, a_1, a_2, \dots, a_c . B has r distinct values, namely b_1, b_2, \dots, b_r . The data tuples described A and B can be shown as a contingency table, with the c values of making up the column and the r values of B making up the rows, Let (A_i, B_j) denote the event that attribute A takes on value b_j , that is, where $(A=a_i, B=b_j)$. Each and every possible (A_i, B_j) joint event has its own cell in the table. The χ^2 value is computed as:

$$\chi^2 = \sum_{i=1}^c \sum_{j=1}^r \frac{(o_{ij} - e_{ij})^2}{e_{ij}} \quad (1)$$

Where o_{ij} is the observed frequency of the joint event (A_i, B_j) and e_{ij} is the expected frequency of (A_i, B_j) , which can be computed as

$$e_{ij} = \frac{\text{count}(A = a_i) \times \text{count}(B = b_j)}{N}$$

Where N is the number of data tuples, $\text{count}(A = a_i)$ is the number of tuples having value a_i for A and $\text{count}(B = b_j)$ is the number of tuples having value b_j for B. The sum in Equation (1) is computed over all of the $r \times c$ cells. Note that the cells that contribute the most to the χ^2 value are those whose actual count is very different from that expected.

IV. MATERIALS AND METHODS

This section details the redundancy detection framework.

A. Redundancy detection framework

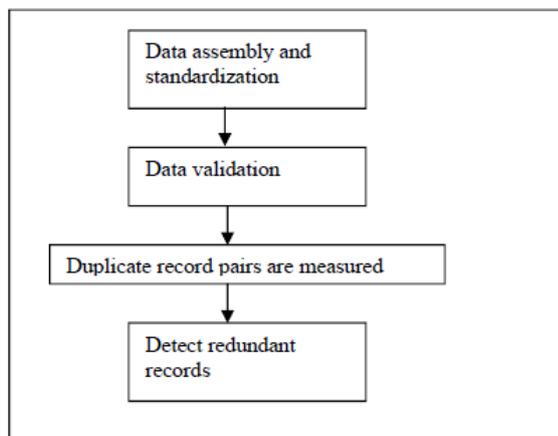


Fig 1: Redundancy detection framework

Figure.1 depicts the redundancy detection framework. First, all the data are collected and we should replace synonyms with one standard term (e.g. Hiway 9 -> Highway 9). Second stage is data validation which specifies acceptance criteria. Based on these acceptance criteria of each duplicate record pairs are measured using varying similarity functions, depending on the data types of the attributes. In the final stage duplicate records are detected.

V. RESULTS AND DISCUSSION

The dataset is a combination of two set of records. The first data set consists of 520 scorpion toxin proteins retrieved from Entrez using the keywords "scorpion AND venom". The second set contains 780 snake PLA2 venom proteins retrieved from Entrez using the keywords "serpentes AND venom AND PLA2". The 700 records were annotated separately; 695 duplicate pairs were identified collectively.

Scorpion toxin = 251
Snake PLA2 toxin = 444
Total Duplicate = 695

Experiments were performed on a Pentium-IV computer with 1GB of main memory, and running Windows XP. The information overload era result in a manifestation of low quality data in real-world databases. The demand for high quality data surges and opens new challenges for data cleaning. This paper aims at handling the data quality problem through correlation analysis.

VI. CONCLUSION

With rapid growth of public biological data and fast development of computational methods based on mining of these data, achieving high quality datasets is becoming increasingly important for effective data mining. In this paper, we discussed the biological database problems. Also we presented a novel method for data cleaning, specifically in redundancy detection, using correlation analysis.

ACKNOWLEDGMENT

We would like to thank our family members for their constant help, support and encouragement throughout our research work.

REFERENCES

- [1] Nevine M. Labib, and Michael N. Malek D, "Data Mining for Cancer Management in Egypt Case Study: Childhood Acute Lymphoblastic Leukemia", Proceedings Of World Academy Of Science, Engineering And Technology, Volume 8, October 2005.
- [2] Frawley, W., Piatetsky-Shapiro, G., Matheus, C., "Knowledge Discovery in Databases: An Overview", AI Magazine, fall 1992, pp. 213-228, 1992.
- [3] Two Crows Corporation, "Introduction to data mining and knowledge discovery", Published by Two Crows Corporation, 36 pages, 1999, ISBN 1892095025, 9781892095022.
- [4] David L. Iverson, "Data Mining Applications for Space Mission Operations System Health Monitoring", NASA Ames Research Center, Moffett Field, California, 94035, 2008.

- [5] Ralf Rantzau and Holger Schwarz, "A Multi-Tier Architecture for High- Performance Data Mining", University of Stuttgart, Institute of Parallel and Distributed High-Performance Systems (IPVR), Breitwiesenstr, D-70565, Stuttgart, Germany, pp. 20-22, 1999.
- [6] Dasu, T. and Johnson, T., "Exploratory Data Mining and Data Cleaning", Journal of Statistical Software, Volume 11, Book Review 9, September 2004.
- [7] Rahm, E. and Do, H.H., "Data cleaning: problems and current approaches", Bulletin of the Technical Committee on Data Engineering, Special Issue on Data Cleaning, Vol. 23, No. 4, pp.3-13, December 2000.
- [8] Chapman, A. D. "Principles and Methods of Data Cleaning – Primary Species and Species-Occurrence Data", version 1.0. Report for the Global Biodiversity Information Facility, Copenhagen, 2005.
- [9] Katherine G. Herbert, Jason T.L. Wang, "Biological data cleaning: a case study", Int. J.Information Quality, Vol. 1, No. 1, 2007.
- [10] M. L. Lee, H. Lu, T. W. Ling, and Y. T. Ko, "Cleansing Data for Mining and Warehousing", DEXA, 751-760, 1999
- [11] V. Raman and J. M. Hellerstein, "Potter's wheel: an interactive data cleaning system", VLDB, pages 381-390, 2001.
- [12] P. Vassiliadis, Z. Vagena, S. Skiadopoulou, N. Karayannidis and T. Sellis, "ARKTOS: A tool for data cleaning and transformation in data warehouse Environments", IEEE Data Engineering Bulletin, 23(4):42-47, 2000.
- [13] B. Beal "Bad Data Haunts the Enterprise" in Search CRM, http://searchcrm.techtarget.com/news/article/0,289142/sid11_gci965128_00.html
- [14] R. Kimball, M. Ross, "The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling", Wiley, John & Sons, Incorporated, 464pp, ISBN-13: 9780471200246, 2002
- [15] M. Lee, T. Ling, W. Low, "IntelliClean: A knowledge-based intelligent data cleaner" in Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining, pp.290-294, 2000.
- [16] Lukasz Ciszak, "Application of Clustering and Association Methods in Data Cleaning", Proceedings of the International Multiconference on Computer Science and Information Technology, pp. 97 – 103, 2008
- [17] P. Bork and R. Copley, "The draft sequences: Filling in the gaps", Nature, vol. 409(6822), pp. 818-820, 2001.
- [18] S. E. Brenner, "Errors in genome annotation", Trends in Genomics (TIG), 15:132-133, 1999.
- [19] R. Guigo, P. Agarwal, J. F. Abril, M. Burset, and J. W. Fickett, "An assessment of gene prediction accuracy in large DNA sequences", Genome Research, vol. 10, pp. 1631-1642, 2000.
- [20] W.R. Gilks, B. Audit, D. De-Angelis, S. Tsoka, and C. A. Ouzounis, "Modeling the Percolation of annotation errors in a database of protein sequences", Bioinformatics, 18(12):1641-1649, 2002.
- [21] H. Müller, F. Naumann, and J. Freytag, "Data Quality in Genome Databases", International Conference on Information Quality, pages 269-284, 2003.
- [22] I. Iliopoulos, S. Tsoka, M. A. Andrade, A. J. Enright, M. Carroll, P. Poulet, V. Promponas, T. Liakopoulos, G. Palaios, C. Pasquier, S. Hamodrakas, J. Tamames, A. T. Yagnik, A. Tramontano, D. Devos, C. Blaschke, A. Valencia, D. Brett, D. Martin D, C. Leroy, L. Rigoutsos, C. Sander, and C. A. Ouzounis, "Evaluation of annotation strategies using an entire genome sequence", Bioinformatics, 19(6):717-726, 2003.
- [23] R. Sorek and H. M. Safer, "A novel algorithm for computational identification of contaminated EST libraries", Nucleic Acids Research, 31(3):1067-1074, 2003.
- [24] H. Pospisil, A. Herrmann, R. H. Bortfeldt, and J. G. Reich, "EASED: Extended Alternatively Spliced EST Database", Nucleic Acids Research, 32(Database issue):70-74, 2004.
- [25] C. M. Teng, "Applying noise handling techniques to genomic data: A case study", IEEE ICDM, pages 743- 746, 2003.
- [26] K. G. Herbert, N. H. Gehani, W. H. Piel, J. T. L. Wang, and C. H. Wu, "BIOAJAX: An extensible framework for biological data cleaning", Sigmod Record, vol. 33, no. 2, pp. 51-57, 2004.
- [27] Altman RB, "Building successful biological databases", Brief. Bioinformatics 5 (1): 4-5, March 2004. PMID 15153301.
- [28] D. A. Benson, I. Karsch-Mizrachi, D. J. Lipman, J. Ostell, and D. L. Wheeler. GenBank. Nucleic Acids Research, 34(Database issue):16-20, 2006.
- [29] Jonathan D. Marra, Katherine G. Herbert, Jason T. L. Wang, "A study of phylogenetic tools for genomic nomenclature data cleaning", Proceedings of the 12th annual SIGCSE conference on Innovation and technology in computer science education, pp. 347, 2007.
- [30] Apiletti, Daniele; Bruno, Giulia; Ficarra, Elisa; Baralis, Elena, "Data Cleaning and Semantic Improvement in Biological Databases", Journal of Integrative Bioinformatics - JIB (ISSN 1613-4516), vol. 3, no. 2, 2006.
- [31] Srinivasan, K.N., Gopalakrishnakone, P., Tan, P.T., Chew, K.C., Cheng, B., Kini, R.M., Koh, J.L., Seah, S.H., Brusica, V. SCORPION, a molecular database of scorpion toxins. *Toxicon*, 40, 23-31, 2002.

Semantic Search and Retrieval of Stock Photography based on MPEG-7 Descriptors

Balasubramani R
Assistant Professor-IT
Sikkim Manipal University- DDE
1 Floor, Syndicate House
Manipal- 576104
Karnataka, India
E-mail: microtech_balu@yahoo.com

Dr.V.Kannan
Dean
Center for Information
Bharath University
Chennai- 600073
Tamil Nadu, India
Email- drvkannan62@yahoo.com

Abstract- With the growing amount of people using the Internet, and creating digital content and information, knowledge retrieval becomes a critical task. Ongoing efforts provide standards and framework for annotating digital and non-digital content semantically to describe resources more precisely and processable in comparison to simple descriptive structured and unstructured metadata. Although the MPEG group provides with MPEG-7, a useful and well-defined theoretical framework for the creation of semantic annotation, retrieval of annotations is not discussed. In this paper we present a retrieval process for MPEG-7 based semantic annotations founded on well proved information retrieval techniques, namely query expansion and regular expressions. Additionally NWCIBIR, a prototype implementation for semantic search and retrieval will be presented.

Keywords:
MPEG-7, NWCIBIR, Semantic Search and Retrieval.

I. INTRODUCTION

In traditional libraries metadata plays a central role, as keywords and taxonomies provide short and meaningful descriptions and cataloguing. It provided for a long time the only alternative way to inspecting all available books of finding what users need within the inventory of a traditional library. In digital libraries this context was "digitized" but remained quite similar to the original concept. Current trends show that the efforts and achievements of the information retrieval research area are integrated to enhance digital libraries ([Lossau2004], [Summan2004]). On the other hand much of the metadata based methods of digital libraries have been adopted in knowledge management, knowledge discovery and information retrieval (i) for providing an application area for techniques like metadata extraction and automatic taxonomy creation and (ii) for enhancing knowledge management, discovery and retrieval by using metadata based retrieval techniques. Especially in the latter field techniques based on query expansion using thesauri or ontologies are very successful. Multimedia

retrieval heavily depends on such techniques and the appropriate metadata. Content based image and video retrieval requires the pre-processing and indexing of content before query time, this pre-processing is the extraction of low level metadata. An often discussed topic in content based image retrieval is the semantic gap ([DelBimbo1999], [Smeulders2000]), which defines the difference between automatically extracted image features and the understanding or description of visual information of a user. If the semantic gap can be bridged by retrieval mechanisms no annotation would be necessary.

Right now semantic descriptions have to be created, at least in parts, manually. Human Computer Interaction (HCI) methods and information retrieval methods exist, that support the user in the annotation task. Different formats and approaches for the storage and definition of semantic descriptions are currently discussed and in use, wherefrom MPEG-7 is one of them.

II. MPEG-7 BASED SEMANTIC DESCRIPTIONS

The standard being used to define the way of handling the metadata has to be a lot more powerful than *EXIF* or for instance Dublin Core [Hunter2000]. *DC* only defines 15 core qualifiers, which can be understood as metadata tags, which can be filled by the user. A combination of Dublin Core and adapted Resource Description Framework structures, *RDF*, would at least permit a structured storage of graphs and a quality rating, although content based image retrieval would not be supported. An import of the *EXIF* information to a *RDF*-based structure is possible. The main proposition against *RDF* is that there exists, at this time, no standardized structure for saving all or most of the metadata defined in the requirements above. Although it would not prove impossible to create such a structure, to gain interoperability with other systems and implementations, agreeing on the same *RDF* based enhancements with all other developers or vendors is necessary. Based on these facts a much better choice is MPEG-7 [Benitez2002].

III. REALIZATION OF NWCIBIR RETRIEVAL TOOL

NWCIBIR gives the user the ability to retrieve annotated photos. Due to the fact, that this is experimental software the retrieval mechanism is file system based. All MPEG-7 documents found by NWCIBIR in a specified directory and in further sub-directories are searched. Figure 1 shows the simplified UML diagram of the NWCIBIR System's Retrieval Tool.

NWCIBIR offers three different ways to search for a matching photo:

1. Defining search options through textboxes with various options.
2. Content based image retrieval using the visual descriptors *ColorLayout*, *ScalableColor* and *EdgeHistogram* defined in the MPEG-7 standard.
3. Searching for a similar semantic description graph.

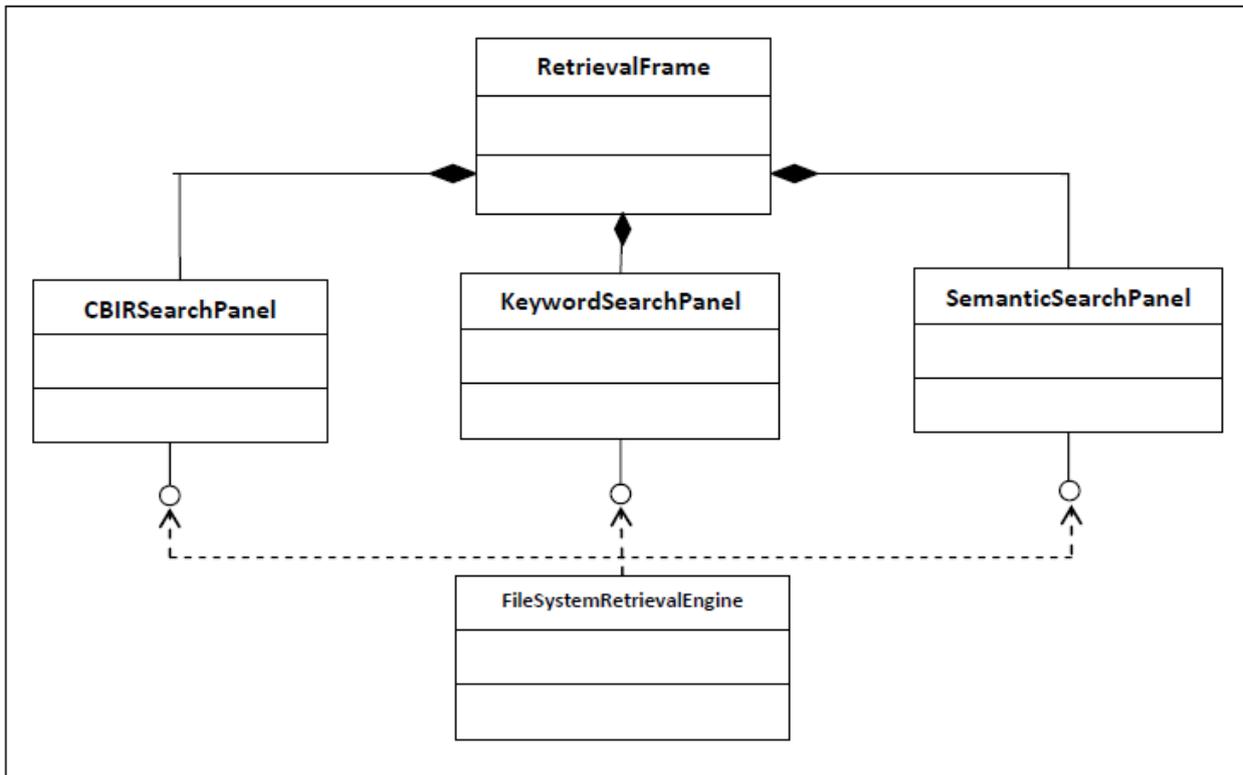


Fig. 1: Simplified UML diagram of the NWCIBIR System's Retrieval Tool

IV. SEMANTIC SEARCH

The input for the retrieval process is a semantic description, given by the user. The output lists all relevant semantic descriptions in the database sorted by their relevance compared to the query. To achieve these goals a mathematical and data model for the semantic descriptions has been built and a retrieval strategy has been created.

V. THE MODEL OF THE MPEG-7 SEMANTIC DESCRIPTION SCHEME

All semantic descriptions consist of nodes, which are semantic descriptors extended from the semantic base descriptor, and relations, which interconnect two different nodes. The MPEG-7 Semantic *DS* can be seen as directed

graph; whereas the nodes are the vertices and the relations are the directed edges. The graph is not necessarily connected, as relations are not mandatory. As all the nodes and relations are identified by descriptors, a semantic description is a labelled graph, whereas the MPEG-7 descriptors are the labels for the edges and vertices. Screenshots of visual representations are given in figure 2. For the sake of simplicity two nodes cannot be connected through two different relations and a relation cannot have one single node as start and end node. In graphs based on Semantic *DS* no two nodes can have the same label (the same descriptor), so the node labels are unique. Each directed edge can be inverted as there exists an inverse of each MPEG-7 based semantic relation.

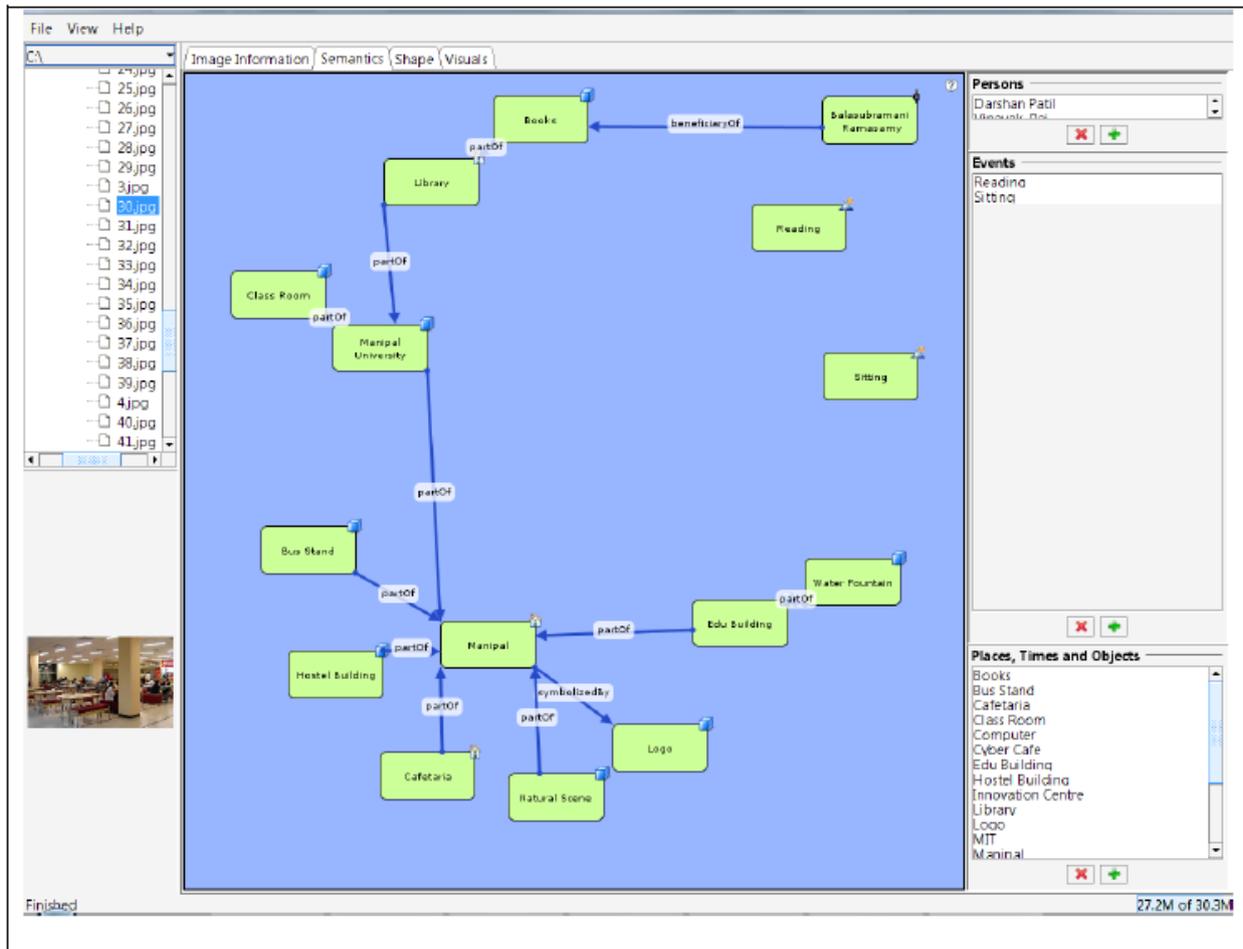


Fig. 2: Example of the representation of a MPEG-7 based semantic description.

VI. THE TERM SPACE FOR THE MPEG-7 SEMANTIC DS

Based on the identified constraints of a directed labelled graph with unique node labels and edges, that can be inverted, an efficient retrieval strategy can be designed as follows: The idea of fast retrieval of graph based structures is not new, as the contribution of Simmons in 1996 [Simmons1966] shows. Common retrieval techniques for graphs are the usage of metrics like the maximum common sub-graph metric, or the graph edit distance. A straight forward implementation using this distance measures results in search time $O(n)$, whereas n defines the number of graphs in the database. Please note that the distance or similarity calculation between two graphs is NP -hard in respect to the number of nodes and edges of the graphs to compare [Valiente2002]. Another approach is the filtering of the database with a fast (less than linear search time) algorithm and the ranking of the results with a slower metric which is described in chapter 12 in [Baeza-Yates1999]. This method has been successfully used for graphs e.g. in [Fonseca2004] for clipart retrieval by using graph eigenvalues as filters like in [Shokoufandeh1999]. A more promising approach for

MPEG-7 semantic *DS*, if the usage of an existing text search engine is constraint, is the usage of a path index [Shasha2002]. A path index allows the fast retrieval of graphs based on paths (sequences of nodes connected by edges, whereas the number of edges defines the length of the path) of different lengths extracted from the graphs. The extracted paths can be interpreted as index terms for a graph.

The graph can be expressed using all paths, which are sequences of nodes interconnected by edges, of chosen length. Paths of length 0 are the nodes themselves while paths of length 1 are triples as used in *RDF*. Paths of length 0 and length 1 have unique string representations for MPEG-7 based semantic descriptions as shown in [Lux2005]. To allow the usage of wildcard nodes at least the paths of length 2 have to be used, for which a unique string representation can be defined as shown below. The graph can be stored using the paths of length 0, 1 and 2 as index terms. Using a query graph all paths of the query graph are extracted and used as search terms. The ranking is done by $TF*IDF$ on the index terms, which are the paths of the graphs.

VII. IMPLEMENTATION

Based on an open source retrieval engine (*Lucene*), an index for node descriptors has been implemented in NWCBIR along with the string representations of paths of length 0 and 1. As the query graph can consist of query strings for the node values, query expansion based on the node descriptors is used as described in [Lux2005]. All path representations are constructed from node IDs, which identify a unique node descriptor in the index, and relation names or wildcards for nodes or relations. For the usage for terms within the retrieval engine (*Lucene*), the path representations were adopted: all white spaces were replaced by „_“ and all paths start with a leading „_“. The leading „_“ allows the usage of wildcards at the start of a path expression.

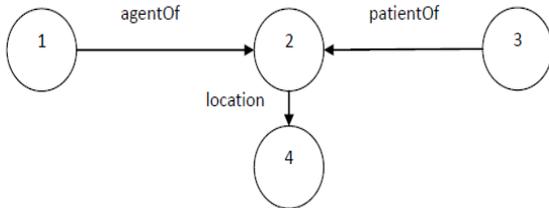


Fig. 3: Example for a graph following the model of MPEG-7 semantic DS graph.

For the graph given in figure 3 the terms for paths of length 0 and 1 are given in the following table.

Table 1: Extracted path terms of length 0 and 1 from graph shown in figure 3.

Term	Path length
_1	0
_2	0
_3	0
_4	0
_agentOf_1_2	1
_locationOf_4_2*	1
_patientOf_3_2	1

(*Note that the path `_locationOf_4_2` has been inverted. This is done to normalize the edge directions in the index.)

For the creation of terms from paths of length 2, following method has been introduced. The input of the method is either a graph representing a semantic *DS* or a query graph. In a first step all paths of length 2 are extracted from the graph [Valiente2002]. For each of these extracted paths the unique string representation has to be created as follows:

1. Compare the start node of the path with the end node of the path.
 2. If the start node is bigger than the end node reverse the path:
 - a. Switch end and start node.
 - b. Switch and invert first and second relation.
3. Create string in order: start node – first relation – middle node – second relation – end node with „_“ as separator.
4. Prepend „_“ to the string.

This results in the table 2 as shown below.

Table 2: This table shows all available extracted path terms of length 2 from the graph shown in figure 3.

Term	Path length
_1_agentOf_2_patient_3	2
_1_agentOf_2_location_4	2
_3_patientOf_2_location_4	2

All these above shown terms are used to index the semantic description with *Lucene*; all terms are used as *Lucene* tokens without stemming or other pre-processing. For queries the terms are constructed in a similar manner with one exception: Wildcards for nodes and relations can be used. For relations the adoption is straightforward: As *Lucene* supports wildcard queries for a wildcard relation the string „*“ is inserted instead of the relation name, e.g. `_*_I_2` instead of `_agentOf_I_2`. To support undirected wildcard relations two relation query terms are constructed and combined with a Boolean OR, like `(_*_I_2 OR *_2_I)`. For paths of length 2 only the „*“ is inserted instead of the relation name as the order of the path only depends on the start and end node.

For nodes in paths of length 0 the query string is omitted. For paths of length 1 and middle nodes in paths of length 2 the node ID is replaced with a „*“. For start and end nodes in paths of length 2 a Boolean query clause has to be constructed as the order of the start and end node cannot be used to identify the term representation, e.g. `(_*_patientOf_2_location_4 OR 4_locationOf_2_patient_*)`. Note that the relations have to be inverted in this case.

A simple example for a wildcard query would be: “Find all semantic descriptions where Balasubramani is doing something at the SMU”. In the first step possible candidates for nodes are identified to construct the query graphs. Assuming that for Balasubramani the node with ID 28 has been found, while for SMU the node with ID 93 has been found, the query graph would look like “[28] [93] [*] [agentOf 1 3] [locationOf 3 2]”. The numbers within the relations reference the node using their position in the node list. Such a query would result in a query like “_28_93_agentOf_28_*_locationOf_*_93_28_agentOf_*_locationOf_93”

VIII. RETRIEVAL MECHANISM FOR SEMANTIC DESCRIPTIONS

This section introduces our retrieval model, which is motivated by providing a fuzzy retrieval mechanism for semantic descriptions and an appropriate ranking scheme, including support for wildcards. As there are already well functioning and well tested tools for text retrieval available one major constraint is that we want to focus on the usage of existing text retrieval tools. All of the used techniques should find their source in existing text retrieval techniques to allow the usage of existing tools if possible to rely on their speed and precision.

For the retrieval process of MPEG-7 based semantic descriptions we can assume that, without loss of generality, a set of image exists, where each image is annotated with a semantic description. Thus, our goal is to retrieve a set of semantic graphs best matching a given input graph. In the following section nodes (or vertices) of the graph are denoted as semantic objects and edges of the graph are denoted as semantic relations. Our model is described in three parts, whereas the first part explains the indexing of semantic objects and semantic descriptions, the second part states on the retrieval process and the third part introduces the ranking method.

IX. INDEXING OF SEMANTIC DESCRIPTIONS

The first step is creating a data structure for accessing nodes N of the graph. As in text retrieval we are using an inverted index as data structure, which holds all semantic objects and offers good possibilities for speeding up the retrieval process as a whole.

In general for every unique semantic object in all semantic descriptions a unique identifier is assigned and an index entry is created, which means the text describing a semantic object is indexed using text retrieval methods. Note that, multiple semantic descriptions can share the same semantic objects. In this case each shared semantic object is treated as one object and obtains the same unique ID within the different semantic descriptions. Figure 4 shows in detail the implementation of the indexing process.

For example if the description shown in figure 5 are processed three different semantic objects with three different IDs are extracted: (i) Balasubramani (Semantic agent, part of description of image A and B, ID: 1), (ii) Talking (Semantic event, part of description of image A, ID: 2) and (iii) Listening (Semantic event, part of description of image B, ID: 3).

For example if the description shown in figure 5 are processed three different semantic objects with three different IDs are extracted: (i) Balasubramani (Semantic agent, part of description of image A and B, ID: 1), (ii) Talking (Semantic event, part of description of image A, ID: 2) and (iii) Listening (Semantic event, part of description of image B, ID: 3).

After indexing and assigning unique IDs to semantic objects, semantic descriptions are indexed using a string representation. Given the semantic descriptions in figure 5, the string representation of image A is: [1] [2] [agentOf 1 2], for image B the description is [1] [3] [agentOf 1 3]. In the first part of the string representation all available semantic objects (vertices) of the semantic description (graph) represented by their ID in square brackets are defined in numerically ascending order. The second part consists of all available semantic relations (edges) in lexicographically ascending order. Each semantic relation is defined in square brackets, whereas the name of the relation is followed by the ID of its source and the ID of its target. Note that the number of possible MPEG-7 based semantic relation types is already limited by the standard itself. Therefore relations do not get unique IDs but are referenced by their names. All possible semantic relations in MPEG-7 are directed edges but have an inverse relation. Based on this fact relations are re-inverted if their inverse relation is used in a graph. It can be seen that the string representation is unique.

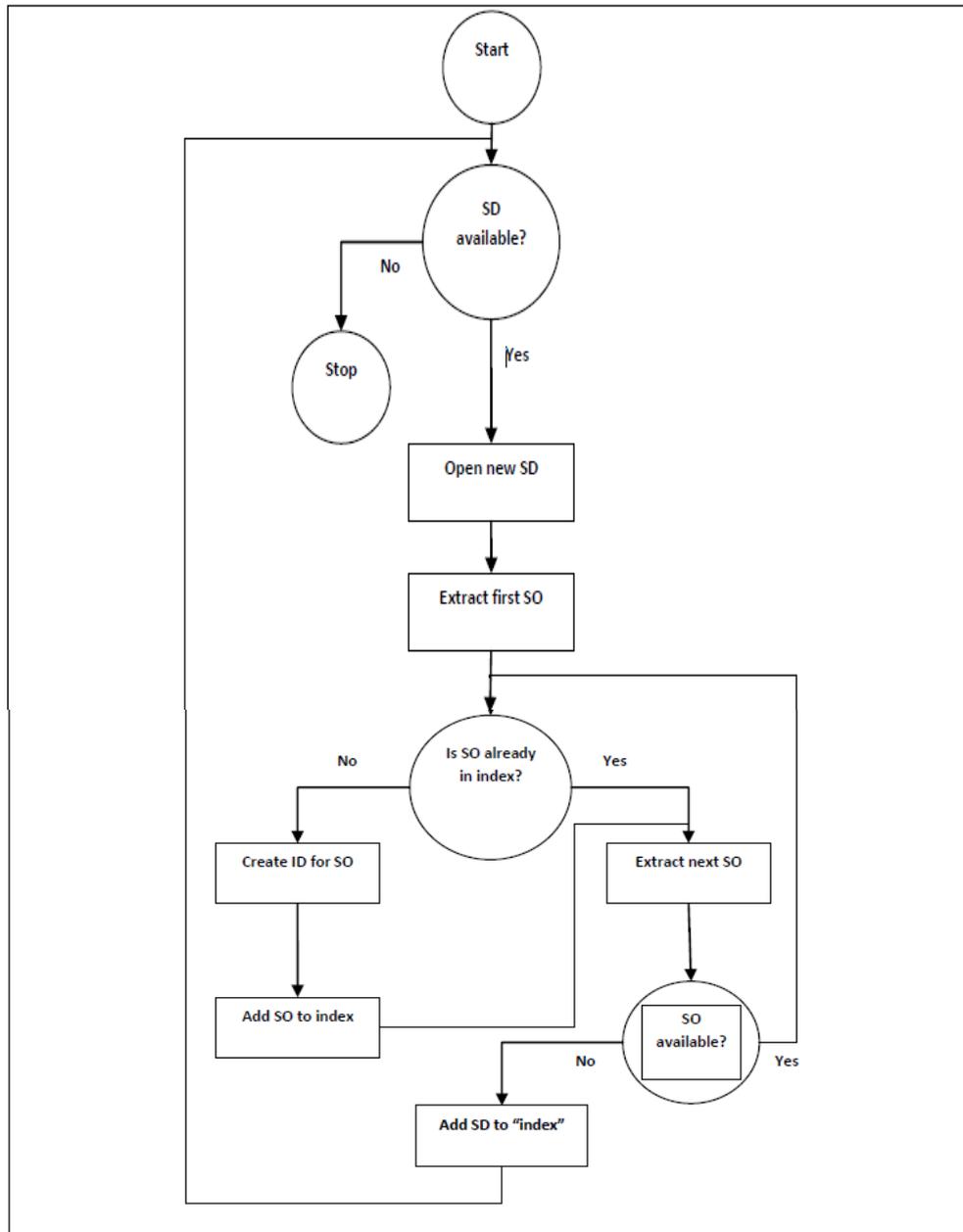


Fig. 4: Flow diagram showing the process of “indexing” semantic descriptions (SD) and semantic objects (SO).

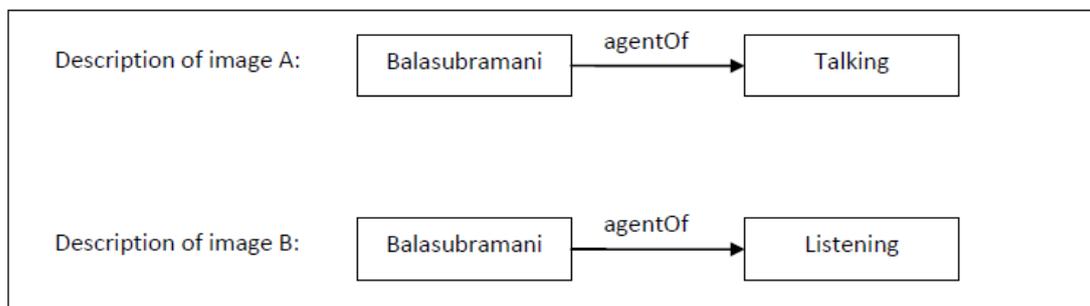


Fig. 5: Examples for semantic descriptions of two different images

X. RETRIEVAL OF SEMANTIC DESCRIPTIONS

Given the fact that above described indices and representations exists, a retrieval mechanism can be implemented as follows:

1. A user provides a query string for each of the k semantic objects he wants to search for and interconnects the nodes with relations. Each query string leads to a node query, $q1$ to qk , which are used to query the inverted index of semantic objects, as described earlier.

2. The retrieval engine searches for a set of available matching node IDs $Lq1$ to Lqk for each node query $q1$ to qk , sorted by relevance of the matches. The relevance returned for each relevant node is in $(0, 1]$, whereas a relevance of 1 indicates an optimal match. The relevance is obtained from using standard text relevance methods (e.g. vector space model).

3. Based on the sets of matching nodes for each node query the original query is expanded to $|Lq1|*|Lq2|*...*|Lqk|$ queries, for which the node IDs and the relevance of the nodes are available. This means that every node returned by qi is combined with every node returned by qj having $i \neq j$. Given the semantic relations of the user queries consisting of semantic descriptions can be created.

4. For each of the above $|Lq1|*|Lq2|*...*|Lqk|$ queries the number of matching documents is found through a search in the string representations of the graphs with regular expressions. A relevance value for each matching document is calculated based on the formula presented in the next section.

5. All resulting sets from step 4 are merged in one result set, whereas for documents which are in more than one set, a higher relevance value is assigned.

XI. RELEVANCE CALCULATION

Taking one specific expanded query q with node set $Nq = \{ , \dots, \} \neq \emptyset$ and relation set $Rq = \{ , \dots, \}$ and one specific matching semantic description d resulting from the search in step 4 with node set $Nd = \{ , \dots, \}$ and relation set $Rd = \{ , \dots, \}$ with $k, l, r, s \in N U \{0\}$. The relevance $r \in (0, 1]$ based on the query nodes relevance values

$r(n_1^q), r(n_2^q), \dots, r(n_k^q) \in (0, 1]$ is defined by:

$$r = \frac{\min(|N^q| + |R^q|, |N^d| + |R^d|)}{\max(|N^q| + |R^q|, |N^d| + |R^d|)} \cdot \prod_{i=1}^{|N^q|} r(n_i^q) \quad (1)$$

The calculated relevance takes the relevance of nodes, which result from the query expansion, into account. The relevance value is in the interval $(0, 1]$ because all node

relevance values are in $(0, 1]$ and the fraction has to be in $(0, 1]$ because the numerator is smaller or of equal size compared to the denominator. Note that all irrelevant nodes are discarded in step 2 by discarding all nodes with relevance below a specific threshold which leads to a minimum relevance above zero. The relevance of semantic relations is not taken into account as the relations in the query are only matched with relations in the database following the Boolean model, not supporting a partial or fuzzy match.

To express the meaning of the relevance formula in words: The more relevant is the matching semantic description. Additionally the smaller the difference in the number of components (nodes and edges) of the query and description graph is, the more relevant is the matching semantic description.

XII. IMPLEMENTATION DETAILS

The above described method was implemented in NWC BIR. NWC BIR uses the *Jakarta Lucene* search engine [*Lucene*], which allows the creation of an inverted index of nodes. The string representations of semantic descriptions are stored in a flat file.

For query formulation a simple query language is used which can be described as follows: All nodes are defined in square brackets, inside these square brackets all features of *Lucene* like fuzzy matching or field matching can be used. Following these node queries the relations are defined using the name of the relation as defined in the MPEG-7 standard followed by the source of the relation and the target of the relation identified by the position in the list of node queries. Following *BNF* expression defines the supported queries:

```
Query ::= NodeQuery {NodeQuery}
{RelationQuery}
NodeQuery ::= "[" NodeQueryString "]"
NodeQueryString ::= ( Clause ) *
Clause ::= ["+", "-"] [<Term> ":" ] (
<Term> | "(" NodeQueryString ")" )
RelationQuery ::= <MPEG-7_Relation>
<Number> <Number>
```

From each of the expanded queries a regular expression for searching in the file of semantic descriptions is created and executed on each semantic description in its string representation. If the regular expression matches the string, the associated documents are put into the result set and the relevance of the documents is calculated. Finally the result sets are merged following above described parameters and the sorted set of results is presented to the user.

XIII. USER INTERFACE

The component of most interest is the panel offering a search mechanism for searching semantic descriptions.

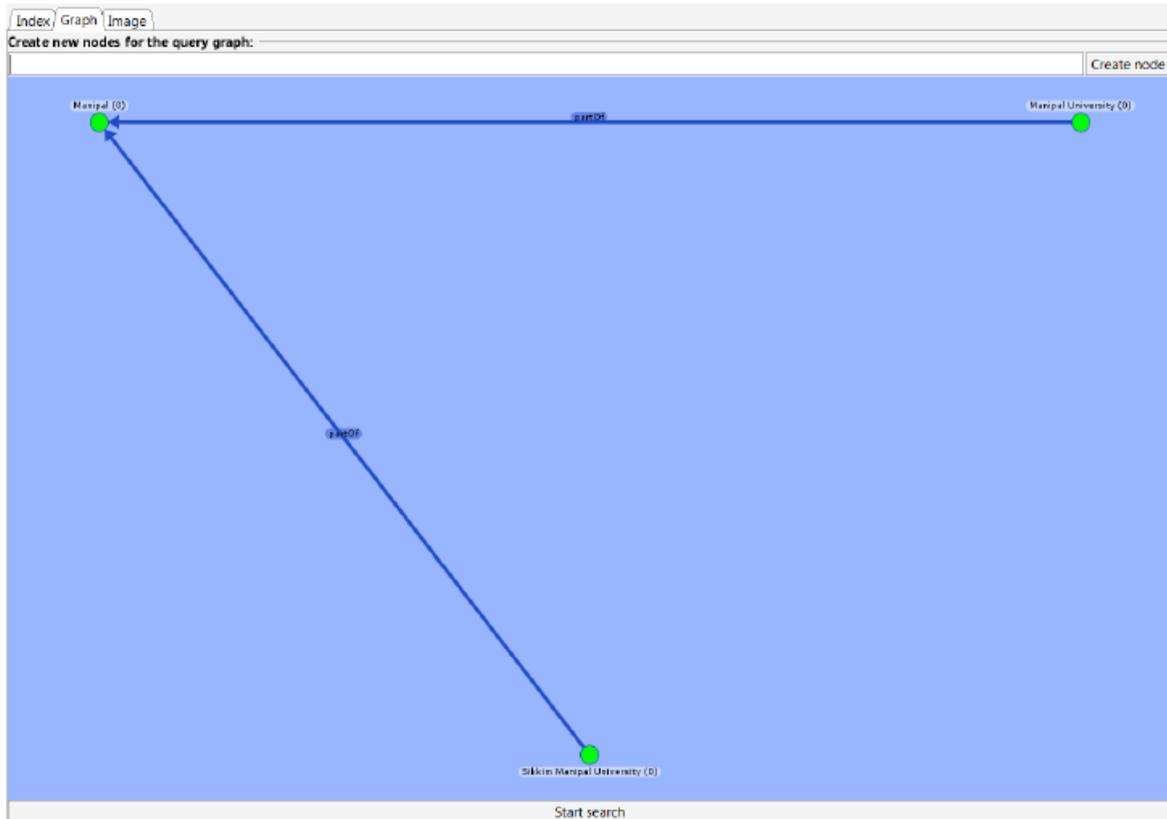


Fig. 6: Starting a semantic search using a graph as input. Three objects and two relations are defined.

This component allows the user to define a graph with minimum one to maximum three nodes and two possible relations. An asterisk is used as wildcard. A search graph which only contains one node with a word defining this node will return each MPEG-7 document wherein a semantic object containing the specified word is found. If two or three nodes and one or two relations are used to define the search graph, the repository of MPEG-7

documents is filtered by the terms defined as objects or relations. If, for example, the graph in figure 7 below is used for search, all documents which contain semantic objects, which contain the terms “Manipal”, “SMU” and “MU”, and a semantic relation containing the term “partOf” are taken from the repository and checked if there is also a structural match with the given graph.



Fig. 7: Starting a semantic search using a graph as input



Fig. 8: Possible search graphs, which are supported in this prototype, are sub graphs of the maximum graph shown in figure 7

The retrieval mechanism follows modular system architecture, an *XPath* statement is given to a class implementing the interface *RetrievalEngine* and the results are received as list of HTML documents, which can be visualized using standard Java Swing components. The only retrieval engine implemented yet is the "*FileSystemRetrievalEngine*", which collects all MPEG-7 documents from a specified directory and its sub-directories and executes the given *XPath* statement. If a matching document is found it is transformed into HTML, using *XSLT*. This HTML result visualization is added to a list of results, which is ordered by relevance. Relevance is calculated using the number of nodes matching the *XPath* statement used as input. Another retrieval engine implementation would connect for instance to an *XML* database, which would result in a significant speedup executing the *XPath* statements.

In case of a content based image search each MPEG-7 document has to be loaded and the required descriptor is located using *XPath*. This descriptor has to be compared to the sample descriptor used as search parameter to calculate relevance. These results are put into a list ordered ascending by relevance, though a relevance of zero would show an exact match. Using a database the comparison of the descriptors has to be implemented on database side like a stored procedure, a server object or a similar mechanism, because of speed issues.

XIV. RESULTS

A common problem with retrieval of *XML* documents is the speed, although Oracle and other big players in creating databases are already working on a possible solution. An also well-known fact is the insufficiency of *XPath* as query language. The upcoming standard *XQuery* represents a possible solution. Nevertheless, most manufacturers and database vendors do not support it yet.

The computation of graph visualizations in a very semantic way with minimum crossings is also complicated in this context. For instance loading a semantic description and reading it does not prove as complicated, but arranging and visualizing the same graph can be quite tricky without generating a complete visual mess, which can only bring confusion to the user. Mostly a semantic description has a central element, take for example an image of a conversation between two persons. The central element is the conversation, therefore it should be placed in the center of the visualization, and the objects representing the persons

taking part in the conversation should be placed around this element along with a place, a time and a context for the conversation. Basically a very similar effect can be achieved if visualization with a minimum number of edge crossings is calculated, because the central element takes part in most edges and therefore it is placed in the middle.

Another problem is that different users produce different descriptions for the same media instance. Also MPEG-7 defines an inverse for each semantic relation; as a result a user can choose either a relation or its inverse to create the description. Therefore the retrieval mechanism must take care of these differences, like combined searching for relations and inverse relations and computing a similarity between semantic objects to identify objects which define the same person, thing or state but differ in the way they are described.

Finally the retrieval using a semantic graph as search parameter is not only a simple graph to graph matching based on a simple description standard but has to mind some parameters set from the MPEG-7 standard. In addition to the above mentioned inverse relations, MPEG-7 allows to integrate objects by reference. This means that the objects are used in the graph, but they are not fully described inside the descriptor itself, but are only referencing the object, which is defined in another part of the document or even in another document.

Although MPEG-7 defines similarity measurement for low-level content based descriptors it fails to define those measurement methods for calculating the similarity of two semantic graphs, so a generalized method has to be found and proposed.

We have carried out experiment to compare the recall and precision for retrieving a given set of semantic-linked image networks under the same set of query conditions. Each network contains thirty image nodes.

Figure 9 compares the change of precision and recall with the change of the number of semantic links. Figure 10 shows the recall and precision change with the number of types of semantic links.

We can see that the retrieval efficiency depends not only on the number of semantic links but also the types of the semantic links included in a semantic-linked network. We are carrying out experiments with larger scale and random samples to verify this phenomenon.

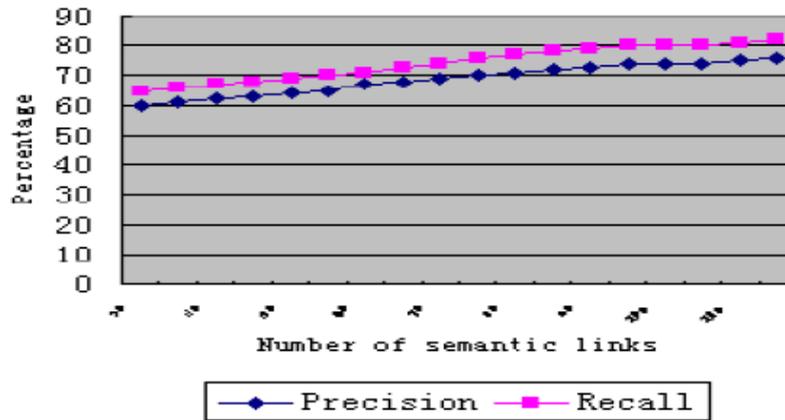


Figure 9: Recall and precision change with the number of semantic links.

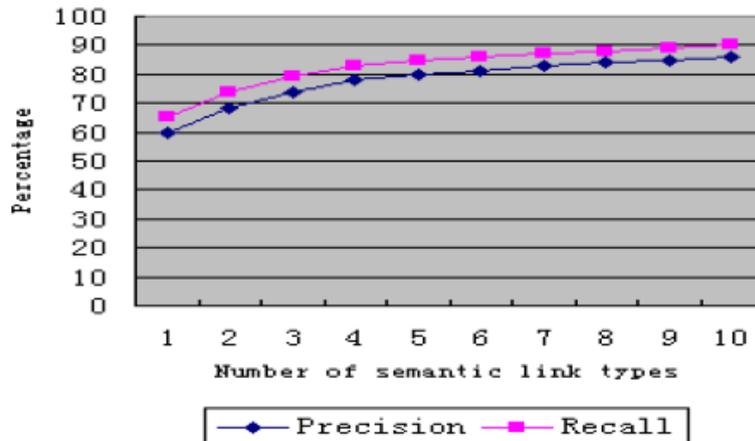


Figure 10: Recall and precision change with the number of types of the semantic links.

The underline premise of the proposed approach is that the image retrieval efficiency depends on the providers’ semantic description on the provided images. If no semantic links are established, the proposed approach becomes traditional text-based or content-based approaches. The hyperlink-based approach also depends on the pre-established hyperlinks.

XV. CONCLUSION

The application areas most likely to benefit from the adoption of CBIR are those where level 1 technique can be directly applied. Users and managers of image collections need to be aware of the capabilities of CBIR technology, and to be capable of making informed decisions about adoption. Specifically:

- Managers of image collections in specialist areas such as fingerprints or trademark images, involving image matching

by appearance, should be encouraged to investigate possible adoption of CBIR technology in the near future.

- Managers of video libraries should certainly investigate the possibility of using one of the proprietary video asset management packages.

- Managers of general-purpose image collections such as art galleries or photographic libraries should be encouraged to keep a watching brief on developments in CBIR, through articles in the specialist press and conferences relating to image retrieval particularly to hybrid text/image feature indexing and cross-media retrieval.

- Software developers or information providers with products designed to handle images, which currently lack

CBIR capabilities, also need to make informed decisions about whether CBIR would add value to their products. In conclusion, CBIR is clearly a technology with potential. The next five to ten years will reveal whether this potential can be turned into solid achievement. Our view is that at present the omen is favorable.

REFERENCES

- [Benitez2002] Semantics of Multimedia in MPEG-7, Benitez Ana B. Et al, 2002, International Conference on Image Processing 2002, Proceedings Volume: 1, Page(s): 137-140
- [Caria2000] Image Description and Retrieval Using MPEG-7 Shape Descriptors, Carla Zibreira, Fernando Pereira, ECDL 2000, Proceedings, pp. 332-335
- [Nack1999] Everything You Want to Know About MPEG-7, F. Nack, A. Lindsay, Part 1, IEEE Multimedia, 6(3), July-September 1999, 65-77
- [MPEG2001] Multimedia Content Description Interface – Part 5: Multimedia Description Schemes, MPEG, 23.10.2001
- [Hunter2000] Proposal for Integration of DublinCore and MPEG-7, Hunter, Jane, October 2000
- [Bormans2002] Bormans, Jan, Hill, Keith, "MPEG-21 Overview", Moving Picture Expert Group MPEG, Shanghai, October 2002, URL: http://www.chiariglione.org/mpeg/standards/mpeg_21/mpeg-21.htm
- [DelBimbo1999] Del Bimbo, Alberto, "Visual Information Retrieval", Morgan Kaufmann Publishers, 1999.
- [Flickner1995] Flickner, Myron, Sawhney, Harpreet, Niblack, Wayne, Ashley, Jonathan, Huang, Qian, Dom, Byron, Gorkani, Monika, Hafner, Jim, Lee, Denis, Petkovic, Dragutin, Steele, David, Yanker, Peter, "Query by image and video content: The QBIC system", IEEE Computer, 28(9), pp. 23-32, September 1995
- [IBM2004a] IBM Research, "MARVEL: MPEG-7 Multimedia Search Engine", <http://www.research.ibm.com/marvel/>, last visited: 25.11.2004
- [IBM2004b] IBM Research, "VideoAnnEx Annotation Tool", <http://www.research.ibm.com/VideoAnnEx/>, last visited: 25.11.2004
- [Kosch03] Kosch, Harald, "Distributed Multimedia Database Technologies supported by MPEG-7 and MPEG-21", CRC Press, November 2003
- [Lossau2004] Lossau, Norbert, "Search Engine Technology and Digital Libraries - Libraries Need to Discover the Academic Internet", D-Lib Magazine, Vol. 10, Num. 6, June 2004
- [Lucene] The Apache Software Foundation, "Jakarta Lucene", a Java based search engine, URL: <http://jakarta.apache.org/lucene>
- [Lux2004] Lux, Mathias, Klieber, Werner, Granitzer, Michael, "Caliph & Emir: Semantics in Multimedia Retrieval and Annotation", 19th International CODATA Conference, Berlin, Germany, November 2004
- [Martínez2003] Martínez, José M., "MPEG-7 Overview", Moving Picture Expert Group MPEG, Pattaya, March 2003, URL: <http://www.chiariglione.org/mpeg/standards/mpeg-7/mpeg-7.htm>
- [Mayer2004] Mayer, Harald, Bailer, Werner, Neuschmied, Helmut, Haas, Werner, Lux, Mathias, Klieber, Werner, "Content-based video retrieval and summarization using MPEG-7", in Proceedings of Internet Imaging V, IS&T/SPIE 16th Annual Symposium, Electronic Imaging, San Jose, California USA, 2004
- [Smeulders2000] Smeulders, A.W.M., Worring, M., Santini, S., Gupta, A., Jain, R. "Content-based image retrieval at the end of the early years", Pattern Analysis and Machine Intelligence, IEEE Transactions on, Vol. 22, No. 12, pp. 1349-1380, December 2000
- [Summann2004] Summann, Friedrich, Lossau, Norbert, "Search Engine Technology and Digital Libraries - Moving from Theory to Practice", D-Lib Magazine, Vol. 10, Num. 6, September 2004
- [Tsinaraki2003] Tsinaraki, Chrisa, Fatourou, Eleni, Christodoulakis, Stavros, "An Ontology-Driven Framework for the Management of Semantic Metadata Describing Audiovisual Information", in Proceedings 15th Conference on Advanced Information Systems Engineering CAiSE 2003, pp. 340-356, Springer, LNCS, 2003
- [Wactlar2002] Wactlar, Howard D., "Extracting and Visualizing Knowledge from Film and Video", in Proceedings 2nd International Conference on Knowledge Management I-KNOW '02, Journal of Universal Computer Science, July 2002

Efficient use of MPEG-7 Color Layout and Edge Histogram Descriptors in CBIR Systems

Balasubramani R
Assistant Professor-IT
Sikkim Manipal University- DDE
1 Floor, Syndicate House
Manipal- 576104
Karnataka, India
E-mail: microtech_balu@yahoo.com

Dr.V.Kannan
Dean
Center for Information
Bharath University
Chennai- 600073
Tamil Nadu, India
Email-drvkannan62@yahoo.com

Abstract: MPEG-7 Visual Standard specifies a set of descriptors that can be used to retrieve similar images from digital photo repository. Among them, the Color Layout Descriptor (CLD) represents the spatial distribution of colors in an image. The Edge Histogram Descriptor (EHD) describes edge distribution with a histogram based on local edge distribution in an image. These two features are very powerful features for CBIR systems, especially sketch-based image retrieval. Further, combining color and texture features in CBIR systems leads to more accurate results for image retrieval. In both the Color Layout Descriptor (CLD) and the Edge Histogram Descriptor (EHD), image features like color and edge distribution can be localized in separate 4 x 4 sub-images. The visual features of each sub-image were characterized by the representative colors of the CLD as well as the edge histogram of the EHD at that sub-image using weighing factors. As most CBIR systems depend on query images and as most users of such CBIR systems are non-professional, we hope to develop a more user-friendly prototype, NWCIBR, implementing both CLD and EHD features.

Keywords:

MPEG-7, Color Layout Descriptor (CLD), Edge Histogram Descriptor (EHD).

I. INTRODUCTION

Color is the most basic attribute of visual contents. MPEG-7 Visual defines five different description tools, each of which represents a different aspect of the color attribute. Of these the Color Layout Descriptor (CLD) represents the spatial layout of color images in a very compact form. It is based on generating a tiny (8x8) thumbnail of an image, which is encoded via Discrete Cosine Transform (DCT) and quantized. As well as efficient visual matching, this also offers a quick way to visualize the

appearance of an image, by reconstructing an approximation of the thumbnail, by inverting the DCT.

The histogram is the most commonly used structure to represent any global feature composition of an image [Jain1996]. It is invariant to image translation and rotation, and normalizing the histogram leads to scale invariance. Exploiting the above properties, the histogram is very useful for indexing and retrieving images. Edges in images constitute an important feature to represent their content. Also, human eyes are sensitive to edge features for image perception. One way of representing such an important edge feature is to use a histogram. An edge histogram in the image space represents the frequency and the directionality of the brightness changes in the image. It is a unique feature for images, which cannot be duplicated by a color histogram or the homogeneous texture features. To represent this unique feature, in MPEG-7, there is a descriptor for edge distribution in the image [ISO/IEC2001]. This Edge Histogram Descriptor (EHD) proposed for MPEG-7 expresses the local edge distribution in the image. That is, since it is important to keep the size of the descriptor as compact as possible for efficient storage of the metadata, the normative MPEG-7 edge histogram is designed to contain only 80 bins describing the local edge distribution. These 80 histogram bins are the only standardized semantics for the MPEG-7 EHD.

II. COLOR LAYOUT DESCRIPTOR (CLD) EXTRACTION

During the CLD Extraction Process, it is assumed that the image consists of three color channels, R, G and B. The CLD descriptor was obtained through the following steps (figures 1 and 2):

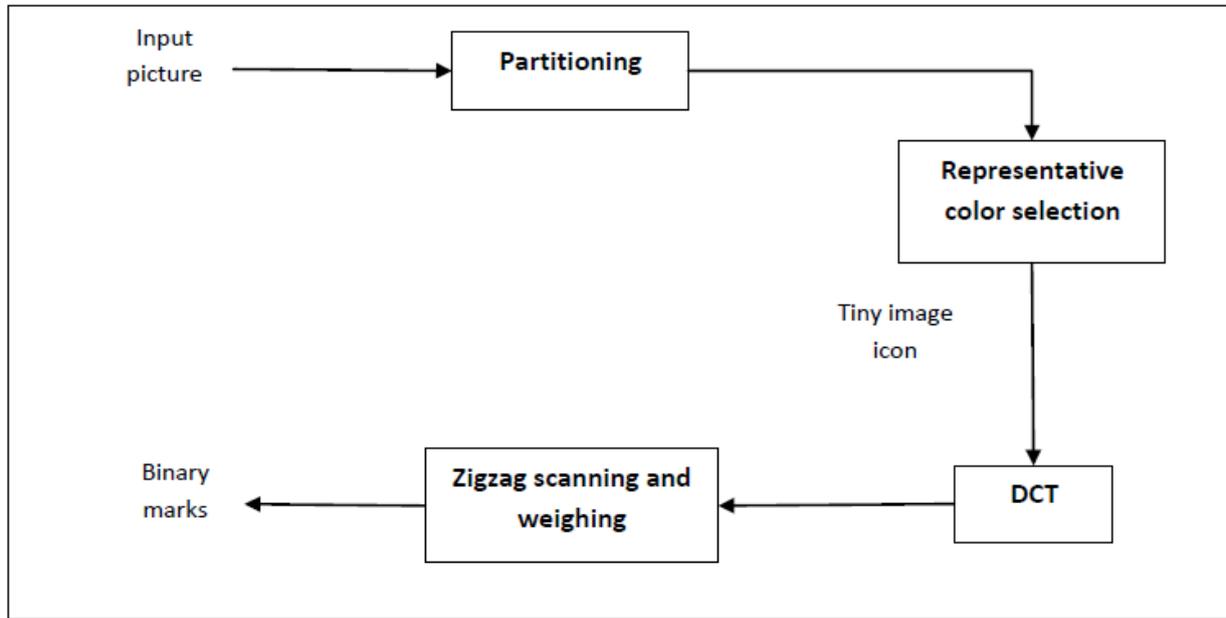


Fig. 1: The CLD extraction process

1. The image was loaded using *opencv* and the width and height of the image was obtained, from which the block width and block height of the *CLD* were calculated by dividing by 8. The division was done using truncation, so that if the image dimensions were not divisible by 8, the outermost pixels are not considered in the descriptor.
2. Using the obtained information, the image data was parsed into three *4D* arrays, one for each color component, where a block can be accessed as a whole and pixels within each block could also be accessed by providing the index of the block and the index of the pixel inside the block.
3. A representative color was chosen for each block by averaging the values of all the pixels in each block. This results in three 8×8 arrays, one for each color component.

4. This step is directly visualized in the first window of figure 2.
5. Each 8×8 matrix was transformed to the *YCbCr* color space.
6. Each 8×8 matrix was transformed to the *YCbCr* color space.
7. These will be again transformed by 8×8 *DCT* (Discrete Cosine Transform) to obtain three 8×8 *DCT* matrices of coefficients, one for each *YCbCr* component.
8. The *CLD* descriptor was formed by reading in zigzag order six coefficients from the *Y-DCT* matrix and three coefficients from each *DCT* matrix of the two chrominance components. The descriptor is saved as an array of 12 values.

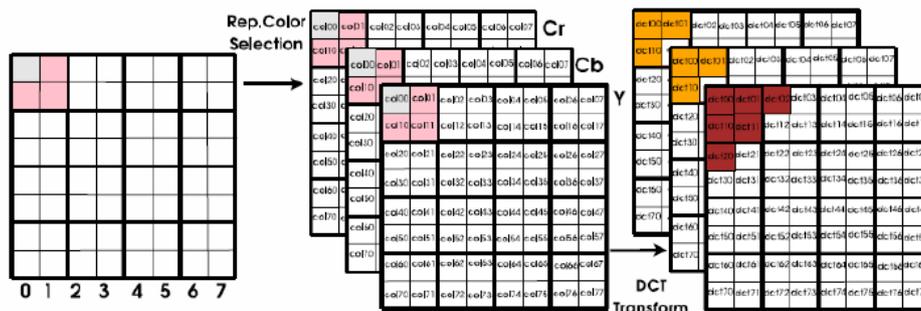


Fig. 2: The CLD Extraction Process - steps

III. DEFINITION AND SEMANTICS OF THE EDGE HISTOGRAM DESCRIPTOR (EHD)

The *EHD* basically represents the distribution of 5 types of edges in each local area called a sub-image. As shown in figure 3, the sub-image is defined by dividing the image space into 4x4 non-overlapping blocks. Thus, the image partition always yields 16 equal-sized sub-images regardless of the size of the original image. To characterize the sub-image, we then generate a histogram of edge distribution for each sub-image. Edges in the sub-images are categorized into 5 types: vertical, horizontal, 45-degree diagonal, 135-degree diagonal and non-directional edges (figure 4) [CheeSunWon1997]. Thus, the histogram for each sub-image represents the relative frequency of occurrence of the 5 types of edges in the corresponding sub-image. As a result, as shown in figure 5, each local histogram contains 5 bins. Each bin corresponds to one of 5 edge types. Since there are 16 sub-images in the image, a total of $5 \times 16 = 80$ histogram bins is required (figure 6). Note that each of the 80-histogram bins has its own semantics in terms of location and edge type. For example, the bin for the horizontal type edge in the sub-image located at (0, 0) in figure 3 carries the information of the relative population of the horizontal edges in the top-left local region of the image.

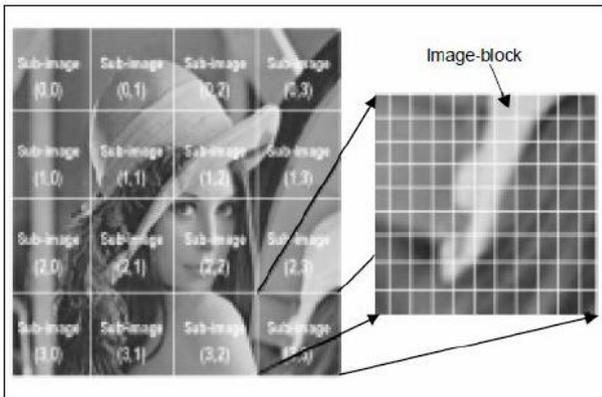


Fig. 3: Definition of sub-image and image-block

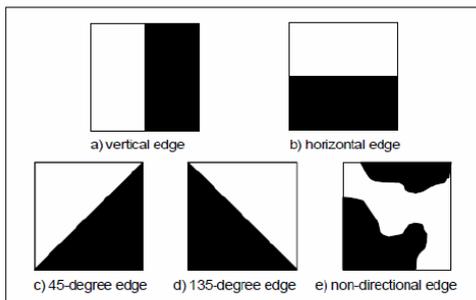


Fig. 4: Five types of edges

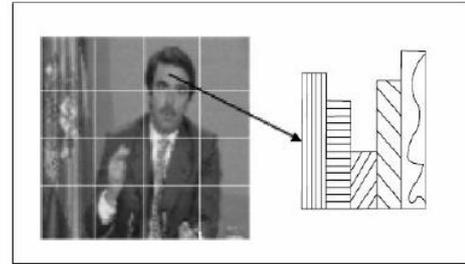


Fig. 5: Five types of edge bins for each sub-image

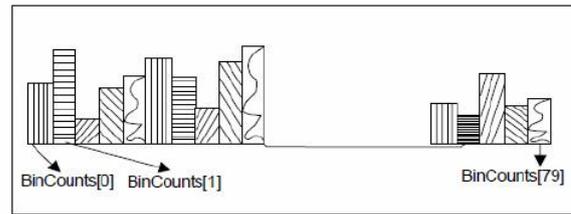


Fig. 6: 1-D array of 80 bins of EHD

The semantics of the 1-D histogram bins form the normative part of the MPEG-7 standard descriptor. Specifically, starting from the sub-image at (0,0) and ending at (3,3), 16 sub-images are visited in the raster scan order and corresponding local histogram bins are arranged accordingly. Within each sub-image, the edge types are arranged in the following order: vertical, horizontal, 45-degree diagonal, 135-degree diagonal, and non-directional. Table 1 summarizes the complete semantics for the EHD with 80 histogram bins. Of course, each histogram bin value should be normalized and quantized. For normalization, the number of edge occurrences for each bin is divided by the total number of image-blocks in the sub-image.

The image-block is a basic unit for extracting the edge information. That is, for each image-block, we determine whether there is at least an edge and which edge is predominant.

When an edge exists, the predominant edge type among the 5 edge categories is also determined. Then, the histogram value of the corresponding edge bin increases by one. Otherwise, for the monotone region in the image, the image-block contains no edge. In this case, that particular image-block does not contribute to any of the 5 edge bins.

Table 1: Semantics of local edge bins.

Histogram bins	Semantics
BinCounts[0]	Vertical edge of sub-image at (0,0)
BinCounts[1]	Horizontal edge of sub-image at (0,0)
BinCounts[2]	45-degree edge of sub-image at (0,0)
BinCounts[3]	135-degree edge of sub-image at (0,0)
BinCounts[4]	Non-directional edge of sub-image at (0,0)
BinCounts[5]	Vertical edge of sub-image at (0,1)
:	:
BinCounts[74]	Non-directional edge of sub-image at (3,2)
BinCounts[75]	Vertical edge of sub-image at (3,3)
BinCounts[76]	Horizontal edge of sub-image at (3,3)
BinCounts[77]	45-degree edge of sub-image at (3,3)
BinCounts[78]	135-degree edge of sub-image at (3,3)
BinCounts[79]	Non-directional edge of sub-image at (3,3)

Table 2: Quantization table for 5 types of edges

BinCounts (3bits/bin)	Representative values for vertical edges	Representative values for horizontal edges	Representative values for 45-degree diagonal edges	Representative values for 135-degree diagonal edges	Representative values for non-directional edges
000	0.010867	0.012266	0.004193	0.004174	0.006778
001	0.057915	0.069934	0.025852	0.025924	0.051667
010	0.099526	0.125879	0.046860	0.046232	0.108650
011	0.144849	0.182307	0.068519	0.067163	0.166257
100	0.195573	0.243396	0.093286	0.089655	0.224226
101	0.260504	0.314563	0.123490	0.115391	0.285691
110	0.358031	0.411728	0.161505	0.151904	0.356375
111	0.530128	0.564319	0.228960	0.217745	0.450972

IV. EDGE HISTOGRAM DESCRIPTOR (EHD) EXTRACTION

Since the *EHD* describes the distribution of non-directional edges and non-edge cases as well as four directional edges, the edge extraction scheme should be based on the image-block as a basic unit for edge extraction rather than on the pixel. That is, to extract directional edge features, we need to define small square image-blocks in each sub-image as shown in figure 7. Specifically, we divide the image space into non-overlapping square image-blocks and then extract the edge information from them. Note that, regardless of the image size, we divide the image space into a fixed number of image-blocks. The purpose of fixing the number of image-blocks is to cope with the different sizes (resolution) of the images. That is, by fixing the number of image blocks, the size of the image-block becomes variable and is proportional to the size of the whole image. The size of the image-block is assumed to be a multiple of 2. Thus, it is sometimes necessary to ignore the outmost pixels in the image to satisfy that condition.

Consequently, each image-block is classified into one of the 5 types of edge blocks or a non-edge block. Although the non-edge blocks do not contribute to any histogram bins, each histogram bin value is normalized by the total number of image-blocks including the non-edge blocks. This implies that the summation of all histogram bin values for each sub-image is less than or equal to 1. This, in turn, implies that the information regarding non-edge distribution in the sub-image (smoothness) is also indirectly considered in the *EHD*.

Now, the normalized bin values are quantized for binary representation. Since most of the values are concentrated within a small range (say, from 0 to 0.3), they are non-linearly quantized to minimize the overall number of bits. Table 2 shows the representative values for coded bits for each edge type. The normalized 80 bin values are non-linearly quantized and fixed length coded with 3 bits/bin as defined in table 2. *BinCounts[0]*, ...and *BinCounts[79]* represent the final coded bits for the *EHD*.

A simple method to extract an edge feature in the image-block is to apply digital filters in the spatial domain. To this end, we first divide the image-block into four sub-blocks as shown in figure 7 [ISO/IEC1999]. Then, by assigning labels for four sub-blocks from 0 to 3, we can represent the average gray levels for four sub-blocks at (i,j) th image-block as $a0(i,j)$, $a1(i,j)$, $a2(i,j)$, and $a3(i,j)$ respectively. Also, we can represent the filter coefficients for vertical, horizontal, 45-degree diagonal, 135-degree diagonal, and non-directional edges as $fv(k)$, $fh(k)$, $fd-45(k)$, $fd-135(k)$, and $fnd(k)$, respectively, where $k=0, \dots, 3$ represents the location of the sub-blocks. Now, the respective edge magnitudes $mv(i,j)$, $mh(i,j)$, $md-45(i,j)$, $md-135(i,j)$, and $mnd(i,j)$ for the (i,j) th image-block can be obtained as follows:

$$m_v(i, j) = \left| \sum_{k=0}^3 a_k(i, j) \times f_v(k) \right| \quad (1)$$

$$m_h(i, j) = \left| \sum_{k=0}^3 a_k(i, j) \times f_h(k) \right| \quad (2)$$

$$m_{d-45}(i, j) = \left| \sum_{k=0}^3 a_k(i, j) \times f_{d-45}(k) \right| \quad (3)$$

$$m_{d-135}(i, j) = \left| \sum_{k=0}^3 a_k(i, j) \times f_{d-135}(k) \right| \quad (4)$$

$$m_{nd}(i, j) = \left| \sum_{k=0}^3 a_k(i, j) \times f_{nd}(k) \right| \quad (5)$$

If the maximum value among 5 edge strengths obtained from (1) to (5) is greater than a threshold (T_{edge}) as in (6), then the image-block is considered to have the corresponding edge in it. Otherwise, the image-block contains no edge.

$$\max\{m_v(i, j), m_h(i, j), m_{d-45}(i, j), m_{d-135}(i, j), m_{nd}(i, j)\} > T_{edge} \quad (6)$$

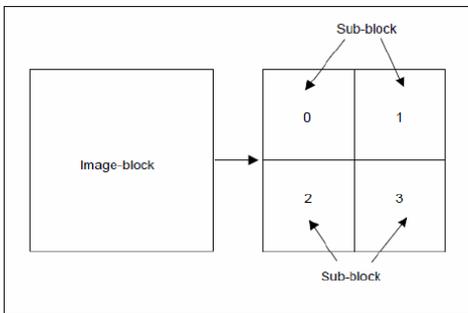


Fig. 7: Sub-blocks and their labelling

In MPEG-7 XM Document, a set of filter coefficients, depicted in figure 8, is recommended. Note that the filter coefficients in figure 8, especially the non-directional edge filter, appear somewhat heuristic. In fact, the non-directional edges by definition do not have any specific directionality. So, it is hard to find filter coefficients that are applicable for all types of non-directional edges. To avoid this problem, we can first check whether the image-block can be classified into one of a monotone block and four directional edge blocks. If the image-block does not belong to any of the monotone or four directional edge blocks, then we classify it as a non-directional block. The flow chart of this method is shown in figure 9.

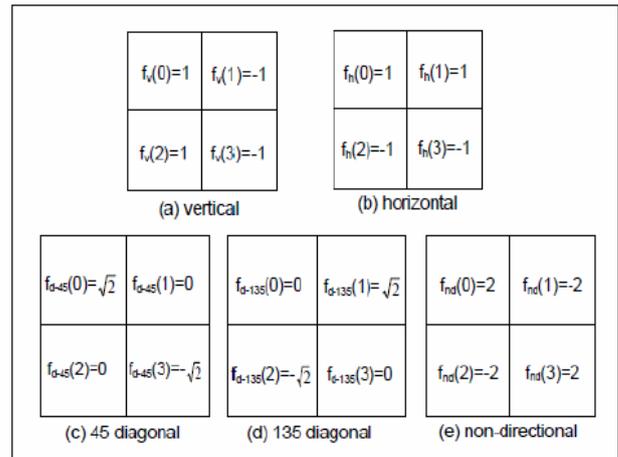


Fig. 8: Filter coefficients for edge detection

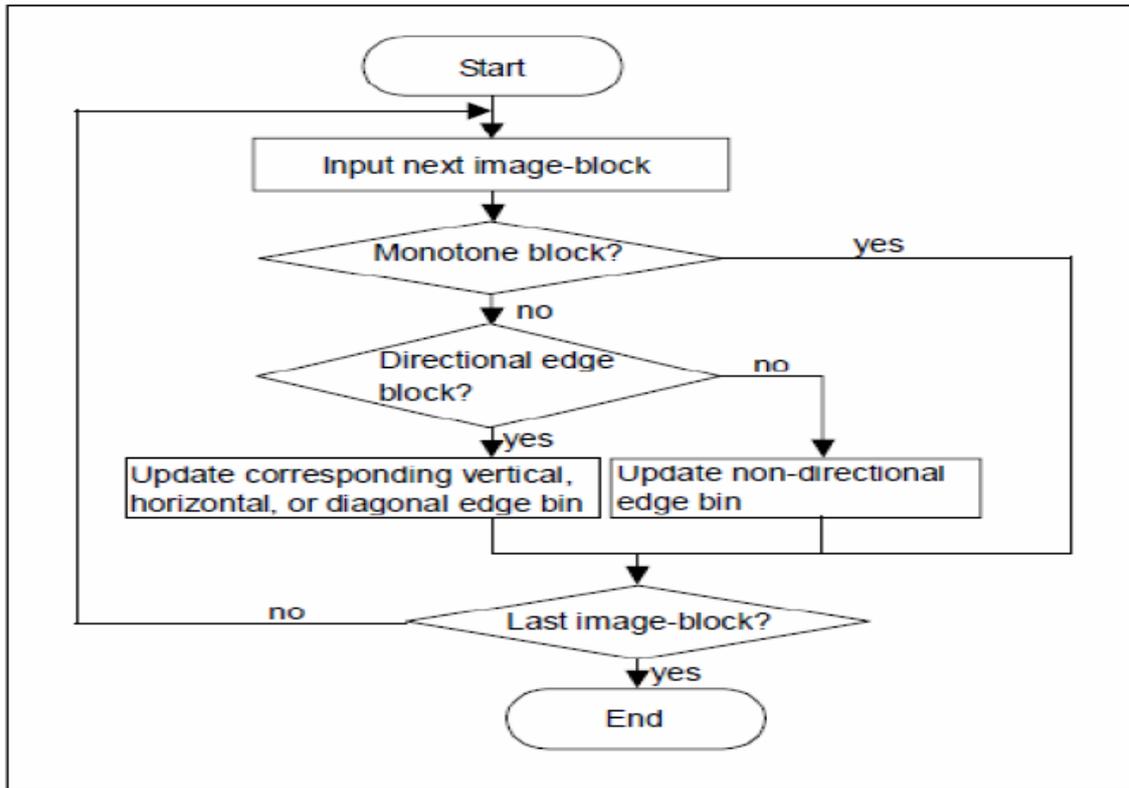


Fig. 9: Flowchart of edge classification without using the non-directional edge-filter.

V. RESULTS

For our experiments, we set the total number of image-blocks at 1100 and the threshold for edge detection (T_{edge}) at 11. For all our experiments, we used the image data set from the MPEG-7 Core Experiment (CE), which has 11639 images in the database. Most of the images in the database are natural images from “Correl One Million Gallery” and others are from images provided by the proponents of other image descriptors such as color and homogeneous texture descriptors. From 11639 images, we used 51 images, which were selected by MPEG-7 CE participants as query images. The ground truths that we used in our experiments were determined by three participants of the MPEG-7 CE. Each participant proposed from 3 to 33 ground truth images for each query image and they were approved by the other two CE participants. As a measure of retrieval accuracy, we used the Average Normalized Modified Retrieval Rank (ANMRR). Precision and Recall are well-known measures for the retrieval performance. They are basically a “hit-and-miss” counter. That is, the performance is based on the number of retrieved images, which have similarity measures that are greater than a given threshold. For more specific comparisons, however, we also need rank information among the retrieved images. ANMRR is the measure that

exploits the rank of the retrieved images as well. It was developed during the MPEG-7 standardization activity and was used for the MPEG-7 Core Experiments (CE). Note that

lower ANMRR values indicate more accurate retrieval performance.

Table 3: Retrieval performance (ANMRR)

	2 bits/bin	3 bits/bin	4 bits/bin	5 bits/bin
Matching with local histogram	0.396012	0.336060	0.317815	0.324698

Table 3 shows the results of the retrieval accuracy for different bits-per-bins numbers. As the bits-per-bin increases, the ANMRR decreases. However, figure 10 demonstrates that a further decrease in the ANMRR is not significant beyond 3 bits-per-bin. This is why 3 bits-per-bin was chosen in MPEG-7.

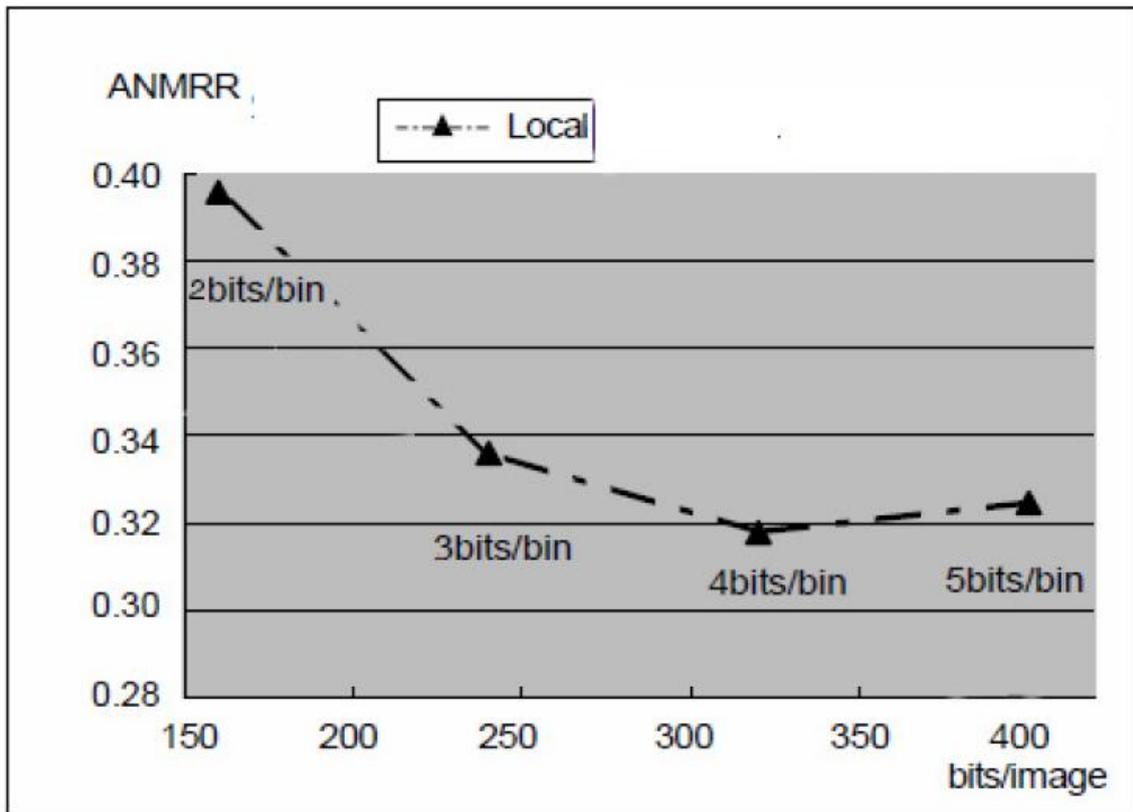


Fig. 10: Graph showing ANMRR vs bits-per-bin

Although MPEG-7 defines similarity measurement for low-level content based descriptors it fails to define those measurement methods for calculating the similarity of two semantic graphs, so a generalized method has to be found and proposed.

VI. CONCLUSION

MPEG-7 matches many of the current requirements for a metadata standard for usage in a personal digital photo library and it defines a lot more useful descriptors, which could be integrated as features in such libraries. In addition it is not only a standard for describing the content of images, but it also defines ways to annotate video and audio documents and it is prepared for general usage with multimedia data.

In conclusion, CBIR is clearly a technology with potential. The next five to ten years will reveal whether this potential can be turned into solid achievement. Our view is that at present the omen is favorable.

REFERENCES

- [Swain1991] M.J. Swain and D.H. Ballard, "Color Indexing," *International Journal Of Computer Vision*, vol.7-1, 1991, pp. 11-32.
- [Jain1996] A.K. Jain and A. Vailaya, "Image Retrieval Using Color and Shape," *Pattern Recognition*, vol. 29, no. 8, 1996, pp. 1233-1244.
- [ISO/IEC2001] ISO/IEC/JTC1/SC29/WG11: "MPEG-7 XM Document: MPEG-7 Visual Part Experimentation Model Version 10.0," MPEG Document N4063, Singapore, Mar. 2001.
- [CheeSunWon1997] Chee Sun Won and Dong Kwon Park, "Image Block Classification and Variable Block Size Segmentation Using a Model-Fitting Criterion," *Optical engineering*, Aug. 1997, pp. 2204-2209.
- [ISO/IEC1999] ISO/IEC/JTC1/SC29/WG11: "Core Experiment Results for Spatial Intensity Descriptor (CT4)," MPEG document M5374, Maui, Dec. 1999.
- [ISO/IEC1999] ISO/IEC/JTC1/SC29/WG11: "Description of Core Experiments for MPEG-7 Color/Texture Descriptors," MPEG document N2929, Melbourne, Oct. 1999.

Computation of Merging Points in Skeleton Based Images

Mrs. J.KomalaLakshmi¹, Research Scholar,
Email: ashwathraju@yahoo.com

Dr.M.Punithavalli²
Email: mpuntihavalli@yahoo.co.in

Abstract-During the midst of 21st century, hope our human computers will reach the world of galaxy with extraordinary robots at hand and computers at mind. We can design and develop new robots or machines that can outperform all the works in the galaxy world. In order to design such a robots, we can utilize the best from Digital image processing. In this research article, the author proposes a new Merge Vertex Detection Algorithm for finding the merging points of the image in skeleton representation. Using those merging points, we can design and develop new machinery parts of different variety of shapes and that can be implemented in reality to produce a galaxy man.

Keywords.

ValenceSkeletonPoints(vsp), CoreSkeletonPoint(csp),
SamplingError, Merge Vertex, Discrete skeleton

I. INTRODUCTION

In all the research areas such as image retrieval and computer graphics, character recognition, image processing, and the analysis of biomedical images [1], the skeleton plays a major role for object representation and recognition. Skeleton-based representations are the abstraction of objects, which contain both shape features and topological structures of original objects. Because of the skeleton's importance, many skeletonization algorithms have been developed to represent and measure different shapes. Many researchers have made great efforts to recognize the generic shape by matching skeleton structures represented by graphs or trees [2], [3], [4], [5], [6],[7]. The most significant factor constraining the matching of skeletons is the skeleton's sensitivity to an object's boundary deformation: little noise or a variation of the boundary often generates redundant skeleton branches that may seriously disturb the topology of the skeleton's graph. To overcome a skeleton's instability of boundary deformation, a variety of techniques have been suggested for matching and recognizing shapes. Zhu and Yuille [4] generate more than one possible skeleton graph to overcome unreliability.

A similar shape descriptor based on the self similarity of a smooth outline is presented in [5]. Aslan and Tari [6] posit an unconventional approach to shape recognition using

unconnected skeletons in the course level. While their approach leads to stable skeletons in the presence of boundary deformations, only rough shape classification can be performed since the obtained skeletons do not represent any shape details. This paper proposes a new algorithm for identifying the skeleton point's viz., Core skeleton point and Valance skeleton point in an effective manner and to store them so that it can be utilized for further processing of the same images. The skeleton formed by these Csp's and vsp's minimizes the boundary deformations and produce topologically stable skeletons and thus we can reconstruct and deduce new images creatively.

II. RELATED WORK.

A.Skeletons

We characterize desirable properties of skeletons [8]. The skeleton of a single connected shape that is useful for skeleton-based recognition should have the following properties: (1) it should preserve the topological information of the original object; (2) the position of the skeleton should be accurate; (3) it should be stable under small deformations; (4) it should contain the centers of maximal disks, which can be used for reconstruction of original object; (5) it should be invariant under Euclidean transformations, such as rotations and translations, and (6) it should represent significant visual parts of objects.

B. Skeletonization

Skeletonization is a technique used to extract skeletons of the Objects [9]. The skeletonization methods are widely used because, it

1. Generates reduced amount data, to be processed.
2. Minimizes the processing time.
3. The critical points like end-points, junction-points, and connection among the components can be extracted effectively.
4. Shape analysis can be more easily made.

The skeletonization algorithms can broadly be classified into four types:

The first type is thinning algorithms, such as those with shape thinning and the wave front/grassfire transform [10], [11], [12]. These algorithms iteratively remove border points, or move to the inner parts of an object in determining an object's skeleton. These methods usually preserve the topology of the original object with many redundant branches, but they are quite sensitive to noise and often fail to localize the accurate skeletal position. In addition, it is important to determine a good stop criterion of this iterative process.

The second type is the category of discrete domain algorithms based on the Voronoi diagram [13], [14]; these methods search the locus of centers of the maximal disks contained in the polygons with vertices sampled from the boundary. The exact skeleton can be extracted as the sampling rate increases, but the time of computation is usually prohibitive. The obtained skeleton is extremely sensitive to local variance and boundary noise, so that complicated skeleton bunches need to be pruned [13], [15]. The third type of algorithms is to detect ridges in a distance map of the boundary points [16], [12]. Approaches based on distance maps usually ensure accurate localization but neither guarantees connectivity nor completeness [16],[17]. Under the completeness the skeleton branches representing all significant visual parts are present (6). The fourth type of algorithms is based on mathematical morphology [18], [19]. Usually, these methods can localize the accurate skeleton [18], but May not guarantee the connectivity of the skeleton [19]. All of the obtained skeletons are subjected to the skeleton's sensitivity.

C. Constraints of skeletons –based shape matching.

There are two main factors that constraint the performance of skeleton-based shape matching: 1) skeleton's sensitivity to object's boundary deformation: little noise or variation of boundary often generates redundant skeleton branches that may disturb the topology of skeleton's graph seriously; 2) the time cost for extraction of skeleton and matching skeleton trees/graphs cannot satisfy the requirement of fast shape retrieval. The performance of skeleton matching depends directly on the property of shape representation.

Therefore, to prune the grassy skeletons into the visual skeletons is usually inevitable [20]. There are two main pruning methods: (1) based on significance measures assigned to skeleton points [13],[21],[16], and (2) based on boundary smoothing before extracting the skeletons [22],[23]. In particular, curvature flow smoothing still has some significant problems that makes the position of skeletons shift and have difficulty in distinguishing noise from low frequency shape information on boundaries [21]. A different kind of smoothing is proposed in [24]. Great progress has been made in the type (1) of pruning approaches that define a significance measure for skeleton points and remove points whose significance is low. Shaked

and Bruckstein [21] give a complete analysis and compare such pruning methods. Propagation velocity, maximal thickness, radius function, axis arc length, and the length of the boundary unfolded belong to the common significance measures of skeleton points. Ogniewicz et al [13] present a few significance measures for pruning complex Voronoi skeletons without disconnecting the skeletons. Siddiqi et al combine a flux measurement with the thinning process to extract a robust and accurate connected skeleton [18].

D. Drawbacks of Pruning.

Drawback 1. Many of them are not guaranteed to preserve the topology of a complexly connected shape (e.g., a shape with holes). This is illustrated in Fig. 2.D.b, where the skeleton in Fig 2.D.b violates the topology of the input skeleton in Fig 2.D.a. This skeleton was obtained by the method in [16]. However, many methods described above would lead to topology violation, particularly all methods presented in [21] (including the method of Ogniewicz at al. [13]). These methods are guaranteed to preserve topology for simply connected objects (objects with a single contour), but not for objects with more than one contour like the can in Fig. 2.D.a



Fig 2.D.a The input object.

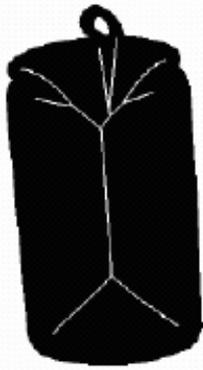


Fig 2 .D.b A pruned skeleton violates the topology.

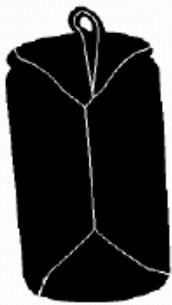


Fig 2.D.c A pruned skeleton obtained guaranteed to preserve the topology.

The second drawback of the methods described above is that main skeleton branches are shortened and short skeleton branches are not removed completely. This may lose important shape information and seriously compromise the structure of the skeletons.

The third drawback is that usually only the local significance of the skeleton points is considered, and the global information of the shape is discarded. However, the same part may represent an important shape feature for one shape while it may represent noise for a different shape. The fourth drawback is that pruning results may be different for different scales as pointed out in [25].

III. .EXISTING METHODS.

A. Discrete Curve Evolution (DCE)

Returning to Blum's definition of the skeleton, every skeleton point is linked to boundary points that are tangential to its maximal circle. These are called **generating points**. The main idea is to remove all skeleton points whose generating points all lie on the same contour segment. This works for any contour partition in segments, but some partitions yield better results than other.

A) Good partition of the Contour into Segments.

Thus, in our framework, the question of skeleton pruning is reduced to finding a good partition of the contour into segments. We obtain such partitions with the process of

Discrete Curve Evolution (DCE) [26], [27], [28], which we briefly introduce as follows.

With respect to contour partition induced by five random points on the boundary in Fig 3.a.a.1. The five points in Fig 3.a.b.1 are selected with DCE.



Fig 3.a.a.1. Input image

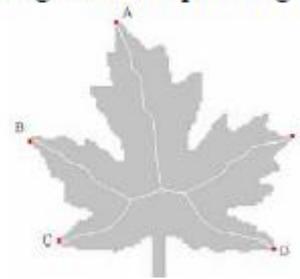


Fig 3.a.b.1. The five points in (b) are selected with DCE.

First, observe that every object boundary in a digital image can be represented without the loss of information as a finite polygon, due to finite image resolution. Let us assume that the vertices of this polygon result from sampling the boundary of the underlying continuous object with some sampling error. There then exists a subset of the sample points that lie on the boundary of the underlying continuous object (modulo some measurement accuracy). The number of such points depends on the standard deviation of the sampling error. The larger the sampling error, the smaller the number of points will lie on the boundary of the continuous object, and subsequently, the less accurately we can recover from the original boundary [31]. The question arises as to how to identify the points that lie on (or very close to) the boundary of the original object or equivalently how to identify the noisy points (that lie far away from the original boundary). The process of DCE is proven experimentally and theoretically to eliminate the noisy points [26], [27],[28]. This process eliminates such points by recursively removing polygon vertices with the smallest shape contribution (which are the most likely to result from noise). As a result of DCE, we obtain a subset of vertices that best represents the shape of a given contour. This subset can also be viewed as a partitioning of the original contour polygon into contour segments defined by consecutive vertices of the simplified polygon.

A hierarchical skeleton structure obtained by the proposed approach is illustrated in Fig. 6, where the (red) bounding

polygons represent the contours simplified by DCE. Because DCE can reduce the boundary noise without displacing the remaining boundary points, the accuracy of the skeleton position is guaranteed. A formal proof of DCE continuity with respect to the Hausdorff distance of polygonal curves is given in [29].

B).DCE evaluates global contour information in order to generate the simplified contour. The existing pruning method can be applied to any input skeleton. We only require that each skeleton point is the center of a maximal disk and that the boundary points tangent to the disk (generating points) are given.

B.Existing method Discrete Skeleton Evolution (DSE)

According to Blum’s definition of the medial axis [1], the skeleton S of a set D is the locus of the centers of maximal disks. A maximal disk of D is a closed disk contained in D that is interiorly tangent to the boundary ∂D and that is not contained in any other disk in D . Each maximal disk must be tangent to the boundary in at least two different points. With every skeleton point $s \in S$ we also store the radius $r(s)$ of its maximal disk. By Theorem 8.2 in [30], the skeleton S is a geometric graph, which means that S can be decomposed into a finite number of connected arcs, called skeleton branches, composed of points of degree two, and the branches meet at skeleton joints (or bifurcation points) that are points of degree three or higher.

Definition 1. The skeleton point having only one adjacent point is an endpoint (the skeleton endpoint); the skeleton point having more than two adjacent points is a junction point. If a skeleton point is not an endpoint or a junction point, it is called a connection point. (Here we assume the curves of the skeleton is one-pixel wide)

Definition 2. A skeleton *end branch* is part of the skeleton between a skeleton endpoint and the closest junction point. Let li ($i = 1, 2 \dots N$) be the endpoints of a skeleton S . For each endpoint li , $f(li)$ denotes the nearest junction point. Formally, an *end branch* P ($li, f(li)$) is the shortest skeleton path between li and $f(li)$.

For example, in Fig. 2, arc from 1 to a is a skeleton end branch: $P(1, f(1)) = P(1, a)$. The arc from a to b is not an end branch; it is a skeleton (inner) branch. Observe that point a is the nearest junction point of two endpoints (1 and 7). Based on Blum’s definition of a skeleton, a skeleton point s must be the center of a maximal disk/ball contained in the shape D .

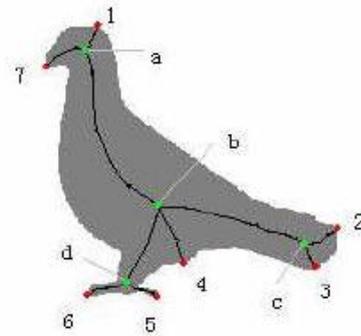


Fig 3.b.1 The endpoints (red) and junction points (green) on the skeleton

Definition 3. Let $r(s)$ denotes the radius of the maximal disk $B(s, r(s))$ centered at a skeleton point s . The reconstruction of a skeleton S is denoted $R(S)$ and given by

$$R(S) = \bigcup_{s \in S} B(s, r(s))$$



Fig 3.b.2



Fig 3.b.3



Fig 3.b.4

The reconstruction Fig 3.b. of the original skeleton Fig 3.b.1 is very close to the original shape in Fig 3.b.4

There are two motivations:

1) *Removing an end branch will not change the skeleton's topology; 2) the end branch with low contribution to the reconstruction is removed first. We define the **weight** w_i for each end branch $P(l_i, f(l_i))$ as:*

$$w_i = 1 - \frac{A(R(S - P(l_i, f(l_i))))}{A(R(S))}$$

Where function $A(\)$ is the area function. The intuition for skeleton pruning is that an end branch with a small weight w_i has a negligible influence on the reconstruction, since the area of the reconstruction without this branch is nearly the same as the area of the reconstruction with it. Therefore, it can be removed. The proposed skeleton pruning is based on iterative removal of end branches with the smallest weights until the desirable threshold is met.

IV. PROPOSED METHOD DISCRETE SKELETON RECONSTRUCTION (DSR).

A. Main Ideas of the Proposed Approach

Our proposed algorithm first splits the contour into number of contour segments[31].for each contour find the end branches using skeleton pruning algorithm.[32].For each end branch perform the confirmation test using the proposed merge vertex detection algorithm.

a) Core Skeleton Point (CSP)

For Core Skeleton Point CSP as in [33], check whether it is the seed point and is in the base of the skeleton [16] by finding the Global maximum of the Euclidean distance map for that contour.

If $ED(CSP) \leq \text{maximalDT}(D)$ then add this as a base CSP.otherwise move to next CSP.

b). Valance Skeleton Point (VSP)

For Valance Skeleton Point (VSP) as in [33,31]check whether the vertex is on the boundary by finding($_$) the Standard Deviation of the Sampling error of the contour position.The larger the sampling error,the smaller the number of boundary points.Hence If $STD(SAMERR(VSP) < \text{Theshold}(_)$.

c) Stop Condition.

Given a threshold T , we can stop DSR if $D_{avP(n,k)} > T$ for some k . Given a sequence of T values, we can obtain a hierarchical sequence of DSR simplified boundary polygons, which leads to a hierarchical sequence of corresponding skeletons. In general, an adequate stop condition depends on the particular application. A stop condition that is adequate for shape similarity is based on the difference of the DSR simplified contour to the original input contour. When the pruned skeletons are input into a

shape similarity measure, this stop condition is recommended.

To summarize, the vertices V_f that are used for contour partitioning induced by DSR are computed as: $V_f _ V_s _ (V_{\text{concave}} \cup V_i)$, where V_s denotes all the vertices of the simplified polygon P obtained by DSR, V_{concave} denotes all of the concave vertices of V_s and V_i denotes vertices of V_s withlow value of the measure D_i .Collect all CSP,VSP into the data store and sort them to find out the priority of the vertices.These vsp's are the computed merging points, that can be used for image processing operations so as to produce new images of shapes that satisfies the skeleton topology.These new images can be used in Engineering drawings,Robotics,Animations and related fields.

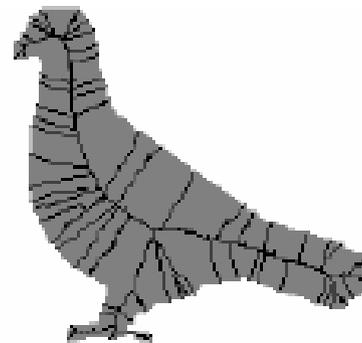
B. Methodology.

We present a new algorithm for computing the $p(l_i, f(l_i))$.This is an extended algorithm of skeleton pruning algorithm and skeleton growing algorithm. From the skeleton growing algorithm We find the end branches to be removed.In the proposed algorithm use that end branch and calculate its $(l_i, f(l_i))$.

a)The Six main tasks of the system are:

1.Prunned skeleton arc extraction.

- Calculate the end points and junction points for the threshold value.
- Calculate the number of skeleton paths between the end points.



t=0.0001

Fig 4.b.1

We are using the input image of $t=0.0001$ and set of pruned skeleton is selected and a table is constructed as described in the following steps.

2.Assign The Relevance Index Based on Euclidean distance

Assign the highest relevance index to the skeleton path based on Euclidean distance nearer to The Core of the contour partition and lowest relevance index to the skeleton path nearer to the boundary.

3. Relevance index table.

Use a table to store the Relevance indeces and the appropriate threshold value.

4.Sort this relevance indices.

Sort all the Relevance indices and select the highest relevance index skeleton arc. For each skeleton arc compute the two end points namely Valence skeleton point (VSP) and Core skeleton point (CSP).

Original shape with maximum skeleton arcs.



Fig 4.b.2

We have a new table with sorted relevance index skeleton arcs so that all the skeleton arcs are pruned and the needed skeleton arcs alone are stored as VSPsand CspS in the DSR Table.

5. Confirmation test for Valence skeleton point (VSP) and Core skeleton point (CSP).

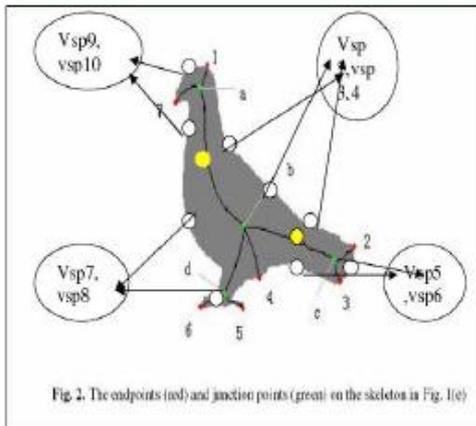


Fig 2. The endpoints (red) and junction points (green) on the skeleton in Fig. 1(e)

following connector symbol (yellow color) is used to denote the core skeleton point.



Compare The VSP and CSP with the initial image,so that the CSP may be one of its Junction Point Whose Removal may affect the Skeleton Topology. And also the VSP be on the Boundary and whose inclusion supports in the reduction of boundary deformation.

6.Create and confirm the data structure to store VSP AND CSP.

The new table with the set of VSP and CSP for the Particular Threshold value will definitely support the reconstruction of original image with remarkable output, having reduced boundary deformation.

INPUT IMAGE	THRESHOLD VALUE	CSP	VSP	RECONSTRUCTED IMAGE(DSR)
	T=0.01001	CSP1 CSP3 CSP4	VSP1 VSP2 VSP3 VSP4 VSP5 VSP6 VSP7 VSP8	

TABLE 4.c.6.2

Fig 4.c.6.1

The following connector symbol (yellow color) is used to denote the core skeleton point.

INPUT IMAGE	THRESHOLD VALUE	VSP	CSP	SKELETON ARC RELEVENCE INDEX
	T=0.0001	VSP1 VSP2 VSP8 VSP2 VSP8 VSP3 VSP6 VSP4 VSP5 VSP6 VSP7 VSP8	CSP1 CSP1 CSP1 CSP2 CSP2 CSP3 CSP3 CSP4 CSP4 CSP2 CSP2 CSP2	R11 R12 R13 R14 R15 R16 R17 R18 R19 R110 R111 R112

Fig 4.c.6.1

Fig 4.c.6.2

Computing CORE SKELETON POINT (CSP) i.e (li) we extended the skeleton growing algorithm in [16] based on the Euclidean distance map. First, we selected a skeleton seed point as a global maximum of the Euclidean distance map. Then, the remainder of the skeleton points is decided by a growing scheme. In this scheme, the new skeleton points are added using a simple test that examines their eight connected points. During this process, the redundant skeleton branches are eliminated by the DCE.

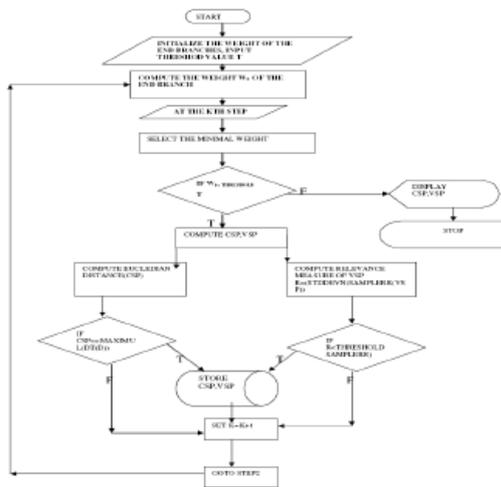
Computing VALANCE SKELETON POINT VSP i.e f (li). Secondly, we extended the skeleton pruning algorithm. The VSP is taken and the standard deviation of the sample error of the particular polygon is calculated.

b) Merging Vertx Detection Algorithm.

Merging Vertex Detection Algorithm:

- Step 1. For each skeleton arc, we compute the weight.
- Step 2. Select the minimal weight $W_i^{(k)}$
 - if the weight is less than the threshold, then goto step 3.
 - Else
- Step 5. Sort the Data structure and select the csp, vsp of the pruned branches whose Value is less than the threshold value and the sampling threshold respectively.
- Step 6. These are the list of points needed (Merging Points) for further researches. Stop and output the final skeleton.
- Step 3.
 - 1) Compute the $CSP(h), VSP(f(h))$ of $P_{min}^{(k)}$
 - 2) Compute Euclidean Distance Of CSP
 - If $ED(csp) = \text{MAXIMAL } DT(D)$ THEN goto step 4a.
 - Else Goto step 4.
 - 3) Compute the relevance measure $R = SD[SE[vsp]]$
 - If $R < \tau$ (theta for sampling threshold) THEN goto step 4a.
 - Else Goto step 4.
 - 4a) Append this vsp and csp into the vertex subset of the contour and store it in the Datastructure.
- Step 4. set $k=k+1$ go to step 2.

c) Flowchart of MergeVertex Detection Algorithm



Every evolutionary step, a pair of consecutive line segments s_1, s_2 is replaced by a single line segment joining the endpoints of S_1 and S_2 .

The key property of this evolution is the order of the substitution. The substitution is achieved according to a relevance measure K given by:

$$K(s_1, s_2) = \frac{\beta(s_1, s_2)l(s_1)l(s_2)}{l(s_1) + l(s_2)}$$

Where line segments s_1, s_2 are the polygon sides incident to a vertex v , $\beta(s_1, s_2)$ is the turn angle at the common vertex of segments s_1, s_2 , l is the length function normalized with

respect to the total length of a polygonal curve C . The main property of this relevance measurement is [16][18]:

C. Addinh a Skeleton Arc from a Distance Transform

The main goal of this section is to show that it is not necessary to have a separate postprocessing step in skeleton pruning, as we can grow a pruned skeleton directly form the distance transform. In this section, we work in the discrete domain of 2D digital images, in which the object contour is still represented with polygons. To achieve our goal we extend the fast skeletongrowing algorithm presented by Choi et al. [16]. We briefly review the skeleton growing algorithm in [16].

First, the Euclidean Distance Transform DT of the binary image of a given shape D is computed. Then the point with the maximal value of $DT(D)$ is selected as a seed skeleton point. Finally, the skeleton is grown recursively by adding points that satisfy a certain criterion, which intuitively means that the added points lie on ridges of the $DT(D)$. The grow process is based on examining every eight-connected point of the current skeleton points.

The skeleton continues growing in this way until it reaches an endpoint of a skeleton branch. Next, other skeleton branches starting at other skeleton points are considered. The proposed extension of the Skeleton Pruning Algorithm is very simple. For a Skeleton arc to be added, it must additionally have its csp point on the base and vsp on the boundary of the same contour segments of a given contour partition.

V. PERFORMANCE EVALUATION

The performance of skeleton matching depends directly on the property of shape representation. We show the performance evaluation of the proposed work in four parts. 1. Stability. 2. Analysis and Comparison. 3. The potential in shape similarity. 4. Time complexity.

The DSR Data structure (Table) is very much useful in reconstructing the image (with reduced boundary deformation, which is a major problem in skeleton topology). This Table can be used as a back support and the VSPs and CSPs are first used to reconstruct the shape of an output image. The input image and the DSR table Together will give you a new way of Reconstructing the



Fig 5.1 Output image.

Here in the input of the existing system, there are so many skeleton arcs which consume a lot of storage space and it reduces the speed. Also the Output image does not guarantee the exact matching of the input image, due to boundary deformation. But Our Proposed system with the DSR table comparatively less storage and produces output image with reduces boundary deformation guaranteeing the 100 % matching of the input image. Also the same technique can be used for different set of threshold values and revised DSR table can be used as a database for shape reconstruction with minimal boundary deformation.

VI. CONCLUSION AND FUTURE SCOPE.

Skeletons are useful in the area of Handwritten and printed characters, Fingerprint patterns, Chromosomes & biological cell structures, Circuit diagrams, engineering drawings. Our proposed method with dsr table having CSP, VSP values produces output images with minimal boundary deformation guaranteeing the 100% matching of the input image. also the Vsp can be used for merging between two different shapes hereby generating new shapes (objects) satisfying all the skeleton properties.

This VSP will surely form a new platform for engineer to design a innovative machines like robots for scientific developments. Thus our aim of identifying the merging points in an image using skeletons can be achieved.

VII. ACKNOWLEDGEMENTS

The author would like to thank the almighty and her parents. The author is grateful to Mrs M.Punithavalli, Director, Department of computer applications, Sri Ramakrishna College of arts and sciences for women, Coimbatore for her constant encouragement. She also truly thankful to the Principal and secretary Mr.V.Sengodan, SNR SONS College, Coimbatore for his blessings and the support.

REFERENCES

- [1] H. Blum, "Biological Shape and Visual Science (Part I)," *J. Theoretical Biology*, vol. 38, pp. 205-287, 1973.
- [2] K. Siddiqi, A. Shkufandeh, S. Dickinson, and S. Zucker, "Shock Graphs and Shape Matching," *Proc. Int'l Conf. Computer Vision*, pp. 222-229, 1998.
- [3] C. Di Ruberto, "Recognition of Shapes by Attributed Skeletal Graphs," *Pattern Recognition*, vol. 37, pp. 21-31, 2004.

[4] S.C. Zhu and A. Yuille, "FORMS: A Flexible Object Recognition and Modeling System," *Proc. Int'l Conf. Computer Vision*, 1995.

[5] T. Liu, D. Geiger, and R.V. Kohn, "Representation and Self- Similarity of Shapes," *Proc. Int'l Conf. computer Vision*, Jan. 1998.

[6] C. Aslan and S. Tari, "An Axis Based Representation for Recognition," *Proc. Int'l Conf. Computer Vision*, 2005.

[7] T.B. Sebastian, P.N. Klein, and B.B. Kimia, "Recognition of Shapes by Editing Their Shock Graphs," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 26, no. 5, pp. 550-571, May 2004.

[8] J.Komala Lakshmi and M.Punitha Valli: A Survey on skeletons in digital image processing ,in the international conference proceedings of IEEE Computer Society ,2009:260-269.

[9] J.Komala Lakshmi and M.Punitha Valli: A Survey on skeletonization in digital image processing ,in the international conference proceedings of Managing Next Generations Software Applications08, Sponsored by CSIR NewDelhi, 2008:825-839.

[10] Mayya, N., Rajan, V.T.: Voronoi Diagrams of polygons: A framework for Shape Representation. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 638-643 (1994)

[11]. Ge, Y., Fitzpatrick, J.M.: On the Generation of Skeletons from Discrete Euclidean Distance Maps. *IEEE Trans. Pattern Analysis and Machine Intell.* 18(11), 1055-1066 (1996)

[12]. Gold, C.M., Thibault, D., Liu, Z.: Map Generalization by Skeleton Retraction. In: *ICA Workshop on Map Generalization*, Ottawa (August 1999).

Separating Words from Continuous Bangla Speech

Nipa Chowdhury,
Dept. of CSE,
Dhaka University of Engineering
& Technology (DUET),
Dhaka ,Bangladesh,
nipa83@yahoo.com

Md. Abdus Sattar,
Dept. of CSE,
Bangladesh University of
Engineering & Technology
Dhaka ,Bangladesh
masattar@ cse.buet.ac.bd

Arup Kanti Bishwas
Dept of Electrical,
Eastern Refinery Limited (ERL)
Chittagong, Bangladesh,
arupeeabd@yahoo.com

Abstract— In this paper we present a new word separation algorithm for Real Time Speech i.e., Continuous Bangla Speech Recognition (CBSR). Prosody has great impact on Bangla speech and the algorithm is developed by considering prosodic feature with energy. Task of this algorithm is to separate Bangla speech into words. At first continuous Bangla speech are fed into the system and the word separation algorithm separate speech into isolate words. Performance of the proposed algorithm is compared to the existing algorithms and result shows that 98% word boundaries are correctly detect.

Keywords:

Pitch, Stress, Word Separation, Continuous Bangla Speech

I. INTRODUCTION

The advent of Graphical User Interface (GUI) and others, gap between the human and computer is reduced and so Human-Computer Interaction (HCI) has gained importance in the age of information technology. Mouse, Keyboard, Touch Screen, Pens are serve to make HCI easy but uses of these devices is in hand. Speech is the primary mode of communication among human being and people expect to exchange natural dialect with computer due to recent development of speech technology. So in future, more development will happen in the field of speech in HCI.

Speech recognition is the process of extracting necessary information from input speech signal to make correct decision Speech recognition can be classified as speaker dependent or independent, isolated or Continuous/Real time and can be for large vocabulary or small vocabulary. An isolated speech recognition system requires that a speaker offer clear signature between words whereas continuous speech consists of continuous utterance which represents our natural speech. Isolated speech recognition is much easier than continuous speech recognition because start and end point determination of a word is easier because of clear pause or silence.

In this paper we investigate problems of existing algorithms and propose a new method for word detection in real time Bangla speech by using prosodic feature. Result shows that not all prosodic parameters convey useful information for

word separation and success rate is increased by 6% compare to other algorithms [5].

II. RELATED WORKS

Although 8% of total population of the word speaks in Bangla, works on Bangla speech recognition is not satisfactory [1]. By using Reflector coefficient, autocorrelations as speech feature and Euclidian distance for taking decision, recognition of vowel [2] was done with 80% efficiency. 66% recognition accuracy was obtained for Bengali phoneme recognition [3] where rms value used as feature and ANN as classifier. In [4] experiment was carried out for a database consisting of 30 different Bangla words. A word separation algorithm had been developed for Continuous speech Recognition [5] by comparing noise energy and zero crossing with speech and for 13 words. Among them 1 word is not separated and it requires huge memory, training time.

Crucial point is that continuous speech does not offer any clear signature like pause, silence between words. In CSR word boundary detection is important otherwise task of speech recognizer is more complicated. Because a new word may begin at any time and require huge search space [6]. But if words can be correctly detected than strategy of isolated speech recognition can be applied for continuous speech recognition.

Existing word separation algorithm was employed to separate words from continuous Bangla speech by comparing noise's energy and zero crossing rates to speech. We implement the existing algorithm and performance is shown below. In Figure 1 at first section input speech 'Artho Anorther Mul' is shown and in second section speech is segmented using existing algorithm. This shows that 'Artho' ('Ar,tho) and Anorther (Anor, ther) is not separated correctly.

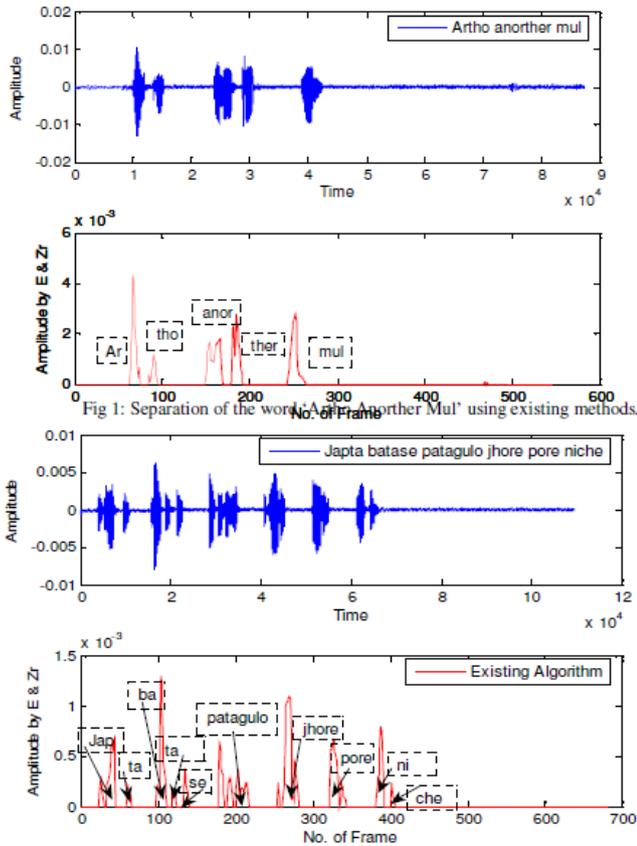


Fig 1: Separation of the word 'Artho Another Mul' using existing methods.

Fig 2: Separation of the word 'Japta batase patagulo jhore pore niche' using existing methods

Figure 2 shows that 'Japta' is separated incorrectly (Jap, ta), Batase as (ba, ta, se), Niche (ni, che). So silence may exist within a word. Considering only word length with energy, zero crossing rates do not give good result.

III. OUR APPROACH

Bangla is a bound stress language unlike English [6, 9]. It gives prominence on word that is high on initial word and become low at end of the sentences. So, our idea is if this prominence can be detected then words can be separated. To achieve this following methodology is set out:

1. Speech pattern is detected by comparing speech's energy to noise energy.
2. Stress consists of pitch, intensity, duration [7]. Pitch means dominant frequency, intensity means power and duration means time duration. To find fundamental frequency of speech autocorrelation methods is used.
3. By using above stress information a Word Separation algorithm is developed.

IV. DETAILS OF EXPERIMENT

Conversion of analog speech signal into digital is the first step of speech signal processing. For this reason speech signal is digitized with sampling frequency 11025Hz and sampling size is 16 bit per sample. Human speech production is dynamic and nonstationary. But within a short

period of time (20-40ms) its behavior is quasi stationary. Speech is framed of 40 ms length with 20ms overlapping. Then windowing is performed to remove unnatural discontinuities in the speech signal. Hamming window is used in this experiment.

Speech is easily affected by noise. Noise removal of speech is the most challenging because noise nature is random. For this reason energy from first 100ms speech data considered as noise energy and then threshold value ($E_{noiseth}$) is set. Now energy for other speech frame, E_{sn} is calculated according to the following formula.

$$E_{sn} = \sum_{m=1}^N s_m^2$$

Here, E_{sn} is the energy of nth frame; S_m is speech data and N is total number of samples in nth frame. Now if E_{sn} of speech frame is greater than noise energy threshold then speech frame exist and next it consider for stress analysis so consecutive frames are tagged as candidate word. Other frames are considered as noise frame hence discard.

A. Stress information:

Stress consists of pitch, intensity and duration. Intensity can be derived by using

$$p = \sum_{i=1}^k |s(i)| \tag{1}$$

Duration is simple time duration of candidate word. Pitch is the most useful information for word separation of Bangla speech. But the general problem of fundamental frequency or pitch estimation is to take a portion of signal and to find dominant frequency of repetition [12]. Because fundamental frequency of the periodic signal may be changing over the time and not all signals are periodic. Signals that are periodic with interval T are also periodic with interval $2T$, $3T$ etc, so we need to find the smallest periodic interval or highest fundamental frequency.

Different methods have been used for pitch detection like Zero Crossing, LPC based, Autocorrelation, Cepstrum, Harmonic structure etc [11]. Out of them autocorrelation methods is still one of the most robust and reliable pitch detectors. The autocorrelation computation is made directly on the waveform and fairly straight forward [10].

Autocorrelation function finds the similarity between the signal and shifted version of itself.

$$y(n) = \sum_{m=1}^N s(m)s(m-\alpha) \tag{2}$$

This finds peaks in fundamental frequency and multiple of fundamental frequencies if the speech signal is harmonic. Actually as the shift value α begins to reach the fundamental period of the signal, the autocorrelation between the shifted

signal and original signal begin to increase and rapidly approaches to peak at the fundamental period. But length of frame is important for autocorrelation [10]. So we have to select frame size as long enough to contain at least one period and detect peak clearly. Hence we use short time autocorrelation function with frame size 40ms which exactly matched to our speech frame length. In Figure 3 resultant value of autocorrelation and frequency that derived by autocorrelation is shown.

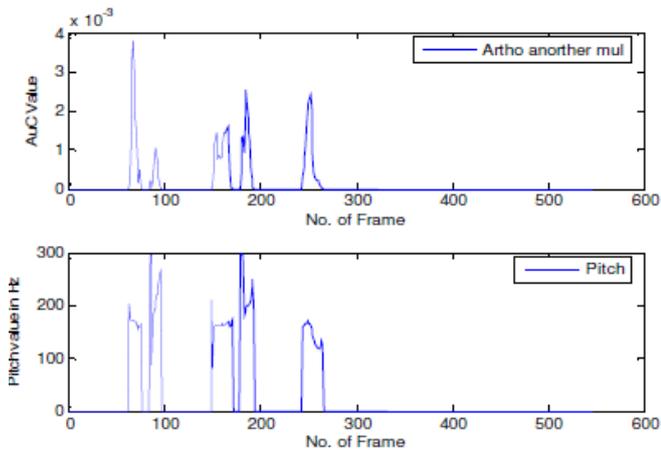


Fig 3: Shows frequency of speech 'Artho Another Mul' by autocorrelation method

Fundamental frequency of male voice is 85-155Hz and female voice is 165-255 Hz. So we estimate fundamental frequency in between 60-300Hz. By using this pitch information we merge candidate words into one word. If pitch information of consecutive candidate words are high within certain gap then those candidate word form a single word. For example, Pitch information of 'Ar' and 'tho' is 170,221 so candidate word 'tho' is merged with 'Ar' and formed 'Artho'. Again Pitch information of 'Ano' 'ther' is 163,230 so formed one word as 'Another'.

V. RESULTS AND DISCUSSIONS

Speech signal are recorded in room environment using microphone. Sampling frequency of 11025Hz and sampling size bits per sample and mono channel is set. Total 400 sentences are uttered by speaker. Last 200 sentences are the repetition of first 200 sentences but word position is varied. For example in first 200, if a sentence forms like this 'Amader deshe onek nodi' then in second 200(201-400), it forms like 'Onek nodi amader deshe'. This is done to verify that our algorithm is error prone to specific position of word or not. So, total 400 sentences form our database. After separation of words from speech each word is saved in a wave file to take decisions. Word Separation of several speeches is shown using existing algorithm and proposed algorithm in Table 1.

Experimental result shows that our proposed algorithm that uses stress information with energy performs excellent compare to existing algorithm. Other stress information like

intensity and duration does not contain useful information for word separation of Bangla speech. 798 words of total 2293 words from 400 sentences are not separated by existing algorithm and the number is 64 in our new algorithm. Success rate is 97% which is 6% better than existing algorithms [5] and also the rate is higher than the previous rate of 81% reported in [6].

VI. CONCLUSIONS:

English is said to 'stress-timed', as opposed Bangla is said to be bound stress. Prosodic behavior is vary from Language to Language (English [7], Hungarian, Finnish [8], Bengali [6, 9] etc). Stress in Bangla is high at initially and become low at the end of speech. We try to correlating this stress information with word. Result shows that energy and pitch information is important parameter for word separation. We use energy, zero crossing rate and pitch to separate words and get same results. Further study can be carried out for large database. Extracting Bangla phoneme from continuous bangla speech can also be challenging task.

REFERENCES:

- [1] Mottalib M. A, "A Review on Computer based Bangla Processing", Proc. of the National Conf. on Comp. processing of Bangla (NCCPB), Independent University, Dhaka Bangladesh, 27 February, 2004, pp.72-81.
- [2] Karim, A. H. M. Rezaul, Rahman M. S and Iqbal M. Z., "Recognition of Spoken Letters in Bangla", Proc. of the 5th Int. Conf. on Comp. and Info. Tech.(ICCIT), East West University, Dhaka, Bangladesh, 27-28 December, 2002, pp. 213-216.
- [3] Hassan M. R., Nath B. and Bhuiyan M. A., "Bangla phoneme recognition- A New Approach.", Proc. of the 6th Int. Conf. on Comp. and Info. Tech. (ICCIT), Jahangirnagar University, Dhaka, Bangladesh, 19-21 December 2003, pp. 365-369
- [4] Islam M. R., Sohail A. S. M., Sadid M. W. H. and M. M. A., "Bangla Speech Recognition using three layer Back-propagation neural network", Proc. of National Conf. on Comp. processing of Bangla,(NCCPB),Independent University, Dhaka Bangladesh, 27 February, 2004, pp.44-48.
- [5] Rahman K. J., Hossain M. A., Das D., Islam A. Z. M. T. and Ali Dr. M. G, "Continuous Bangla Speech Recognition System", Proc. of the 6th Int. Conf. on Comp. and Info. Tech. (ICCIT), Jahangirnagar University, Dhaka, Bangladesh, 19-21 December 2003, pp. 303-307.
- [6] Mandal S. K. D., A. K. Datta, Gupta B., "Word Boundary Detection of Continuous Speech Signal for Standard Colloquial Bengali (SCB) Using Supra segmental Features", The Int. Symposium on Frontiers of Research on

Speech and Music (FRSM), IIT, Kanpur, India. 15-16 February, 2003.

[7] Imoto K., Dantsujiy M., Kawahara T., "Modeling of the Perception of English Sentences Stress for Computer-Assisted Language Learning", 6th Int. Conf. on Spoken Language Processing (ICSLP), Beijing, China, 16-20 October, 2000, Vol-3, pp 175-178.

[8] Vicsi K. and Szaszák G., "Automatic Segmentation of Continuous Speech on Word Level based on Supra-

segmental features", Int. Journal of Speech Tech., Netherlands, December, 2005, Vol-8, pp 363-370.

[9] www.en.wikipedia.org/wiki/Bangla_language

[10] Rabiner L. R., "On the use of Autocorrelation Analysis for pitch detection", IEEE Transactions on Acoustics, Speech and Signal Processing, February 1977, Vol. ASSP-25, No-1.

[11] http://en.wikipedia.org/wiki/Pitch_detection_algorithm.

A Survey on User Interface Defect Detection in Object Oriented Design

VIJAYAKUMAR ELANG OVAN

Abstract- The interruptions and errors are frequent occurrence for the user having obscurity with a user interface. These delays and errors can result in brutal problems, predominantly for the real time and mission-critical applications in which the speed and accuracy are of the fundamental nature of the software. The difficulty of the users is often occurred by interface-design defects that may confuse or mislead the users. The current methodologies for separating such kind of defects are highly time consuming and very expensive this is so as they require human analysts to identify the defects manually where the users experience the difficulty. These methods only then can analysis and mend of the defects take place to the user. The tribulations of the complex human-machine interactions remain a challenge that is becoming very serious and the resources to get enhancement in their dependability and security needs to be identified and integrated. Ambiguous labels on interface controls, incomprehensible instructions, confusing icons, and inadequate system-status feedback are some examples. These are the reasons for causing delays and errors in user's improvement in the direction of target achievement.

I. INTRODUCTION

The most frequently used Object-Oriented models, may possibly be affected by a variety of defects introduced simply due to misunderstanding of modeled reality and incorrect assumptions. The defects must be recognized and corrected as early as possible, rather before the model is used as the base for the later representations of the system. Improved study methods and efficient tool support are required to assure the effectiveness of the defect detection process.

The user interface is a vital part of most software so the quality of its progress is of decisive importance. The criteria of a software quality with the user interface have usability. The user estimates the application program based on its user interface flexibility [1, 2, 3]. Estimating the usability is an exclusive task in terms of time and labor which would cost high [18]. This is the main problem is usually resolved by gradually rising the number of testers or by automation of the process. The main task of this component is to detect defects of usability in a user interface based on its model and to give advice on their elimination.

Quality of Object-Oriented designs is put down by design defects resulting from poor design. Consequently, there is a chance for perfection is required. On the other hand, design defects are not accurately specified and there available, a small number of apt techniques allocate their detection and correction. This research focuses the problem with user interface defects in object oriented software metrics which has influence on the quality of the software, creating a metrics tool based on object oriented software. These metrics are proposed to add more quality in refining any object oriented software during the different stages.

II. DEFECTS IN OBJECT-ORIENTED DESIGN

1) Design defects

Design defects and design patterns are identical used and studied in the business and education [8]. Excellent solutions are given by design patterns to persistent problems in object-oriented design. Design defects cause fault in the software design due to the absence or the misuse of design patterns. Thus, Gu eheneuc et al. define design defects as distorted forms of design patterns, i.e., micro-architectures alike but not equivalent to those proposed by design patterns [12].

2) Code smells

Code smells means the code level warning sign or problem. Beck and Fowler describe code smells as "certain structures in code that suggest the possibility of refactoring" [9]. Some examples are duplicated code, long method, large class, long parameter list, data class. Possible presence of an anti pattern and of a design defect can be recommended by the presence of code smells. Code smells are usually linked to the inner workings of classes where as design defects comprise the relationships among classes which are much situated on a micro- architectural level. So code smells can be called as intra-class defects and design defects as inter-class defects.

III. USER DIFFICULTY

The user difficulty is abstractly, an internal cognitive state in which the capacity to achieve a goal is impaired. Since the user difficulty is not openly observable from a usability test data, it must be contingent from the events that are directly observable in data, such as audio, video and think-aloud and other audio recordings.

1) Criteria for User difficulty

There are many reasons for the difficulties occurred when the user interfaces with a product [24]. The major criteria are the following:

Statements of Users: This occurs when a user tries to communicate or understand exceptions like makes a statement or asks a question indicating confusion, frustration, uncertainty, lack of knowledge, or indecision.

Inactivity of the user: This occurs as the user is not active with both the mouse and keyboard and does not react for a long time that genuinely indicating confusion in the data used in this study of the user.

Toggling: These are occurred when the user toggles an interface control, such as a checkbox, or a dialog box which confuses the user through one or more full cycles of its various states, without any intervening actions. These toggling generally indicate that a user is confused about what the control does or about how to operate it.

Help: This occurs when a user consults an online Help file or the help documentation in the product or moves to the service centre for the help about the product. Though, the difficulty is measured to have started at a period earlier to consulting Help. The statement is that users happen to get confused in the stage before consulting some sort of Help, so the hit it off that brings up the Help window would be too late to be measured as the onset of the period of difficulty.

Accuracy: It is measured in terms of hit ratio and false alarm ratio proportion. The Hit rate is the percentage of all periods of genuine user difficulty that the detector detects, while false alarm ration is the percentage of the rate of events for which the detector incorrectly indicates user difficulty when none was present.

IV. USER INTERFACE DEFECTS IN OBJECT ORIENTED DESIGN

User Interface Defect

The User Interface detection is the set of methods where an evaluator inspects a user interface. This is in usability of the interface evaluated by testing it on real users. The usability interfaces can be used in the development process by evaluating or specifications for the user interface [6]. User Interface detection methods are generally considered to be cheaper to implement than testing on users.

User Interface (UI) is intended into an information device with which a person can interact together with display screen, keyboard, mouse, illuminated characters, help messages, interaction of website and its response.

The User Interface detection method includes the Cognitive walkthrough, Heuristic evaluation and Pluralistic walkthrough [19, 20]. The User Interface is full of means by which the users interact with the system a particular machine, device, computer program or other complex tool. The user interface provides means of input, to manipulate a system and the output, to indicate the effects of the manipulation of the user. The user interface can perhaps

includes the total ease of the users which may include the aesthetic appearance of the device, response time, and the content that is presented to the user within the context of the user interface.

Major reasons for the defects

There are several defects that occur in user interface for any product in object oriented methodology. Some of the major causes are:

1) *Omission*

The necessary information about the system has been omitted or not clearly given from the software artifact.

2) *Incorrect Fact*

Some of the information in the software artifact contradicts with the information in the requirements document or the general domain knowledge for the usage of the software.

3) *Inconsistency*

The information within one part of the software artifact is inconsistent with other information in the software artifact and such types of user design could also lead to defect.

4) *Ambiguity*

The information within the software artifact is ambiguous. That is any of a number of interpretations may be derived that should not be the prerogative of the developer doing the implementation.

5) *Extraneous Information*

Information is provided that is not needed or used can also confuse the user and lead to defects.

Rectification Methodology for User Interface Defects

User interface [5] defect in Object-Oriented Design is a high visibility defect and simple to rectify. The procedure follows as below,

- Defect should be reproduced
- Defect screen shots are captured
- Document correct inputs that are used to obtain the defect in the defect report

First of all computer hard ware configuration should be checked whether it is the same as that of the developer's system configuration and also make sure that the system's graphic drivers are installed appropriately. If the problem is found in the graphic drivers, the user interface error will occur [12]. So first check if it is correct from user's side then report the defect by following the above method.

V. USER INTERFACE TESTING

The testing of the user interface (UI) is to ensure that it follows accepted UI standards and meets the requirements defined for it. UI Testing can be classified into four stages – Low level, Application, Integration, Non-functional.

- **Low Level Testing:** The low level consists of the Navigation and checklist testing
- **Application Testing:** The Application testing consists of Equivalence Partitioning, Boundary Values, Decision Tables and State Transition Testing.

- **Integration Testing:** The Integration testing consists of the Desktop Integration, C/S Communications and Synchronization
- **Non-Functional Testing:** The non functional testing consists of the Soak testing, Compatibility testing and Platform/environment.

1) *Defect testing*

The goal of defect testing is to discover defects in programs a successful defect test is a test that is caused by a program to behave in an anomalous way Tests show the presence and not the absence of defects [6]. To discover the faults or defects in the software that is to find where its behavior is incorrect or not in conformance with its specification. A successful test is a test that makes the system perform incorrectly and so exposes a defect in the system.

Manual Testing

The manual testing includes the various stages of finding the defects of the user interface faults. The proficiency and flexibility to provide manual (Black-Box) software application testing services for detection of the user interface errors can be done through the following stages Functional testing, Compatibility testing, Performance testing, Regression testing, Unit testing and the Conformance testing. The main testing technique that is used for the detection of the user interface faults is the functional testing.

Functional testing

The functional testing or the user interface testing is testing of the application's user interface that describes how the application and the user interrelate and if the application performs properly all its functions to respond to the user. This normally includes all responses including how the application handles the input from the keyboard and mouse and how it replies these responses to the displays the screen. The functional testing is frequently done by human testers but this is now made a lot efficient and more reliable by usage of automated testing tools which offers many other additional features that extend the productivity of automated functional testing of the user interface defect for any product.

VI. TECHNIQUES USED FOR DETECTION

There are various methodologies and techniques used for the detection of user interface defects the most common methodologies and the techniques and their specifications are discussed below.

1) *Software Inspection Technique*

G. Cockton et al., discussed the one of the most well identified software quality techniques is the software inspection. Software inspection is currently far and wide accepted as an effectual technique for defect detection. A software inspection refers a group meeting carried out to expose defects in software products, occur during requirements specification, user interface design, and code and test plan. Software inspection approach is a pre-planned process relating a sequence of well defined inspection steps

and roles, formal collection of process and product data and a checklist to aid error detection. Software inspections are conducted in business because of their effective way to expose defects. Rate of defect detection for an inspection differs on the basis of the product type being inspected and also based on the particular inspection used [3].

2) *Checklist-based and Perspective-based reading Techniques*

Giedre Sabaliauskaite et al., projected that Checklist-based reading (CBR) and Perspective-based reading (PER), are the two reading techniques most commonly used. The result was that both techniques are alike in order to detect defect effectively through individual inspection in inspection meetings. Considering the effectiveness of inspection meetings, we found out that the teams that used CBR technique exhibited significantly smaller meeting gains (number of new defect first found during team meeting) than meeting losses (number of defects first identified by an individual but never included into defect list by a team); meanwhile the meeting gains were similar to meeting losses of the teams that used PER technique [22]. Consequently, CBR 3-person team meetings turned out to be less beneficial than PER 3-person team meetings.

3) *Ontology Models*

Gribova V et al., has described the principal necessities to a system of automated detection of usability defects are expandability of the system, informing the developer about defects, and giving advice on its elimination of these defects. The chief initiative of this conception is to form an interface model using ontology models which portray features of every component of the model. Based on this high-level specification the code is generated for the user interface. The main components of the interface model are a domain model, a presentation model, a model of linking to an application program and a model of a dialog scenario. All the component of the interface model is formed by a structural or graphical editor managed by a domain-independent ontology model [17].

The principal requirements to a system of automated detection of usability defects are expandability of the system, informing the developer about defects, and giving advice on its elimination. The author has described a conception of user interface development based on ontologies in [2]. The main idea of this conception is to form an interface model using universal ontology models which describe features of every component of the model and then, based on this high-level specification, generate a code of the user interface. Components of the interface model are a domain model, a presentation model, a model of linking to an application program and a model of a dialog scenario. Every component of the interface model is formed by a structural or graphical editor managed by a domain-independent ontology model.

4) *Metristation Technique*

Metristation is a new transferable user-interface evaluation system as discussed by Ivory M.Y. et al., [16] that runs on an IBM-compatible desktop or notebooks and computer. Metristation monitors user interactions and their usage, automatically capturing detailed data about user-system behavior including keystrokes, mouse clicks, mouse movements, video display context, user speech observer speech and critical incidents. This is used for automatic location of potential user-interface defects. The data can be analyzed and presented in as much as three orders of magnitude less time than is conventionally possible; the uniformity of the data and analysis provide increased repeatability and accuracy over traditional evaluation methods.

The user interface dependability can only be achieved if the interface defects are detected, diagnosed, and recovered from. Nevertheless, merely detecting user-interface defects has confirmed to be a very difficult problem. A variety of techniques, including inspection-based methods user modeling user opinion surveys, field observation of users at work, and laboratory user testing have been studied by researchers and used in practice. However, all of these methods have significant weaknesses, including both failure to detect defects and classification of non-problematic aspects of interfaces as defects [14]. Of all available methods, observation-based techniques, i.e., field observation and laboratory user testing, are generally accepted as the best for acquiring valid results.

5) *Hesitation Analysis Technique*

R.W. Maxion et al., had proposed Hesitation detection refers automatically detect the instances of user difficulty. Thus the instances of user difficulty is defined as an instance in which a user's ability to achieve a goal is impaired. Although such hesitations can occur for many reasons, they often indicate user difficulty. Hesitation detection accuracy can be characterized by two measures: the percentage of all user difficulties that it detects and by the percentage of benevolent events that it mistakenly classifies as difficulties assuming it does have the ability to detect instances of user difficulty accurately [15]. Hesitation detection provides several momentous enhancements to observation by a human usability analyst alone. They are:

It is cheap since hesitation detection is automated, it can save human-analyst time. It also provides better coverage much more data can be searched for instances of user difficulty [13]. It is invulnerable to human error; it does not miss instances of user difficulty due to limited attention.

The hesitation analysis does not detect the user interface defects directly it detects periods of user complexity that are the likely effect of interface defects. After the hesitation-detector output has been obtained, a usability analyst must examine other sources of data, usually video and audio of

user sessions, to determine which hesitations really indicate difficulty, as well as which particular defects caused each period of difficulty.

This methodology can be applied to both field and lab-based user studies to save time that a usability analyst would otherwise have spent combing the data for trouble spots. The results show that hesitations are an effective means for detecting instances of user difficulty, and that hesitation detection promises to make usability studies less expensive and more comprehensive. For example, up to 96% of an analyst's wasted time can be saved by using hesitation detection, while still detecting 81% of all defects manifested in usability data.

6) *Orthogonal Defect Classification Technique*

Gribova V et al., had described the methodology that automates static analysis that is effective at identifying and checking faults in user interface design. The majority of the defects found by automated static analysis appear to be produced by a few key types of programmer errors. Some of these types have the potential to cause security vulnerabilities. Statistical analysis results indicate the number of automated static analysis faults can be effective for identifying problem modules. Static analysis tools are complementary to other fault-detection techniques for the economic production of a high-quality software product.

7) *Refactoring Detection Technique*

According to Filip Van Rysselberghe, Refactoring, this technique is the process of altering a software system in a way such that it does not change the external behavior of the code however improves its internal structure. It is a regimented way to clean up code that minimizes the chances of introducing bugs or errors. In concentrate when refactor the design of the code is improved after it has been written. While refactoring, the balance of work changes. The design, rather than occurring all up front, occurs continuously during development. The resulting interaction leads to a program with a design that stays good as development continues.

- **OptimalAdvisor:** The tool enables the developers to refactor their code automatically in the code analysis. Optimal Advisor supports class, package rename, move, remove unused import, and the dependency inversion refactoring.
- **IBM Structural Analysis for Java** does not refactor the code automatically but gives an exhaustive map of dependencies for assistance in refactoring.
- **The Refactory Browser** supports also like Optimal Advisor for class, variable and method insert, move, and remove refactoring [14].

The present survey relates to defect detection or the observation method that detects fine defects in the course of defect inspection and observation, does not detect locations not constituting defects, or classifies a defect candidate as a

grain phenomenon or other phenomenon that does not affect a product. In one embodiment, a method for inspecting defects of a product having a plurality of product units formed repetitively at different locations comprises obtaining an image of the product units on the product having an appearance to be observed [10]. detecting regions of the image each having an appearance which differs from an expected appearance by greater than a preset threshold; calculating feature amounts for the detected regions; classifying the detected regions into groups of defect candidates; forming an aggregate of the feature amounts of the detected regions in the different product units, for each of the groups of defect candidates; and determining for each product unit attributes for the detected regions by comparing the feature amounts of the detected regions belonging to each group of defect candidates with a distribution of the aggregate of the feature amounts for the group of defect candidates.

8) *Reading Techniques*

Victor R. Basili et al., proposed reading is a key technical activity for analyzing and constructing software artifacts. It is critical for reviews, maintenance, and reuse. It is a concrete set of instructions given to the reader saying how to read and how to look for in a software product and the individual analysis of a software artifact. Specifically to achieve the understanding needed for a particular task e.g., defect detection, reuse, maintenance [21].

“Software reading techniques” try to enhance the efficiency of inspections by giving procedural rule that has to be followed by individual commentators to read a software artifact which has been given and spot defects [4]. There is experiential proof that software reading technique is a hopeful technique to increase the efficacy of inspections on special sorts of software artifacts, not with some degree of source code.

VII. OBJECT-ORIENTED SPECIFIC METRICS

Many different metrics have been proposed for object-oriented systems. The selected object-oriented metrics are primarily applied to the concepts of classes, coupling, and inheritance [11]. In some cases, the counting method for a metric is determined by the software analysis package being used to collect the metrics [7 and 2].

1) *Weighted Methods per Class*

The WMC is the count of the methods implemented within a class. The second measurement is difficult to implement since not all methods are accessible within the class hierarchy due to inheritance. The number of methods and the complexity of the methods involved is a predictor of how much time and effort is required to develop and maintain the class. The larger the number of methods in a class, the greater the potential impact on children since children inherit all of the methods defined in a class.

2) *Response for a Class*

The RFC is the set of all methods that can be invoked in response to a message to an object of the class or by some method in the class. This includes all methods accessible within the class hierarchy. This metric looks at the combination of the complexity of a class through the number of methods and the amount of communication with other classes. The larger the number of methods that can be invoked from a class through messages, the greater the complexity of the class.

3) *Cohesion*

Cohesion is the degree to which methods within a class are related to one another and work together to provide well-bounded behavior. Effective object-oriented designs maximize cohesion since it promotes encapsulation. The third class metrics investigates cohesion.

4) *Lack of Cohesion of Methods*

LCOM measures the degree of similarity of methods by data input variables or attributes structural properties of classes. Any measure of separateness of methods helps identify flaws in the design of classes.

5) *Coupling between Object Classes*

CBO is a count of the number of other classes to which a class is coupled. It is measured by counting the number of distinct non-inheritance related class hierarchies on which a class depends. Excessive coupling is detrimental to modular design and prevents reuse. The more independent a class is, the easier it is reuse in another application. The larger the number of couples, the higher the sensitivity to changes in other parts of the design and therefore maintenance is more difficult.

6) *Inheritance*

Inheritance is a type of relationship among classes that enables programmers to reuse previously defined objects including variables and operators. Inheritance decreases complexity by reducing the number of operations and operators, but this abstraction of objects can make maintenance and design difficult. The two metrics used to measure the amount of inheritance are the depth and breadth of the inheritance hierarchy.

7) *Depth of Inheritance*

Tree The depth of a class within the inheritance hierarchy is the maximum length from the class node to the root of the tree and is measured by the number of ancestor classes. The deeper a class is within the hierarchy, the greater the number of methods it is likely to inherit making it more complex to predict its behavior. Deeper trees constitute greater design complexity, since more methods and classes are involved, but the greater the potential for reuse of inherited methods.

8) *Number of Children*

The number of children is the number of immediate subclasses subordinate to a class in the hierarchy. It is an indicator of the potential influence a class can have on the

design and on the system. The greater the number of children, the greater the likelihood of improper abstraction of the parent and may be a case of misuse of sub classing.

For each metric, threshold values can be adopted, depending on the applicable quality attributes and the application objectives. That is, acceptable ranges for each metric will have to developed, based on the effect of the metric on user design.

VIII. FUTURE WORK

The aim of our survey is to formalize defects of the user interface design including anti patterns and design defects for their detection and correction in object-oriented architectures and techniques to correct them. The techniques of detection and correction shall be generic. To detect automatically sad defects based on their formalization and to propose corrections with explanations to maintainers will be the future challenge to do it exactly as expected. Future work will be to define criteria for the metrics. In Hesitation method, the file-permissions domain shares individuality with many common task domains, like system configuration, data and image manipulation, so the consequences obtained are expected to generalize at least the common domains. Future work will test the method in these and other task domains, such as typing-intensive and long-duration tasks.

IX. CONCLUSION

This survey focuses the problem with user interface defects in object oriented software metrics which has influence on the quality of the software, creating a metrics tool based on object oriented software. These metrics are proposed to add more quality in refining any object oriented software during the different stages. The basic idea of the approach is to add a system of automated detection of usability defects to the tool for user interface development operated by a knowledge base of interface defects. In this survey all the common methodologies and techniques of automated detection of usability defects for user interface development is been discussed. The main task of all the system discussed is to detect defects in a user interface model within the design phase and to give advice to the developer on their elimination.

REFERENCES

- [1] Yann-Ga el Gu eheneuc and Herv´e Albin-Amiot. Using design patterns and constraints to automate the detection and correction of inter-class design defects. In Quioyun Li, Richard Riehle, Gilda Pour, and Bertrand Meyer, editors, proceedings of the 39th conference on the Technology of Object-Oriented Languages and Systems, pages 296–305. IEEE Computer Society Press, July 2001.
- [2] Ms Puneet Jai kaur1, Ms Amandeep Verma, Mr. Simrandeep Thapar ,Software Quality Metrics for Object-Oriented Environments, Proceedings of National Conference on Challenges & Opportunities in Information Technology (COIT-2007).
- [3] G. Cockton, D. Lavery, and A. Woolrych. Inspection-based evaluations. In J. A. Jacko and A. Sears, editors, The Human-Computer Interaction Handbook, chapter 57, pages 1118–1138. Lawrence Erlbaum Associates, Mahwah, NJ, 2003.
- [4] J. S. Dumas. User-based evaluations. In J. A. Jacko and A. Sears, editors, The Human-Computer Interaction Handbook, chapter 56, pages 1093–1117. Lawrence Erlbaum Associates, Mahwah, NJ, 2003.
- [5] D. Kieras. Model-based evaluations. In J. A. Jacko and A. Sears, editors, The Human-Computer Interaction Handbook, Chapter 58, pages 1139–1151. Lawrence Erlbaum Associates, Mahwah, NJ, 2003.
- [6] Robert W. Reeder and Roy A. Maxion, IEEE, International Conference on Dependable Systems & Networks: Philadelphia, PA, 25-28 June 2006.
- [7] Yves Ledru, Lydie du Bousquet, Olivier Maury, and Pierre Bontron. Filtering tobias combinatorial test suites. In Tiziana Margaria-Steffen MichelWermelinger, editor, proceedings of ETAPS/FASE04 – Fundamental Approaches to Software Engineering, volume 2984. LNCS, Springer-Verlag, April 2004.
- [8] Kim Mens, Isabel Michiels, and Roel Wuyts. Supporting software development through declaratively codified programming patterns. Elsevier Journal on Expert Systems with Applications, 23(4). Lecture Notes in Computer Science (LNCS), November 2002.
- [9] Herv´e Albin-Amiot and Yann-Ga`el Gu´eh´eneuc. Meta-modeling design patterns: Application to pattern detection and code synthesis. In Bedir Tekinerdogan, Pim Van Den Broek, Motoshi Saeki, Pavel Hruby, and Gerson Suny´e, editors, proceedings of the 1st ECOOP workshop on Automating Object-Oriented Software Development Methods. Centre for Telematics and Information Technology, University of Twente, October 2001. TR-CTIT-01-35.
- [10] Amnon H. Eden and Rick Kazman. Architecture, design, implementation. In Laurie Dillon and Walter Tichy, editors, proceedings of the 25th International Conference on Software Engineering, pages 149–159. ACM Press, May 2003.
- [11] "Software Quality Metrics for Object-Oriented Environments" Ms Puneet Jai kaur, Ms Amandeep Verma, Mr. Simrandeep Thapar, Proceedings of National Conference on Challenges & Opportunities in Information Technology (COIT-2007) RIMT-IET, Mandi Gobindgarh. March 23, 2007.
- [12] Avizienis, J.C. Laprie, B. Randell, and C. Landwehr. Basic concepts and taxonomy of dependable and secure computing. IEEE Transactions on Dependable and Secure Computing, 1(1):11–33, January-March 2004.

- [13] User Interface Defect Detection by Hesitation Analysis, Robert W. Reeder and Roy A. Maxion, International Conference on Dependable Systems & Networks: Philadelphia, PA, 25-28 June 2006.
- [14] Using Experiments to Build a Body of Knowledge, Victor R. Basili, Experimental Software Engineering Group, Institute for Advanced Computer Studies, Department of Computer Science, University of Maryland and Fraunhofer Center for Experimental Software Engineering, Maryland
- [15] Dependable Systems and Networks, 2006. DSN 2006, User Interface Defect Detection by Hesitation Analysis Reeder, R.W. Maxion, R.A. Dept. of Comput. Sci., Carnegie Mellon Univ., Pittsburgh, PA.
- [16] Ivory, M.Y., Hearst, M.A.: State of the Art in Automating Usability Evaluation of User Interfaces. ACM Computing Surveys, 33 (December 2001) 1–47. Accessible at <http://webtango.berkeley.edu/papers/ue-survey/ue-survey>.
- [17] Gribova V., Kleshchev A. From an Ontology-oriented Approach Conception to User Interface Development. International Journal "Information Theories & Applications". 2003. vol. 10, num.1, p. 87-94.
- [18] "A Method of Estimating Usability of A User Interface Based on its Model", Valeriya Gribova, International Journal "Information Theories & Applications" Vol.14 / 2007.
- [19] Alan Dix, Janet Finlay, Gregory Abowd and Russell Beale, Human-Computer Interaction, Prentice Hall, International, 1993. Chapter 11 contains information on evaluation techniques.
- [20] Clayton Lewis and John Rieman, Task-Centered User Interface Design: A practical introduction. A shareware book published by the authors, 1993. Original files for the book are available by FTP from <ftp.cs.colorado.edu>.
- [21] Victor R. Basili, "Evolving and Packaging Reading Techniques", Through Experimentation Experimental Software Engineering Group.
- [22] Giedre Sabaliauskaite, Fumikazu Matsukawa, Shinji Kusumoto, Katsuro Inoue, "An Experimental Comparison of Checklist-Based Reading and Perspective-Based Reading for UML Design Document Inspection," isese, pp.148, 2002 International Symposium on Empirical Software Engineering (ISESE'02), 2002.
- [23] Filip Van Rysselberghe, Serge Demeyer, "Evaluating Clone Detection Techniques from a Refactoring Perspective," ase, pp.336-339, 19th IEEE International Conference on Automated Software Engineering (ASE'04), 2004.
- [24] "User Interface Defect Detection by Hesitation Analysis", Robert W. Reeder and Roy A. Maxion, 2006

A Survey - Object Oriented Quality Metrics

C. Neelamegam^a, Dr. M. Punithavalli^b

^aLecturer, Department of Information Technology

Sri Nehru Maha Vidyalaya College Arts and Science, Coimbatore

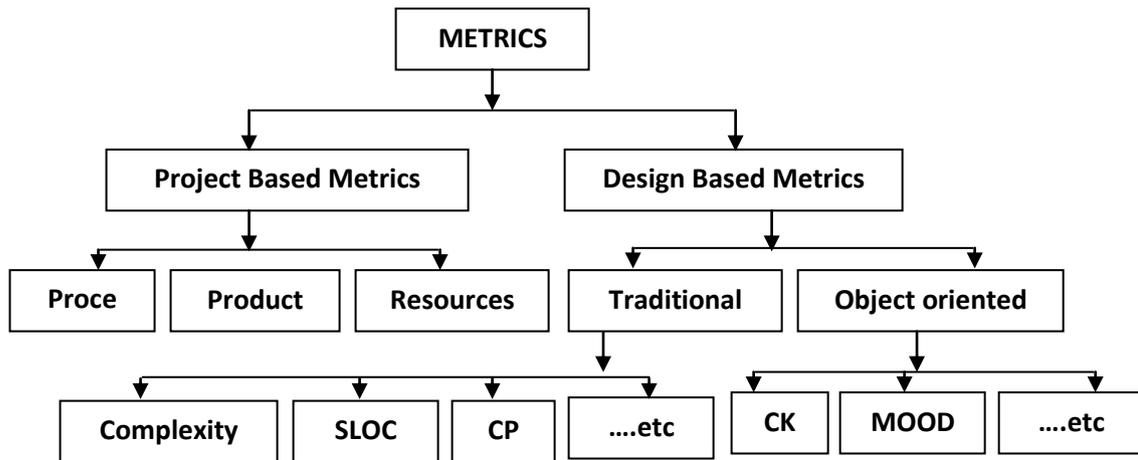
^bDirector & Head Department of Computer Applications,

Sri Ramakrishna College of Arts and Science for women, Coimbatore

Abstract- Object oriented design is becoming more popular in software development environment and object oriented design metrics is an essential part of software environment. This study focus on a set of object oriented metrics that can be used to measure the quality of an object oriented design. The metrics for object oriented design focus on measurements that are applied to the class and design characteristics. These measurements permit designers to access the software early in process, making changes that will reduce complexity and improve the continuing capability of the design. This report summarizes the existing metrics, which will guide the designers to support their design. We have categorized metrics and discussed in such a way that novice designers can apply metrics in their design as needed.

I. INTRODUCTION

Numerous software metrics related to software quality assurance have been proposed in the past and are still being proposed. Several books presenting such metrics exist, such as Fenton's [25], Sheppard's [26] and others. Although most of these metrics are applicable to all programming languages, some metrics apply to a specific set of programming languages. Among metrics of this kind, are those that have been proposed for object-oriented programming languages.



Nowadays, a quality engineer can choose from a large number of object-oriented metrics. The question posed is not the lack of metrics but the selection of those metrics which meet the specific needs of each software project. A quality engineer has to face the problem of selecting the appropriate set of metrics for his software measurements. A number of object-oriented metrics exploit the knowledge gained from metrics used in structured programming and adjust such measurements so as to satisfy the needs of object-oriented programming. On the other hand, other object-oriented metrics have been developed specifically for object-oriented programming and it would be pointless to apply them to structured programming. The above figure shows the hierarchical structure of the metrics.

II. CK METRICS MODEL:

Chidamber and Kemerer define the so called CK metric suite [13]. CK metrics have generated a significant amount of interest and are currently the most well known suite of measurements for OO software [17]. Chidamber and Kemerer proposed six metrics; the following discussion shows their metrics.

Weighted Method per Class (WMC)

WMC measures the complexity of a class. Complexity of a class can for example be calculated by the cyclomatic

complexities of its methods. High value of WMC indicates the class is more complex than that of low values.

Depth of Inheritance Tree (DIT)

DIT metric is the length of the maximum path from the node to the root of the tree. So this metric calculates how far down a class is declared in the inheritance hierarchy. The following figure shows the value of DIT for a simple class hierarchy. DIT represents the complexity of the behaviour of a class, the complexity of design of a class and potential reuse.

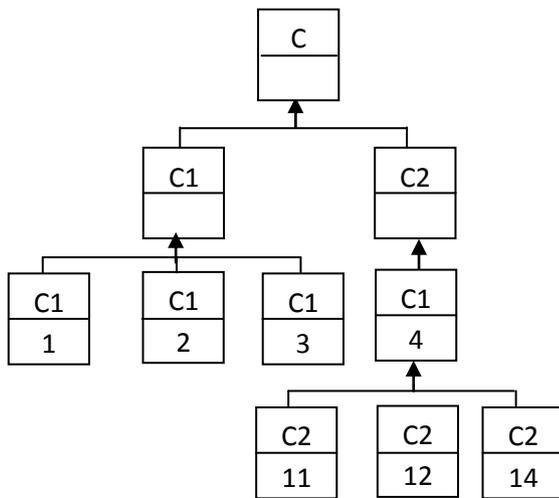


Fig. The value of DIT for the class hierarchy

Thus it can be hard to understand a system with many inheritance layers. On the other hand, a large DIT value indicates that many methods might be reused.

Number of Children (NOC)

This metric measures how many sub-classes are going to inherit the methods of the parent class. As shown in above figure, class C1 has three children, subclasses C11, C12, and C13. The size of NOC approximately indicates the level of reuse in an application. If NOC grows it means reuse increases. On the other hand, as NOC increases, the amount of testing will also increase because more children in a class indicate more responsibility. So, NOC represents the effort required to test the class and reuse.

Coupling between objects (CBO)

The idea of this metrics is that an object is coupled to another object if two object act upon each other. A class is coupled with another if the methods of one class use the methods or attributes of the other class. An increase of CBO indicates the reusability of a class will decrease. Thus, the CBO values for each class should be kept as low as possible.

Response for a Class (RFC)

RFC is the number of methods that can be invoked in response to a message in a class. Pressman [20] States, since RFC increases, the effort required for testing also increases because the test sequence grows. If RFC increases, the overall design complexity of the class increases and becomes hard to understand. On the other hand lower values indicate greater polymorphism. The value of RFC can be from 0 to 50 for a class, some cases the higher value can be 100- it depends on project to project.

Lack of Cohesion in Methods (LCOM)

This metric uses the notion of degree of similarity of methods. LCOM measures the amount of cohesiveness present, how well a system has been designed and how complex a class is [23]. LCOM is a count of the number of method pairs whose similarity is zero, minus the count of method pairs whose similarity is not zero. Raymond [24] discussed for example, a class C with 3 methods M1, M2, and M3. Let $I1 = \{a, b, c, d, e\}$, $I2 = \{a, b, e\}$, and $I3 = \{x, y, z\}$, where $I1$ is the set of instance variables used by method M1. So two disjoint set can be found: $I1 \cap I2 = \{a, b, e\}$ and $I3$. Here, one pair of methods who share at least one instance variable ($I1$ and $I2$). So $LCOM = 2 - 1 = 1$. [13] States "Most of the methods defined on a class should be using most of the data members most of the time". If LCOM is high, methods may be coupled to one another via attributes and then class design will be complex. So, designers should keep cohesion high, that is, keep LCOM low.

III. MOOD METRICS MODEL - (METRICS FOR OBJECT ORIENTED DESIGN)

The **MOOD** metrics set refers to a basic structural mechanism of the OO paradigm as encapsulation (MHF and AHF), inheritance (MIF and AIF), polymorphisms (PF), message-passing (CF) and are expressed as quotients. The set includes the following metrics:

Method Hiding Factor (MHF)

MHF is defined as the ratio of the sum of the invisibilities of all methods defined in all classes to the total number of methods defined in the system under consideration. The invisibility of a method is the percentage of the total classes from which this method is not visible.

Attribute Hiding Factor (AHF)

AHF is defined as the ratio of the sum of the invisibilities of all attributes defined in all classes to the total number of attributes defined in the system under consideration.

Method Inheritance Factor (MIF)

MIF is defined as the ratio of the sum of the inherited methods in all classes of the system under consideration to the total number of available methods (locally defined plus inherited) for all classes.

Attribute Inheritance Factor (AIF)

AIF is defined as the ratio of the sum of inherited attributes in all classes of the system under consideration to the total number of available attributes (locally defined plus inherited) for all classes.

Polymorphism Factor (PF)

PF is defined as the ratio of the actual number of possible different polymorphic situation for class C_i to the maximum number of possible distinct polymorphic situations for class C_i .

Coupling Factor (CF)

CF is defined as the ratio of the maximum possible number of couplings in the system to the actual number of couplings not imputable to inheritance.

IV. MOOSE (METRICS FOR OBJECT-ORIENTED SOFTWARE ENGINEERING)

When **Chidamber and Kemmerer** Introduced the **MOOSE** (Metrics for Object-Oriented Software Engineering) metrics suite [1], also known as C.K. metrics suite, they inaugurated a plethora of Object-Oriented design metrics suits. Since 1994, many other OOD metrics suites [16, 20, 9, and 5] were presented; most of them are built upon the original C.K. metrics suite.

The C.K. metrics suite consists of six metrics that assess different characteristics of the OOD:

- The **Weighted Method per Class (WMC)** assesses complexity of a class through aggregating a complexity measure of its methods. Chidamber and Kemerer did not state, deliberately, a complexity measure to use, leaving the matter as an implementation detail.
- The **Depth of Inheritance Tree (DIT)** assess how deep, in a class hierarchy, a class is. This metric assesses the potential of reuse of a class and its probable ease of maintenance. A class with small DIT has much potential for reuse. (i.e. it tends to be a general abstract class).
- The **Number of Children (NOC)** is a simple measure of the number of classes associated with a given class using an inheritance relationship. It could be used to assess the potential influence a class has on the overall design. "God" classes [24] (i.e. classes with many children) are considered a bad design habit that occurs frequently. NOC helps detecting such classes.
- The **Response for a Class (RFC)** is defined as a count of the set of methods that can be potentially executed in response to a message received by an instance of the class.
- The **Lack of Cohesion in Methods (LCOM)** is the difference between the number of methods whose similarity is zero and the number of methods whose similarity is not zero. LCOM can judge

cohesiveness among class methods. Low LCOM indicates high cohesiveness, and vice versa.

V. QMOOD (QUALITY MODEL FOR OBJECT-ORIENTED DESIGN)

The **QMOOD** (Quality Model for Object-Oriented Design) is a comprehensive quality model that establishes a clearly defined and empirically validated model to assess OOD quality attributes such as understandability and reusability, and relates it through mathematical formulas, with structural OOD properties such as encapsulation and coupling. The QMOOD model consists of six equations that establish relationship between six OOD quality attributes (reusability, flexibility, understandability, functionality, extendibility, and effectiveness) and eleven design properties.

All these are measurable directly from class diagrams, and applicable to UML class diagrams.

VI. OTHER OO METRICS

Chen et al.[9] proposed metrics are 1.CCM (Class Coupling Metric), 2.OXM (Operating Complexity Metric), 3.OACM (Operating Argument Complexity Metric), 4.ACM (Attribute Complexity Metric), 5.OCM (Operating Coupling Metric), 6.CM (Cohesion Metric), 7.CHM (Class Hierarchy of Method) and 8.RM (Reuse Metric). Metrics 1 through 3 are subjective in nature; metrics 4 through 7 involve counts of features; and metric 8 is a Boolean (0 or 1) indicator metric.

Since terminology varies among object oriented programming languages, the authors consider the basic components of the paradigm as objects, classes, attributes, inheritance, method, and message passing. They propose that each object oriented basic concept implies a programming behaviour. They assembled metrics are: Data Abstraction Coupling (DAC), Number of methods (NOM), Message Passing Coupling (MPC), and Number of semicolons per class (Size1), Number of methods per attributes (Size2). There is no individual breakdown of which of these metrics is significant in the prediction [3].

VII. FUTURE WORKS

I've surveyed metrics for software model complexity which is a combination of some of the metrics mentioned above with a new approach. With these metrics we can measure software's overall complexity (including all its components and classes). Also there are metrics for measuring software's run-time properties and would be worth studying more.

VIII. CONCLUSION

In this paper, we have surveyed four object-oriented design quality models. The work of Chidamber and Kemerer has been seminal in defining, and validating quality models. Lorenz and Kidd metrics are criticized for not being a part of a quality model, however, they have the advantages of

being well-defined, easy to collect, and could be computed in the early phases of design. MOOD model is as a very well-defined, through mathematical formulas and OCL statements, empirically validated, supported by a tool, and most importantly provide thresholds that could be used to judge the metrics collected from a given design. The QMOOD model enjoys similar properties as the MOOD model of being well-defined, empirically validated and supported by a tool. QMOOD distinguishes itself by providing mathematical formulas that links design quality attributes with design metrics. This allowed computing a Total Quality Index (TQI), which were already used by [7] authors to compare fourteen class diagrams.

We surveyed a group of desirable properties for OOD quality models, and then we used them to compare the presented OOD quality models. Based on this comparison, we conclude that the QMOOD suite is the most complete, comprehensive, and supported suite.

REFERENCES

- [1] Jagdish Bansiya, and Carl G.Davis, 2002," A Hierarchical Model for Object-Oriented Design Quality Assessment", IEEE Transactions on Software Engineering, VOL. 28, No. 1, Jan 2002.
- [2] Fernando Brito e Abreu and Rogerio Carapuca, 1994,"Object-Oriented Software Engineering: Measuring and Controlling the Development Process", 4th Int. Conf. on Software Quality, McLean, VA, USA, 3-5 October 1994.
- [3] Fernando Brito e Abreu and Walcelio Melo, 1996, "Evaluating the Impact of Object-Oriented Design on Software Quality", 3rd Int'l S/W Metrics Symposium, March 1996, Berlin, Germany.
- [4] Abreu, Fernando B., Carapuca, Rogerio. "Candidate Metrics for Object- Oriented Software within a Taxonomy Framework.", Journal of systems software 26, 1(July 1994).
- [5] Abreu, Fernando B: "The MOOD Metrics Set," Proc. ECOOP'95 Workshop on Metrics, 1995.
- [6] Abreu, Fernando B: "Design metrics for OO software system", ECOOP'95, Quantitative Methods Workshop, 1995.
- [7] Abreu, Fernando B, Rita, E., Miguel, G.: "The Design of Eiffel Program: Quantitative Evaluation Using the MOOD metrics", Proceeding of TOOLS'96 USA, Santa Barbara, California, July 1996.
- [8] Bellin, D., Manish Tyagi, Maurice Tyler: "Object-Oriented Metrics: An Overview", Computer Science Department, North Carolina A, T state University, Greensboro, Nc 27411-0002.
- [9] Chen, J-Y., Lum, J-F.: "A New Metrics for Object-Oriented Design." Information of Software Technology 35,4(April 1993):232-240.
- [10] Lorenz, Mark & Kidd Jeff: "Object-Oriented Software Metrics", Prentice Hall, 1994.
- [11] Bellin, D., Manish Tyagi, Maurice Tyler: "Object-Oriented Metrics: An Overview", Computer Science Department, North Carolina A, T state University, Greensboro, Nc 27411-0002.
- [12] Archer C., Stinson M.: "Object Oriented Software Measure", Technical report CMU/SEI-95-TR-002, ESC-TR-95-002, 1995.
- [13] Chidamber, Shyam, Kemerer, Chris F. "A Metrics Suite for Object- Oriented Design" M.I.T. Sloan School of Management E53-315, 1993.
- [14] Li, Wei, Henry, Salley. "Maintenance Metrics for the Object Oriented Paradigm", First International Software Metrics Symposium. Baltimore, Maryland, May 21-22, 1993. Los Alamitos, California: IEEE Computer Society Press, 1993.
- [15] Rosenberg, H. Linda, Lawrence E. Hyatt: "Software Quality Metrics for Object-Oriented Environments", Crosstalk Journal, 1997.
- [16] Booch, G: "Object-Oriented Analysis and Design with Applications", 2nd ed., Benjamin Cummings, 1994.
- [17] Harrison, R., Samaraweera, L.G., Dobie, M.R., and Lewis, P.H: "Comparing Programming Paradigms: An Evaluation of Functional and Object-Oriented Programs," Software Eng. J., vol. 11, pp. 247-254, July 1996.
- [18] Demeyer, S., Ducasse, S. and Nierstrasz, O: "Refractoriness via change metrics". In Proc. Int. Conf. 2000, ACM Press.
- [19] Fenton, N., S.L. Pfleeger: "Software Metrics: A Rigorous and Practical Approach", PWS Publishing Co.
- [20] Roger S. Pressman: "Software Engineering", Fifth edition, ISBN 0077096770.
- [21] Harrison, R., Counsel, S.J. Nithi, R.V: "An Investigation into the Applicability and Validity of Object-Oriented Design Metrics", technical report.
- [22] Ramil, J.E and Lehman, M.M: "Metrics of evolution as effort predictors – a case study". In Conf. Software Maintenance, pages 163-172, October.
- [23] Raymond, J. A, Alex, D.L: "A data model for object oriented design metrics", Technical Report 1997, ISBN 0836 0227.
- [24] Alexander et al 2003,"Mathematical Assessment of Object-Oriented Design Quality", IEEE Transactions on Software Engineering, VOL. 29, NO. 11, November 2003.
- [25] N. Fenton & S.L. Pfleeger, "Software Metrics: A Rigorous & Practical Approach", Second edition, 1997, International Thomson Computer Press.
- [26] M.J. Sheppard & D. Ince, "Derivation and Validation of Software Metrics, Clarendon Press, Oxford, UK, 1993

Modeling and Analysis of the Associative Based Routing (ABR) Protocol by Using Coloured Petri Nets

Rahul Bhargava

Deptt. of Computer Science and Engineering
M.A.N.I.T.
Bhopal, India
rahuluitbu@gmail.com

Rekha Kaushik (P.hd.)

Deptt. of Computer Science and Engineering
M.A.N.I.T.
Bhopal, India

Abstract— In an ad-hoc mobile network where mobile hosts (MHs) are acting as routers and where routes are made inconsistent by MHs' movement. In associativity-based routing protocol (ABR) where a route is selected based on nodes having associativity states that imply periods of stability. In this manner, the routes selected are likely to be long-lived and provides maximum throughput. In this paper we create a Coloured Petri Nets (CPN) model of the route discovery phase of the ABR protocol. The description of the discovery phase of the ABR protocol is given by arc inscriptions and functions in the CPN tool. Some dynamic properties of the discovery phase of the ABR protocol like fairness and liveness have been verified by means of CPN state space tool.

Keywords-

Mobile Ad-hoc Network (MANET), Associativity-Based Routing Protocol (ABR), Coloured Petri Nets (CPN), Modeling and Analysis.

I. INTRODUCTION

In a Mobile Ad-hoc Network (MANET) basically there are two types of routing protocols. Proactive and Reactive. The Proactive protocols are table driven and each node maintains a route to every other node in the MANET. Due to limited memory, processing power and battery capacity this protocol model is not suitable. In Reactive protocol route has been created on-demand that generates less traffic than proactive protocol. ABR [6, 8] is reactive routing protocol that only maintain routes for sources that actually desire routes. In addition, routing decisions are performed at the destination and only the best route will be selected and used while all other possible routes remain passive.

The aim of this paper is to provide the first Coloured Petri nets (CPN) model of the route discovery phase of the ABR protocol by using CPN tool [5]. It is a best tool in terms of explicit description of both states and actions. It is a graphical language for constructing models of wireless protocols and analyzing their properties. In this paper dynamic properties like fairness and liveness of the discovery phase of the ABR protocol have been verified.

This paper is organized as follows. Section 2 summarizes the related work. Section 3 gives an overview of the protocol. Section 4 introduces the assumed CPN model of the route discovery phase of the ABR protocol. Section 5 gives the partial analysis and results of the protocol. Finally, in section 6 we sum up the conclusion and future work

II. RELATED WORK

Yuan and Billington [1] provides the first CPN model of the basic functions of the Destination-Sequenced Distance-Vector (DSDV) routing protocol as a first step towards its formal specification and verification. It introduces the basic features of DSDV and provides abstract CPN model of the DSDV routing mechanism. DSDV is a proactive protocol. The modeling of the Dynamic MANET on-demand routing protocol (DYMO) is presented in [2, 3]. DYMO [3] uses the highly compact CPN model of the DYMO protocol and it provides the simulation to investigate properties of the protocol while DYMO [2] describes the hierarchical CPN model of the DYMO protocol.

III. PROTOCOL OVERVIEW

ABR is routing protocols that checks the association between nodes and transfer the messages from sources to destinations. Association [6] is a concept that verifies the wireless connection between two nodes that actually involve in the procedure of transferring the packets.

1) Property of Associativity

The rule of Associativity states that a Mobile Host (MH) association with its neighbor would depend on number of beacons that actually receives by both nodes. A MH periodically broadcasts beacons identifying itself and constantly updates its associativity ticks in accordance with other MH sighted in its neighborhood. A MH is said to exhibit a high state of mobility when it has low associativity ticks with its neighbors. However, if high associativity ticks are observed, the MH is in the stable state

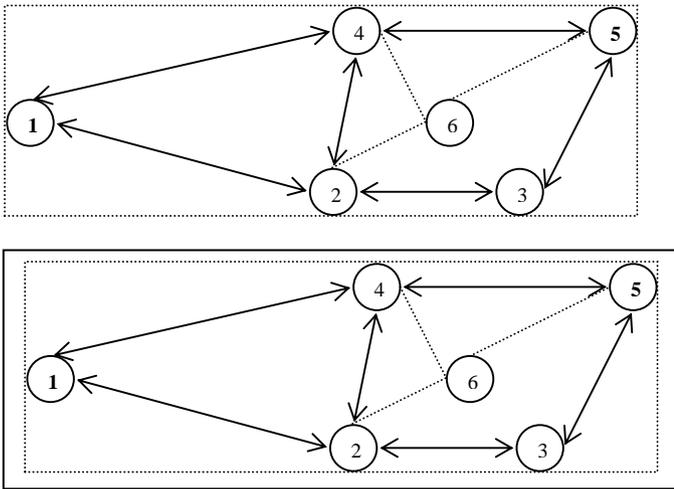


Figure 1. A simple topology with six nodes. Dotted Line shows the low association between nodes, Double Arrow indicates the high association between nodes.

and this is the ideal point to select the MH to perform ad-hoc routing.

2) Description of the Protocol

A simple example scenario illustrating only route discovery phase of the protocol is shown in Fig 1. The scenario consists of six nodes 1 to 6. A double arrow indicates that the two nodes are in high association with each other. It means both nodes can send messages. Dotted line indicates low association between nodes. It means that there is no communication between these two nodes due to low associativity ticks. Source node 1 wants to establish a route to destination node 5. Node 1 broadcast a BQ request which is received by node 2 and 4. When receiving the BQ request, node 2 and node 4 will check that it is desired destination node or not. Now these nodes will create an entry in their routing table specifying a route back to the originator node 1. These nodes also save the route quality information (associative ticks etc.) in BQ request with hop count value. Since node 2 and node 4 are not the target of BQ request. So both nodes will broadcast this BQ request to their neighboring nodes. Node 4 broadcast to node 2, 1 and 5. Node 1 and node 2 will drop this BQ request because both have processed this same BQ request earlier. Only node 5 checks that it is destination node desired by source node and will create the routing table entry. Node 5 is the destination node, but it waits before creating the REPLY message because it will choose best path. Now node 2 broadcast to nodes 3, 4 and 1. Node 4 and node 1 will drop this request from node 2. Only node 3 will create the entries in its routing table and check that it is destination node or not. Again it will broadcast this request to its neighboring node. Only node 5 will receive this request. Node 5 enters its routing entry in its table and check that is desired destination node. Now node 5 chooses best path among them. It should be noted that Node 6 will not receive any broadcast messages because it does not have association with the broadcasting nodes.

At destination node shortest hop count and high associative ticks are the selection criteria. It will create the REPLY message to source node by using the 5-4-1 path because it is having shortest path and highest associative ticks. However

another path is available 5-3-2-1. But this path having longer hop count so it will be discarded. Now REPLY message unicast to path 5-4-1. Node 5 unicast this message to node 4. Node 4 checks whether it is source node or not. Node 4 will create its routing entries and unicast this message to node 1. Now node 1 updates its routing entries and check that it is source node or not. So this is the overall procedure for creating the path between source to destination in the route discovery phase of the ABR protocol. Now source is ready to send data packet over ad-hoc network.

IV. CPN MODEL OF ABR ROUTING PROTOCOL

1) Assumptions

Some assumptions have been taken to describe the model.

- For route qualities only one parameter associative ticks has been taken in the model.
- An associative tick is assumed as static value (either HIGH or LOW). HIGH shows that nodes are in the range of one another and LOW value indicates that nodes are not in the range of each other.
- Mobility has not been modeled. We assume that either node is in the transmission range of another node or not.
- We are not modeling broadcast completely in the model. In a MANET, the broadcast of a node may be received by: no nodes, one node (only one node can communicate with this node) or any number of nodes. Broadcast messages may be lost, indicating that no nodes receive it or may be received by any arbitrary node. Thus this receiving node may be only receiver or one of the several receivers. Then the node transmits the message after processing if it not the target. In addition we do not model operations of other receivers of a broadcast.

2) CPN Model

Based on the assumptions, we create the CPN model of ABR protocol. This model represents the route discovery phase of the ABR protocol. The nodes are stored in MNODE, which is typed by the color set MNODE. MNODE is a product of "NODE", "RT", "DEST", "try", "seq" and AT". "NODE" as an integer from zero to five. "RT" is a list of route entries that represent the incoming, outgoing, source and destination node entries. It contains two tuple "DEST" and "hopc". hopc represents the number of hope count from that DEST (destination). This DEST represents incoming, outgoing, source and destination node addresses. Next "DEST" represents the destination is the node address, this is desired address where we want to send the data. "try" represents the number of tries (how many times broadcast request must repeat itself). It is also an integer from zero to five. "seq" represents the sequence number uniquely identified the routing entry for given route request and last tuple "AT" represents neighboring table that contains only the parameter that is associative ticks. It defines as "HIGH" and "LOW" values.

We consider a configuration of five nodes in the protocol for our initial experiments. Thus ABR nodes are initially marked by five nodes, where route table has a single entry to itself. It means that initially it does not have any route entries for incoming, outgoing, source and destination nodes. Third zero indicates that no destination address has been allocated for broadcast request. try denotes number of tries for broadcast

```

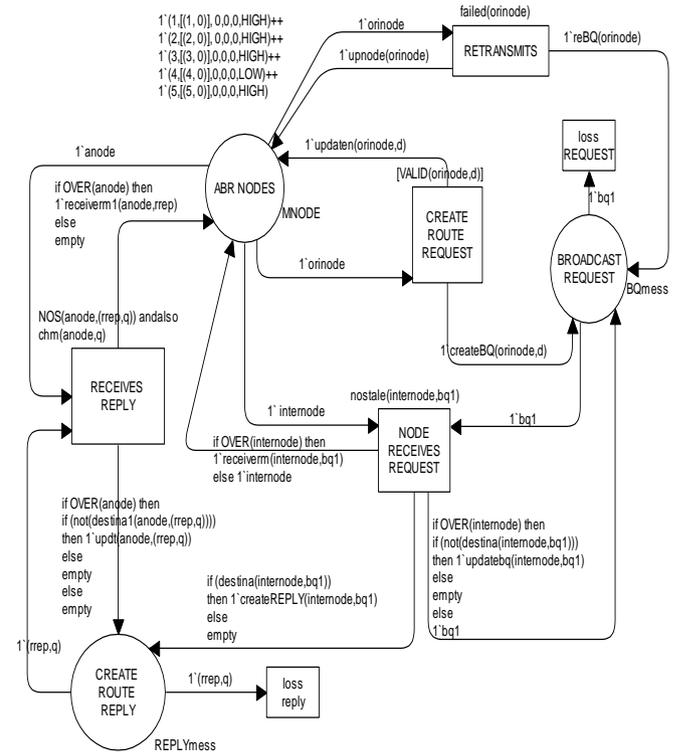
colset NODE=int with 0..5;
colset DEST=NODE;
colset SSADD=NODE;
colset hopcount=int ;
colset try=int with 0..5 ;
colset AT=with LOW|HIGH;
colset seq=int ;
colset hopc=int;
colset rt=product DEST*hopc;
colset RT=list rt;
colset MNODE=product NODE*RT*DEST*try*seq*AT;
colset SADD=NODE;
colset DADD=NODE;
colset msgtype=with BQREQ|REPCP|RN|LQ|R-D;
colset ADD=NODE;
colset RQ =product ADD*AT*hopc;
colset Rquality=list RQ;
colset TYPE=msgtype;
colset BQmess=product TYPE*SADD*DADD*hopc*Rquality*seq;
colset REPLYmess=product BQmess*NODE;
var internode,orinode,anode:MNODE;
var bq1,rrep:BQmess;
var d,q:NODE;
    
```

Global declarations of the CPN model

request. It set to zero, indicating that at the beginning no BQ request has been broadcast. AT represents the neighboring table that contains HIGH or LOW values. HIGH indicates that this node is in range of another node (high association) and it can receive and send the broadcast request in the wireless network. While LOW indicates that nodes are not in the range of another node (low association) and can not receive and send the broadcast request in the network.

“BQmess” is type of broadcast request and it is a product of “TYPE”, “SADD”, “DADD”, “hopc”, “Rquality” and “seq”. “TYPE” is a message type is a color set of “BQREQ”, “REPCP”, “RN”, “LQ”, and “RD”. “BQREQ” represents the broadcast request message and “REPCP” represents the unicast reply messages and “RN”, “LQ” and “RD” represent the Route Notification, Localized Query and Route Delete control messages. We have not modeled “RN”, “LQ” and “RD” messages in the CPN model, because all these messages are part of route reconstruction and route deletion phase of the ABR protocol. “SADD” and “DADD” is a type of node addresses, “hopc” represents number of hop count between source to destination counted by every

intermediated nodes. “Rquality” is a list consists of “ADD”, “AT” and “hopc”. Where “ADD” is a node address (intermediate nodes). “AT” represents associative ticks, which is type of boolean color set and “hopc” is a hop count a type of integer color set. It is basically a distance of source to that intermediate node. “Seq” indicates sequence number uniquely identify the BQ request.



Proposed CPN model.

Reply Messages is a type of color set “REPLYmess”. “REPLYmess” is a product of “BQmess*NODE”, Where “BQmess” is a type of broadcast request and “NODE” is type of color set integer, which represents intermediate nodes that will come in the path of destination to source whenever reply messages has been sent back to source. Transition NODE RECEIVES REQUEST models the process of a node handling a received BQ request, the arc inscriptions from the place MNODE to this transition has a variable 1`internode, representing a arbitrary intermediate node along the path to the destination. The arc inscription from BROADCAST REQUEST messages to this transition has a variable 1`bq1, indicating the received BQ request. The guard nostale(internode,bq1) ensures that intermediate node is not a source node as well as these nodes do not have any invalid route entry. The function OVER (internode) checks the associativity of the receiving node. The arc inscription 1`receiverm (internode, bq1) returns the MNODE values of respective intermediate node. The function destina(internode, bq1) checks the values of the intermediate nodes whether that it matches with the

destination. If it does not match then BQ request holds the new intermediate node entry. The function l^{updatebq} (internode, bq1) performs this task. If it matches then receiving intermediate node is the destination node. This will create the reply messages towards the source node. Function $l^{\text{createREPLY}}$ (internode, bq1) creates the reply message. The transition RECEIVES REPLY represents the process of modelling of the reply messages which unicasted towards the source. The arc inscription $l^{\text{(rrep,q)}}$ represents the REPLYmess. Where rrep is a type of BQmess and q is a type of NODE (integer colour set). The arc inscription l^{anode} from ABR NODES to RECEIVES REPLY represents the next coming intermediate nodes which hold the path between destination to source in reply message. The guard NOS (anode, (rrep,q)) andalso chm (anode,q) represents the two functions separately. First function NOS (anode, (rrep,q)) checks that route table of the intermediate nodes do not have the entries of the incoming nodes in the reply messages. Second function chm(anode,q) checks whether next coming node is the right intermediates node for which reply message keeps the entries in its table. Now we can say that a reply message is fresh and now ready to update the route table entries of the intermediate nodes. The function $l^{\text{receiverm1}}$ (anode, rrep) represents the updating procedure of the route table of the respective intermediate node. Function (not (destinal (anode, (rrep, q)))) checks whether anode is the source node or not. If it is not the source node then this reply message again send to its next intermediate node. Function updt (anode, (rrep, q)) update the reply message. Now a reply message is ready to send its next intermediate node.

Transition RETRANSMITS represents the retransmission of the BQ request message. The guard failed (orinode) checks that correct node rebroadcast the messages. try must be less than its maximum tries. If it is satisfied then the function l^{upnode} (orinode) updates the values of orinode. seq (sequences number) and try both will be incremented by one. Function l^{reBQ} (orinode) again creates the broadcast request for orinode.

V. SIMULATION AND RESULTS

In order to provide some insight into the operation of our CPN model, we consider the following interactive simulation of its behavior. The number of nodes in the MANET is governed by the initial marking of the CPN model in fig. (3) (i.e. the marking of place ABR NODES), and can be extended easily. We have taken 5 nodes for our CPN operation. Suppose their addresses are represented as 1,2,3,4 and 5 respectively. As shown in fig. (4).

```

(1)  l` (1, [(1, 0)], 0, 0, 0, HIGH) ++
      l` (2, [(2, 0)], 0, 0, 0, HIGH) ++
      l` (3, [(3, 0)], 0, 0, 0, HIGH) ++
      l` (4, [(4, 0)], 0, 0, 0, LOW) ++
      l` (5, [(5, 0)], 0, 0, 0, HIGH)

```

```

(2)  l` (1, [(1, 0)], 5, 0, 1, HIGH) ++
      l` (2, [(2, 0)], 0, 0, 0, HIGH) ++
      l` (3, [(3, 0)], 0, 0, 0, HIGH) ++
      l` (4, [(4, 0)], 0, 0, 0, LOW) ++
      l` (5, [(5, 0)], 0, 0, 0, HIGH)
      l` (BQREQ, 1, 5, 0, [(1, HIGH, 0)], 1)
-----
(3)  l` (1, [(1, 0)], 5, 0, 1, HIGH) ++
      l` (2, [(2, 0)], 0, 0, 0, HIGH) ++
      l` (3, [(1, 1), (3, 0)], 0, 0, 0, HIGH) ++
      l` (4, [(4, 0)], 0, 0, 0, LOW) ++
      l` (5, [(5, 0)], 0, 0, 0, HIGH)
      l` (BQREQ, 1, 5, 1, [(1, HIGH, 1), (3, HIGH, 0)], 1)
-----
(4)  l` (1, [(1, 0)], 5, 0, 2, HIGH) ++
      l` (2, [(2, 0)], 0, 0, 0, HIGH) ++
      l` (3, [(1, 1), (3, 0)], 0, 0, 0, HIGH) ++
      l` (4, [(4, 0)], 0, 0, 0, LOW) ++
      l` (5, [(5, 0)], 0, 0, 0, HIGH)
      l` (BQREQ, 1, 5, 0, [(1, HIGH, 0)], 2)
-----
(5)  l` (1, [(1, 0)], 5, 1, 2, HIGH) ++
      l` (2, [(2, 0)], 0, 0, 0, HIGH) ++
      l` (3, [(1, 1), (3, 0)], 0, 0, 0, HIGH) ++
      l` (4, [(4, 0)], 0, 0, 0, LOW) ++
      l` (5, [(5, 0)], 0, 0, 0, HIGH)
      l` (BQREQ, 1, 5, 1, [(1, HIGH, 1), (3, HIGH, 0)],
2)
-----
(6)  l` (1, [(1, 0)], 5, 1, 2, HIGH) ++
      l` (2, [(1, 2), (2, 0), (3, 1)], 0, 0, 0, HIGH) ++
      l` (3, [(1, 1), (3, 0)], 0, 0, 0, HIGH) ++
      l` (4, [(4, 0)], 0, 0, 0, LOW) ++
      l` (5, [(5, 0)], 0, 0, 0, HIGH)
      l` (BQREQ, 1, 5, 2, [(1, HIGH, 2), (2, HIGH, 0),
(3, HIGH, 1)], 2)
-----
(7)  l` (1, [(1, 0)], 5, 1, 2, HIGH) ++
      l` (2, [(1, 2), (2, 0), (3, 1)], 0, 0, 0, HIGH) ++
      l` (3, [(1, 1), (3, 0)], 0, 0, 0, HIGH) ++
      l` (4, [(4, 0)], 0, 0, 0, LOW) ++
      l` (5, [(1, 3), (2, 1), (3, 2), (5, 0)], 0, 0, 0, HIGH)
      l` ((REPCP, 5, 1, 3, [(5, HIGH, 0)], 1), 2)
-----
(8)  l` (1, [(1, 0)], 5, 1, 2, HIGH) ++
      l` (2, [(1, 2), (2, 0), (3, 1), (5, 1)], 0, 0, 0, HIGH)
++
      l` (3, [(1, 1), (3, 0)], 0, 0, 0, HIGH) ++
      l` (4, [(4, 0)], 0, 0, 0, LOW) ++
      l` (5, [(1, 3), (2, 1), (3, 2), (5, 0)], 0, 0, 0, HIGH)
      l` ((REPCP, 5, 1, 3, [(5, HIGH, 1), (2, HIGH, 0)],
1), 3)
-----
(9)  l` (1, [(1, 0)], 5, 1, 2, HIGH) ++
      l` (2, [(1, 2), (2, 0), (3, 1), (5, 1)], 0, 0, 0, HIGH)
++
      l` (3, [(1, 1), (3, 0), (5, 2), (2, 1)], 0, 0, 0, HIGH)
++
      l` (4, [(4, 0)], 0, 0, 0, LOW) ++

```

1` (5, [(1, 3), (2, 1), (3, 2), (5, 0)], 0, 0, 0, HIGH)
 1` ((REPCP, 5, 1, 3, [(5, HIGH, 2), (2, HIGH, 1), (3, HIGH, 0)], 1), 1)

 Markings of the CPN model during the simulation

Source node 1 wants to send its data packets to destination node 5 but there is no path between these two nodes. Now node 1 will create the path to node 5. So for this purpose in simulation node 1 will send BQ request to node 5. Fig. 4 shows whole simulation of the discovery phase of the ABR protocol.

First this request is being sent to node 3. Node 3 is the intermediate node. So in (3) of Fig. 4 routing table of node 3 is updated and intermediate node entry for node 3 in the BQ request has been filled. Entries of the BQ request show the list of nodes with its associative ticks and hop count from the source node. In (4) of Fig. 4 BQ request has been dropped. So BQ request is being created again with higher sequence number. Sequence number is the unique identification of the BQ request in the wireless network. Number of tries has been incremented by value of one. Next intermediate node 3 received this request in (5) of Fig. 4. Again entries of the routing table of node 3 and BQ request have been updated. Now this request will be sent to node 2 and update its entries in (6) of Fig. 4. Node 2 is the intermediate node. Finally destination node 5 receives this BQ request. Whenever node 5 will receive this BQ request it chooses the best path (in terms of minimum hop count and high associative ticks) and creates the REPLY message. Now a REPLY message is ready to send back to source node. REPLY message sent back by using the path from destination to source. Whenever this REPLY message sent back to source then every intermediate node will update its incoming, outgoing, source and destination node entries in its routing table. (7) of Fig. 4, (8) of Fig. 4, and (9) of Fig. 4 showing the whole entries of the routing tables.

So this is the overall procedure of the route discovery phase of the ABR protocol by means of CPN model.

1) *State Space Report*

It is necessary to first debug and investigate a system by means of simulations. Simulation works in a similar way to program testing. For verifying the properties of the behavior of the system, state space analysis is needed. State space is a directed graph which has a node for each reachable marking and an arc for each occurring binding element. The statistical information (TABLE I) shows the size of the state space. State space having 8546 nodes and 18141 arcs and it has been calculated in 300 secs. This report is partial.

TABLE I. STATISTICS

State Space
Nodes: 8546
Arcs: 18141
Secs: 300
Status: Partial

TABLE II. LIVENESS PROPERTIES

Dead Transition Instances
None
Live Transition Instances
None

TABLE III. FAIRNESS PROPERTIES

New_Page'CREATE_ROUTE_REQUEST 1
No Fairness
New_Page'NODE_RECEIVES_REQUEST 1
Impartial
New_Page'RECEIVES_REPLY 1
No Fairness
New_Page'RETRANSMITS 1
No Fairness
New_Page'loss_REQUEST 1
No Fairness
New_Page'loss_reply 1
No Fairness

These results (TABLE II and III) show that our protocol support set of dynamic properties. Liveness property verifies the looping behavior of the system. This tells us that how set of binding elements remains active.

According to TABLE II there are no dead transitions. This is similar to dead code in a programming language and it means that each transition is enabled in at least one reachable marking. No live transitions show that protocol is loop free. So there is no packet duplication in the protocol.

The final part of the state space report is shown in TABLE III. It provides information about the fairness properties. It means that how individual transitions occur.

Fairness property tells us how often the different binding elements occur. Transition `NODE_RECEIVES_REQUEST` is impartial. This tells us that it cannot have an infinite occurrence sequences in the transition. There are Transitions `CREATE_ROUTE_REQUEST`, `RECEIVES_REPLY`, `RETR-ANSMITS`, `loss_REQUEST` and `loss_reply` have shown the "No fairness property". This tells us that it is possible to have an infinite occurrence sequences in all these transitions.

VI. CONCLUSION AND FUTURE WORK

In this paper working methodology of the route discovery phase of the ABR protocol has been observed. This means that how discovery phase has worked. Then we gave proposed CPN model of the route discovery phase of the ABR protocol. The interactive environment of the CPN tool has shown the dynamic behavior of the proposed CPN model. Dynamic behavior of the CPN model works as expected. The expected behavior has been verified by the simulation results of the protocol model. Proposed model also analyze the dynamic properties of the route discovery phase of the ABR protocol by using state space tool. These properties ensure that our protocol is loop free and deadlock [9]

free. It is also verified that transitions in the CPN model works correctly.

Verification of the hierarchical CPN model of this phase can be done as a future work. Simulation and analysis of the route re-construction and route deletion phase of the ABR protocol will be a part of future extension.

REFERENCES

- [1] C. Yuan and J. Billington, "An Abstract Model of Routing in Mobile Ad Hoc Networks," in Proc. of CPN'05, pp. 137–156, DAIMI PB-576, 2005.
- [2] K. Espensen, M. K.Kjeldsen, and L. M. Kristensen, "Towards modelling and validation of the DYMO routing protocol for mobile ad-hoc networks," in Proc. of CPN'07, pp. 243-262, DAIMI PB-584, 2006.
- [3] C. Yuan and J. Billington, "A Coloured Petri Net Model of the Dynamic MANET On-demand Routing Protocol," in Proc. of CPN'06, pp. 37–56, DAIMI PB-579, 2006.
- [4] K. Jensen, "An Introduction to the Theoretical Aspects of Coloured Petri Nets," in Lecture Notes in Computer Science (803), pp. 230-272, SpringerVerlag, 1994.
- [5] Design/CPN homepage. <http://www.daimi.au.dk/designCPN/>
- [6] C.-K. Toh, "Associativity-Based Routing For Ad Hoc Mobile Networks," *Wireless Pers. Commun. J.*, Special Issue on Mobile Networking and Computing Systems, Kluwer Academic, vol. 4, no. 2, Mar. 1997, pp. 103-39.
- [7] L.M. Kristensen, S. Christensen, and K. Jensen, The practitioner's Guide to Coloured Petri Nets. *International Journal on Software Tools for Technology Transfer (STTT)*, 2(2):98–132, 1998.
- [8] C-K Toh, "Long-Lived Ad-Hoc Routing Based On The Conpect Of Associativity," <http://www.ietf.org/proceeding/99nov/1-D/draft-ietf-manet-longlived-adhoc-routing-00.txt>, 2000.

A Framework of Distributed Dynamic Multi-radio Multi-channel Multi-path Routing Protocol in Wireless Mesh Networks

¹K.Thangadurai, ²Anand Shankar

¹Lecturer, ²Research Scholar,

Department of Computer Science and Engineering,
V.M.K.V.Engineering College, Salem, Tamilnadu,
India.

Email: {thangadurai_mk@rediffmail.com,
anand_achu2@rediffmail.com}

Abstract— Wireless Mesh Networks (WMNs) have gained a lot of attention recently. Many efforts are made to design proper routing protocols for WMNs. Existing multi-radio multi-channel routing protocols utilize only one path for transmission, and some multi-path routing protocols consider only single channel situation, in which multi-path routing won't improve end-to-end throughput efficiently. In our paper, we propose a framework for distributed reactive routing protocol in WMNs, which utilizes multi-radio multi-channel technique, as well as multi-path transmission strategy. Dynamic channel assignment is used to avoid the inter-flow and intra-flow channel competition and interference. Our protocol establishes and maintains two or more channel-dimensional disjoint paths, and then every data flow is splitted into multiple paths, in order to increase the total end-to-end transmission throughput. Demo and NS2 simulations are carried out for the evaluation of the performance of our proposed protocol comparing with AODV and other related routing protocols. It is shown our proposal can increase end-to-end throughput significantly.

I. INTRODUCTION

Recently, the Wireless Mesh Network (WMN) [1] becomes popular and important in wireless technology and industry fields. WMNs are believed to be a promising technology to offer high bandwidth for wireless access to the Internet. In the infrastructure of a typical WMN, fixed wireless mesh routers and gateways are highly connected together in a ad-hoc manner. Mesh routers are practically Access Points (APs) equipped with functionalities of IEEE 802.11 standard series [2], e.g. 802.11a/b and 802.11g, where normal wireless devices can connect for communication services. Mesh routers performs not only as a role of data aggregator, but also as a role to relay data to gateways. WMN gateways are

devices with high bandwidth that can provide internet connections to routers. Data flows can be formed in multi-hop manner from wireless devices through each mesh routers to the gateways, or to other mesh routers and devices in other areas.

WMN infrastructure benefits from large coverage of multi-hop wireless connections, but it also suffers channel competition and collision problems. Because of half-duplex property of radio antenna, one network radio cannot transmit and receive at the same time, the capacity of transmission link can only achieve a half of basic MAC layer rate. Broadcast nature of the wireless medium makes nodes work in common communication channel, therefore nodes have to wait for other nodes that are occupying the channel, and then compete with each other for next chance. If two nodes in each other's transmission range and in the same channel transmit at the same time, there will be collision. It is difficult to avoid transmission collisions, although some mechanism like RTS/CTS are invented to fix the hidden terminal problem resulting in reduction of throughput.

Multi-channel technique can significantly avoid transmission competition and collision in the same channel. Orthogonal channels use non-overlapping frequency bands, thus there is no interference among them, for example, in IEEE 802.11a there are 12 orthogonal channels. Routing protocols assigning diverse channels to each hop of data flow can reduce intra-flow channel interference and competition therefore can improve end-to-end throughput times. Radio is a network function with antenna that can switch and transmit data in a specified channel, and wireless devices are able to equip two or even more radio which are working in different channels, to make full-duplex transmission and provide more efficient routing.

Multi-path routing strategies are also designed to split and transmit data through two or more different paths to destination simultaneously. However, multi-path routing

cannot achieve times of throughput as we expect since inter-intra-flow channel competition and interference. Therefore we propose a novel framework for multi-channel and multi-path routing protocol in WMNs which use both techniques.

The rest of this paper is organized as follows: In Section II we will briefly compare different routing strategies in WMNs then propose the motivation of our work. In Section III we will explain our protocol and algorithm in detail. Section IV is the part of simulations showing evaluation of our proposal. Conclusion and future work will be mentioned in Section V.

II. MOTIVATION

A. Comparisons of Routing Strategies



Fig.1. Single Radio Single Path

1) *SRSP(Single Radio, Single Path)*: Among a great number of routing protocols, it is common to use a single-radio and single-path routing method, e.g. Dynamic Source Routing (DSR) [3], Ad-hoc On demand Distance Vector routing (AODV) [4].

In this case, packets travel along the chain of nodes toward their destinations and all nodes are working with one radio in the same channel, as shown in Figure.1. Successive packets on a single chain may interfere with each other causing channel competition and collision in the MAC layer. Ideally end-to-end throughput could achieve at most $1/3$ of the effective MAC layer data rate, since at one time, among any three continuous nodes only one can make transmission in the same channel.

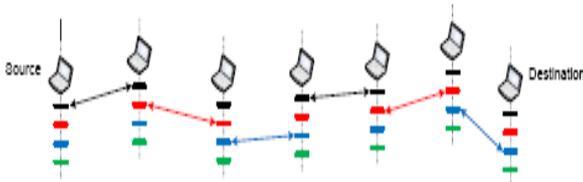


Fig.2. Multiple Radio Single Path

2) *MRSP(Multiple Radio, Single Path)*: Some researchers have proposed multi-channel multi-radio solutions using more channels/radios to receive and send data in different channels simultaneously, such as [5], [6] and [7]. In this scenario an ideal multi-channel multi-radio

routing protocol could help achieve end-to-end throughput almost as high as the effective MAC data rate. Considering the scenario in Figure.2, assume that the MAC protocol can always select an appropriate radio and schedule perfectly. At time slot 1, the first node transmits the first packet to second node on channel 1. At time slot 2, they can transmit at the same time using different radios as well. If radio resources are enough for MAC protocol, every node can continuously inject one packet every time slot.

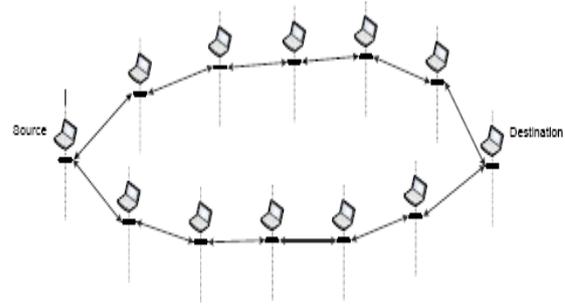


Fig.3.Single Radio Multiple Path.

3) *SRMP(Single Radio, Multiple Path)*: In multi-path routing, packets are split into two or more disjoint paths to destinations, like SMR routing [8], and AOMDV [9]. In Figure.3, there are two paths, each of which performs the same as the SRSP. Hopefully, twice end-to-end throughput can be achieved. However, broadcast nature of wireless medium degrades throughput significantly since all nodes are still working in the same channel, especially the first and last node, which are mostly the bottle neck nodes. Consequently, practically SRMP routing protocols help make limited improvement of throughput.

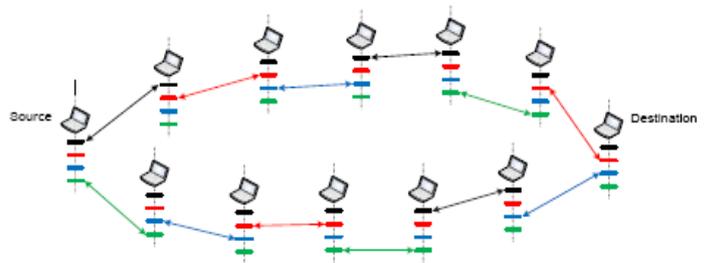


Fig. 4. Multiple Radio Multiple Path

4) *MRMP(Multiple Radio, Multiple Path)*: Ultimately, we show a protocol using a multi-radio MAC protocol combined with multi-path can overcome every issue we mentioned above. In MRMP scenario, the ideal MAC end-to-end throughput can be as high as the effective MAC data rate multiplying the number of paths. Figure.4 shows a case in which a routing protocol can split data flow properly and radios used in transmission are enough and

well assigned by the MAC protocol. At time slot 1 of this case, one node transmits a packet along the upper path to next node on a channel, and simultaneously, another node in the lower path also transmits a packet to its successive node. At time slot 2, still all nodes can keep receiving packets using one radio from previous node, while forwarding to next nodes with the other radio at the same time. Therefore, this achieves an end-to-end throughput twice as full MAC data rate.

B. Related Work

To our knowledge, JMM protocol[10] is the only protocol utilizing multi-channel technique and multi-path strategy in WMNs. JMM divides the time into slots, and coordinates channel usage among slots using a receiver-based channel assignment and schedules transmissions along dual paths. JMM efficiently increases the performance by decomposing contending traffic over different channels, different time, and different paths. However, JMM protocol could not dynamically assign channels upon changeable network status. Also it is centralized algorithm, which requires high amount of controlling messages exchanges inducing high overhead. According to this, we propose a novel routing protocol dynamically utilizing multiple radios of each node and multi-path strategy upon current network status. Every node will select optimal channel and radio based on latest one-hop neighbor information, and the route establishments are in distributed manner.

III. FRAMEWORK OF PROPOSED ROUTING PROTOCOL

Challenge in our framework is how to set up two or more optimal paths with different channel assignment diverse enough to make no intra-path and inter-path influence to improve the throughput of the transmission. Our distributed algorithm for each WMN router will be discussed in detail in this section.

A. Assumptions and Definitions

We assume each mesh router can have maximum N radios working in N orthogonal channels respectively. This is not limited to current standards, which means our proposal can be adapted to future standards easily.

WMN routers connect together as homogeneous networks with bi-directional links, and we define two words to evaluate a path.

- Topology-dimensional Disjoint paths: From the view of network topology, there is no graphical joint node of two paths. In other words, they share no node in common.
- Channel-dimensional Disjoint paths: From the view of channel assignment, if a node is involved in two or more paths of the same flow, it will be assigned to make two radios work in different radios, therefore different

channels, for each path. This means even if paths are graphically joint, but at the joint, different radios are assigned, and there will be no channel competition and interference there.

B. Routing Algorithm

Our scheme mainly bases on origin AODV, a reactive routing protocol which uses a broadcast route discovery mechanism and relies on dynamically establishing route table entries at intermediate nodes. Our approach modifies some bits in the hello messages of AODV protocol in order to make each router know the channel usage status of its neighbors. Also, some bits and frames of RREQ/RREP are changed slightly, and the abandon and rebroadcast mechanisms of the AODV RREQ are changed to meet the need of the new scheme.

Node Id	Relationship	Channel Usage List
4	Myself	00101...
2	CS	11001...
3	CS	11010...
5	CS	00001...
7	CS	01100...
Total Channel Usage Index		23213...

Ch#1	Ch#2	Ch#3	Ch#4	Ch#5	...
0	0	1	0	1	...

1 - Occupied by data flow
0 - Idle

Fig.5. Channel Usage List

1) *Channel Usage List and Modified Tables:* We predefine Channel Usage List showed in Figure.5 which is maintained by each node. It stores the channel usage status of the node itself and its 1-hop neighbors. As the figure shows, node will choose the least channel#4 to rebroadcast the RREQ. Also AODV routing tables are modified a little adding radio info. AODV routing discovery table is used to store path info temporarily to temporarily store the first path info and to help construct the second path.

2) *Path Discovery:* The Path Discovery process is initiated whenever a source node needs to communicate with another node for which it has no routes in its routing table. The source node used flood mechanism such as AODV routing protocol. The modified RREQ carries more information about its path ID. The RREQ flooding is as shown in Figure.6. The first RREQ with path ID=1 will choose the least used channel to broadcast, related bit in routing discovery table will be modified. After a short random period, the second RREQ with path ID=2 will be broadcasted in the second least used channel, so as the following RREQ if more paths are used. This makes sure the second path of same flow won't use the same radio as the first path at this node. Note that our algorithm guarantees channel-dimensional disjoint

characteristic of the paths of same data, by using discovery table, as the first priority. The topology channel dimensional disjoint characteristic of paths of different flows can be guaranteed if radio resource is enough, which means it will be treated as the second priority.

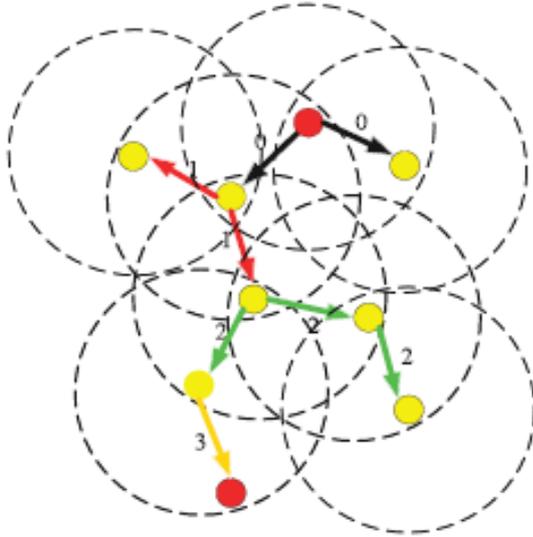


Fig.6. Flooding RREQ

Each neighbor either satisfies the RREQ by sending RREP back to the source, or then rebroadcasting the RREQ to its own neighbors. To choose an optimal channel for rebroadcast is one of the key issues in the proposed scheme. As each node is maintaining its Channel Usage List it clearly knows the channel status of itself and its 1-hop neighbor nodes. Therefore it will select the least used channel to rebroadcast the RREQ.

3) *Reverse Path Setup*: A node records the address of the neighbor from which it received the first copy of the RREQ in case that it will set up a reverse path. These reverse path route entries are maintained for at least enough time for the RREQ to traverse the network and produce a reply to the sender. Besides, the abandon mechanism of RREQ is a bit different from the one in AODV. When an intermediate node receives a RREQ if it has already received a RREQ with the same broadcast id, source address, and also the radio, or the RREQ is from one of its next hop node, it recognize the RREQ as a redundant one and does not rebroadcast but drop it. Algorithm is showed in Figure.7

4) *Forward Path Setup*: Once the destination node received the RREQ it replies the RREP. Then the RREP trace the vectors in each node to reach the source node. This is the way of setting up the forward path from all nodes back to the source. Addition to AODV routing protocol reverse path setup duration is accompanying the channel status table maintenance. Once RREP reach a

node, the node has to broadcast a channel announce message (CAM) which contain the channels it used for receiving and forwarding the RREP to its neighbor nodes. All of the neighbor nodes received the CAM and then use it to update its Channel Usage List.

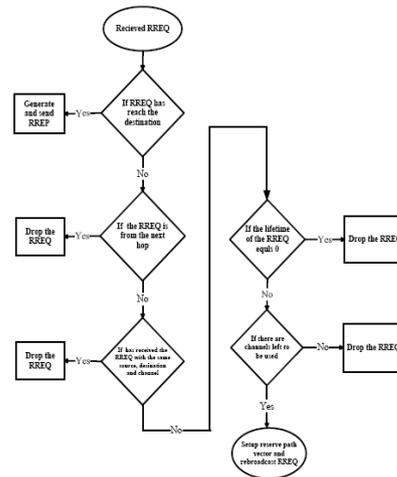


Fig.7. Procedure of RREQ

5) *Multi-Path Maintenance*: After the path initialization, in ideal situation, we can get two or more paths in the wireless mesh network. A special calculation will be done to select two or three of the paths for data transmission simultaneously. Dynamic maintenances based on the quality of WMNs are still in consideration.

6) *Data Transmission*: Data is divided into several concurrent flows, and be transmitted to the destination simultaneously. Currently in dual paths, a packet with odd ID number is sent to path 1 and a packet with even ID number is sent to path 2. Also we can use improved scheme that we assign packets to paths based on the practical bandwidth of each path from feed back, which utilizes network resource better.

IV. SIMULATIONS

A demo based on our algorithm was developed to discover routes in a virtual network. Routers are placed in a grid in where only the four routers around are considered as neighbors.

A. Simulation without Interference Flow

In this evaluation scenario the network starts from an idle state and there is no other concurrent flows.

1) *Path Discovery*: Figure.8 contains some of the paths our demo discovered while the virtual network contains not any other interference flow, which is evaluated in the NS2 platform in the next step. Links with different color mean the different radios, and they show the routing path from the source node to the destination one with specified hop counts.

Origin AODV routing is used in SRSP scenario, and for MRMP Scenario, we use our proposed protocol. In the SRMP scenario, we simply use the same paths as MRMP but all nodes

MRMP has the best throughput performance. It could achieved an about 10 Mb throughput. MRSP could also provide a high throughput at about 5 Mb, a half performance of the MRMP scenario, because it uses only one path to transmit data between the source and destination.

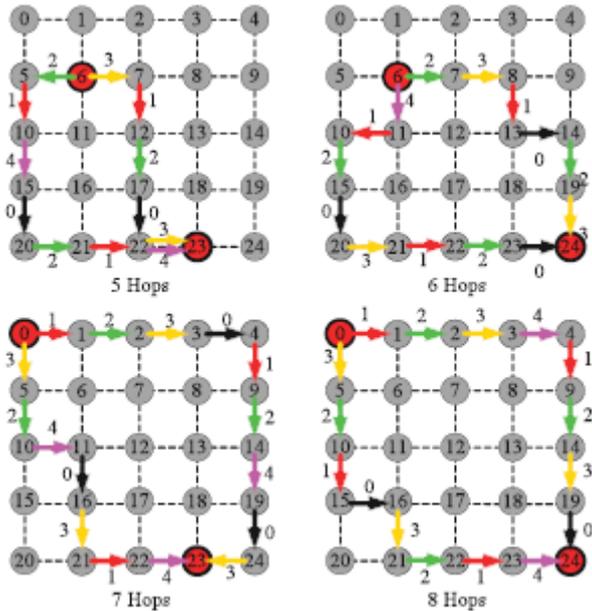


Fig.8. Protocol Demo showing Channel Assignments and Paths.

with only one channel. For the MRSP scenario, we selected the shorter path in MRMP as the only optimal path and same channel assignments are used.

Scenarios with single radio have a poor performance because the channel competition and interferences intra-flow in a single channel scenario will seriously affect the capacity of the throughput. Meanwhile, if the number of the radios is enough for the assignment in multiple channel scenarios, the decreases of the throughput along with the increase of the hops will not impact the throughput and each hop is transmitting at maximum speed. SRSP could only achieve about 1 Mb throughput, which is about 1/10 of the MRMP scenario, and SRMP do not make any enhancement on the throughput comparing to SRSP even if it uses multi-path, since the intra-/inter-path channel competition and interference could not be reduced when single channel is used.

B. Simulation with interference flow

2) Simulation Performance related to Hops: We analyze the details of the four scenarios on the trends of them related to the hops. The tested topology is a 7x7 grid. 802.11a at 6Mbps rate and UDP traffic are used. We utilize 5 channels and radios. Figure.9 compares the throughputs of the four scenarios.

1) Path Discovery: In order to verify whether our proposal still works well if it already exists other flow in the network, as shown in Figure.10, we simulate a interference transmission flow from node 2 to node 22, and then we simulate a data flow from node 6 to node 18. Throughput of each flow by AODV and our proposal are tested.

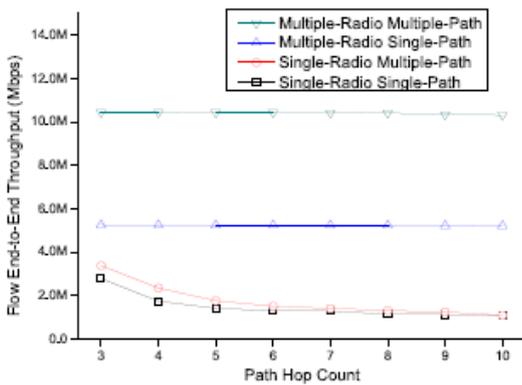


Fig.9. Performance in Idle Network Scenario

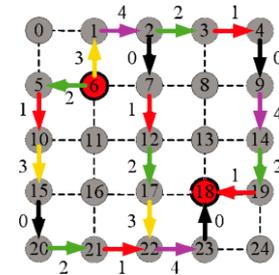


Fig.10.Paths Discovery in Scenario with Concurrent Flow.

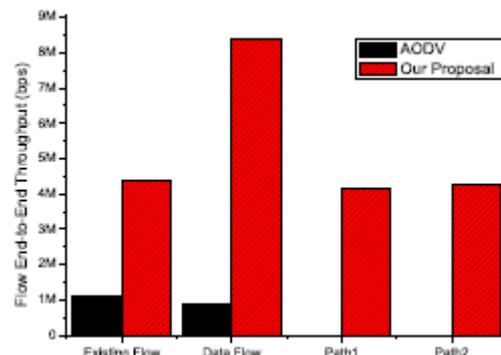


Fig.11. Protocol Performance in Scenario with Concurrent Flow.

2) *The Enhancement of Throughput:* We evaluated this scenario in NS2, comparing with origin AODV. According to the existing flow, channel competition and interference cause performance reduction for origin AODV protocol working in one channel and one path. For this reason, by using AODV, from the Figure.11, the data flow could only achieve 802Kb throughput, and the interference flow can achieve nearly 1Mbps. On the other hand, the throughput of the flow based on our proposal changes little by the interference flow. If our proposed routing protocol is used for data flow and interference flow, both of them can achieve very high throughput. For the data flow, it is divided into two paths, and as we can see path 1 and path 2 can achieve 4Mbps throughput. This shows that our proposal also has a better performance comparing to other scenarios in the network if the radio resources are still enough and they are well allocated. The reason is that the dynamic path establishing mechanism and channel selection of our proposed protocol make routes more flexible when some of the channels in the network are occupied.

V. CONCLUSIONS

In this paper, we proposed a framework for a multi-radio and multi-path routing protocol based on origin AODV for Wireless Mesh Network systems. The protocol can dynamically establish multiple paths with diverse channel assignment, which are topology-dimensional and channel-dimensional disjoint for data transmission, and the routing initialization works in a distributed manner. JAVA demo and NS2 simulations are carried out to evaluate our proposed protocol compared with other routing strategies: SRSP, MRSP and SRMP. Our proposal can make significant enhancement on achievable throughput in WMNs if the network is initially idle, and it performs still better than AODV in scenarios where there is also other concurrent ongoing flow. In future we are planning to make more evaluation on scenarios with heavier traffic and make optimization, because the exhaustion of radio resources will impact our proposal much. Also routing discovery overhead and delay will be relatively worse than other protocols. We will also mathematically model and we hope implement on QualNet4.0 and even real test bed.

ACKNOWLEDGMENT

This research is partly supported by the Foundation of Ubiquitous Computing and Networking(UCN) Project, the Ministry of Knowledge Economy(MKE) 21st Century Frontier R&D Program in Korea and a result of subproject UCN-08B3- B3-10M, and partly supported by the IT R&D program of MKE/IITA [2007-F-038-02, Fundamental Technologies for the Future Internet].

REFERENCES

- [1] I. Akyildiz and X. Wang, "A survey on wireless mesh networks," *IEEE Communications Magazine*, vol. 43, pp. S23–S30, 2005.
- [2] IEEE Standard 802.11, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. IEEE, 1999.
- [3] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad-hoc wireless networks," in *SIGCOMM*, 1996.
- [4] C. E. Perkins and E.M. Royer, "Ad-hoc on-demand distance vector routing," in *WMCSA*, 1999, pp. 90–100.
- [5] R. Draves, J. Padhye, and B. Zill, "Routing in multi-radio multi-hop wireless mesh networks" in *MobiCom*, 2004, pp. 114–128.
- [6] R. A and T. cker Chiueh, "Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network," in *INFOCOM*, vol. 3, 2005, pp. 2223–2234.
- [7] M. X. Gong and S. F. Midkiff, "Distributed channel assignment protocols: A cross-layer approach," in *WCNC*, 2005.
- [8] S. ju Lee and M. Gerla, "Split multipath routing with maximally disjoint paths in ad hoc networks," in *ICC*, 2001.
- [9] M. M.K. and D. S.R., "On-demand multipath distance vector routing for ad hoc networks" in *ICNP*, 2001, pp. 14–23.
- [10] W. H. Tarn and Y. C. Tseng, "Joint multi-channel link layer and multi-path routing design for wireless mesh networks," in *INFOCOM*, 2007, pp. 2081–2089.

A Security Analysis Framework for Dynamic Web Applications

1Selvakumar.R, 2Mohamed Saleem.S

1Senior Lecturer, 2Undergraduate Student,

Department of Computer Science and Engineering,
V.M.K.V.Engineering College, Salem, Tamilnadu, India.

Email: {ashwath.cute, saleemsweetsms}@gmail.com

Abstract This paper proposes a security analysis framework for dynamic web applications. A reverse engineering process is performed over a dynamic web application to extract a rolebased access control security model. A formal analysis is applied on the recovered model to check access control security properties. This framework can be used to verify that a dynamic web application conforms to access control policies specified by a security engineer.

I. INTRODUCTION

Current technologies such as anti-virus software programs and network firewalls provide reasonably secure protection at the host and network levels, but not at the application level. When network and host-level entry points are comparatively secure, public interfaces of web applications become the focus of attacks [26].

In this paper, we focus on one of most serious web application vulnerabilities, broken access control. Access control, sometimes called authorization, governs how web applications grant access to functions and content to some users and not to others [1]. Depending on the access control model, sets of users can be grouped into roles, where privileges are assigned to roles rather than users. This kind of access control model facilitates the administration of user management and is called a Role-Based Access Control model (RBAC) [24]. Broken access control in web applications is considered one of the top ten web application security vulnerabilities [1]. Most web applications try to implement access control policies using obscurity, where links to pages are not presented to unauthorized users. This method of protection is not sufficient because attackers can attempt to access hidden URLs, knowing that sensitive information and functions lie behind these URLs. Attackers also try to access unauthorized objects and resources other than URL pages in an indirect way, for instance, indirect access to back-end resources such as databases.

The consequences of allowing unprotected flows to crafted requests could be very destructive, especially when the web application allows administrators to remotely manage users and contents over the web. In such cases the attackers are not only able to view unauthorized content, but also to take

over site administration. Broken access control is usually caused by an unreliable implementation of access control techniques. In many current web applications, access control policies are spread over the code, which makes the process of understanding and maintaining such rules a difficult if not impossible task [1]. To protect against this attack, access control policies should be based on a strong model that is implemented at all levels of the web application, including both the presentation level and the business level as well. Checking for authorization should be done on every attempt to access secure information, and access control mechanisms should be extensively tested to ensure that there is no way to bypass them [1].

A. State of the Art

Many methods and tools have been proposed to check for attack vulnerabilities in web applications such as SQL injection and cross site scripting [16, 17], but none of them attempts to detect broken access control attacks, either by testing or by model checking. In our previous work [7, 8], we found that many methods propose static models and tools to check static properties of web applications, and some of them try to model and check dynamic features, but none of them is able to check or even model the access control features of web applications.

In general there is little work [19, 9, 14, 3] on UML-based security modeling. The focus of UMLsec [19] is on modeling security issues other than access control, such as data confidentiality and integrity. Basin et al. propose Model Driven Security (MDS) and its tool SecureUML [14] to integrate security models into system models. The authors first specify a secure modeling language for modeling access control requirements as a generalization for RBAC, after which, they embed this language within an extension of UML Class diagrams. The authors of authUML [9] take a step back and focus on analyzing access control requirements before proceeding to the design modeling to ensure consistent, conflict-free and complete requirements. The Ahn and Hu method [3] differs from the above approaches in using standard UML to represent the access control features of the security model. They provide a policy validation based on Object constraint Language (OCL) and Role-based Constraints Language 2000(RCL2000) [4], and

then translate the security model to enforcement code. All of these are forward engineering approaches, while the real need is for a reverse engineering approach that is not only able to model access control policies, but also able to check them in real applications. There is a critical need for an approach that is able to test or model check web applications to ensure that they are protected from broke.

II. RESEARCH APPROACH

Our proposed framework (Figure 1) is aimed at recovering an RBAC security model from dynamic web applications. Based on a formal version of this model, the framework can be used to verify whether a dynamic web application conforms to the access control policies specified by a security engineer, either with a correctness check, or with a counterexample if an access control violation is encountered in the code.

The framework involves two main phases:

1. Static and dynamic reverse engineering of the web application structure and behavior.
2. Security model construction and analysis.

In the following subsections we will outline the entire framework components and the flow of data between them.

A. Web Application Reverse Engineering

In the first phase, static and dynamic analysis of the dynamic web application is used to recover the basic elements of an RBAC model [24]. We need to specify the set of users, roles, resources and their hierarchies, as well as the relations and access policies between them. Extracting static models such as class diagrams and behavioral models such as sequence diagrams help us in this regard.

i. Static Analysis

The static analysis shown in Figure 1(B) extracts class diagrams that help in identifying the set of users, roles, resources and any relations between them. We have proposed and implemented [6] an automated transformation from an SQL (DDL) schema to an open XMI 2.1 UML-adapted class model. The adapted model is a tailored UML class model to represent the basic ER diagram components, including entities, attributes, relations, and primary keys.

Our transformation technique is a novel one in that it is open, nonvendor specific, and targeted at the standard UML 2.1 exchange format, XMI 2.1. Although comparable commercial transformations exist, they are closed technologies targeted at formats tightly coupled to the vendor's tools, hindering portability and preventing users from choosing their preferred tools in the development process. This analysis is supported by a dynamic analysis that may refine the class diagram, as well as recover behavioral models.

ii. Dynamic Analysis

Static analysis is not adequate because it does not take into account the runtime behavior of web applications. Dynamic analysis is required to perform a full security analysis, including tracking user sessions, cookies, and user inputs. To recover the implicit permissions from dynamic web

applications, we have proposed and implemented an approach and tool [5] to automatically instrument dynamic web applications using source transformation technology [13], and to recover a sequence diagram from execution traces generated by the resulting instrumentation, Figure 1(A).

Using an SQL database to store generated execution traces, our approach automatically filters traces to reduce redundant information that may complicate program understanding. The elements in the sequence diagram are the interactive user and browser session, the Application Server, and the application pages and entities. The messages between these elements represent page transitions and how they affect the application entities, either with read or write operations. While our current implementation supports all versions of the PHP scripting language, the framework is not tied to any particular language and can be extended in plug-and-play fashion to other scripting languages.

Our proposed framework will address code coverage by augmenting the dynamic analysis with instrumentation for code coverage, combined with a mutation approach like that of Bellettini et al. [10] for flow coverage. This will decrease the percentage of false positives due to an analysis that results in a model that only partially covers the code (leading to verifications of properties that may in fact not hold).

Even using code and flow coverage methods, enumerating all execution paths is difficult. Ideally our framework should be able to identify all execution paths, but in some cases the human factor may be unavoidable, for instance when valid or critical information is needed to fill forms, user names or passwords. Like web security scanning tools such as Veri-Web [20] and AppScan [18], we may adopt a profile-based solution which requires administrators to manually supply valid values for form fields.

B. Model Construction and Analysis

In this phase a UML-based security model is constructed based on the Basin et al. [14] security meta-model (SecureUML). A transformation from this model to a statebased formal analysis model is then performed to ease the process of security analysis and verification.

i. RBAC-Model Construction

The core part of the proposed framework is the security model. In order to be able to check the web application's access-control security properties, the framework must be based on a strong security model, and be able to extract it from the source code. We construct our security model using a Role-Based Access Control (RBAC) approach, Figure 1(C). Since users are not assigned permissions directly, but rather acquire them through their role (or roles), management of individual user rights is simplified. In a role-based model, permissions for common operations such as adding a user or changing a user's department become obvious.

Our RBAC model is constructed by binding the recovered application ER model [6] with the recovered dynamic behavioral model. The recovered sequence diagram is generated based on execution traces collected from the dynamic analysis part of our framework [5]. Web crawling

tools that mimic user interactions with web applications, such as clicking links, filling in forms and pressing buttons [15, 25] are used to automate collecting traces, while the application roles themselves are recovered manually by studying the software documentation.

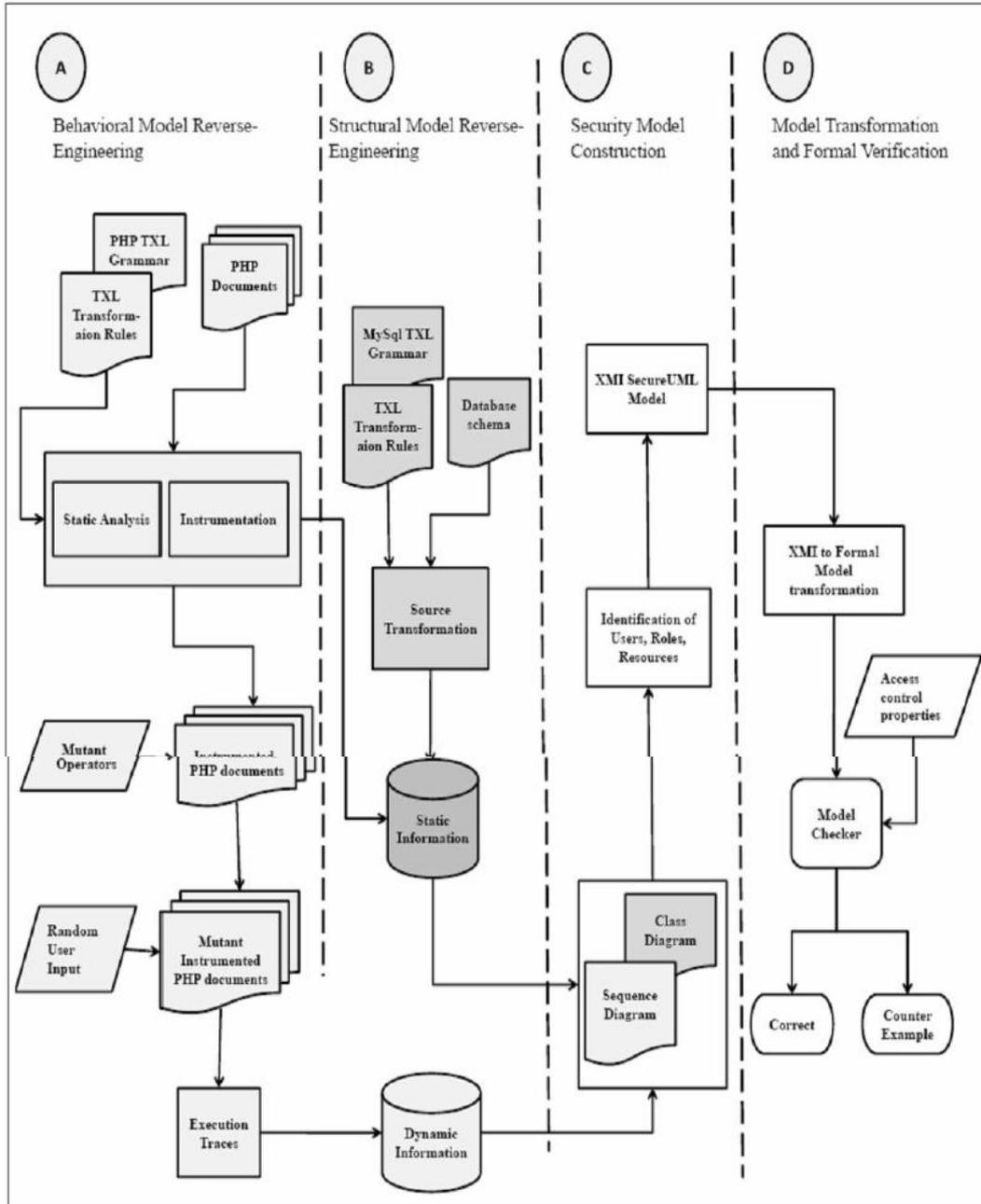


Figure.1. The Proposed Framework

The generated sequence diagrams are combined into one single sequence diagram for the entire application in XMI 2.1 format, which is then combined with the application XMI 2.1 form of the ER model recovered by the static analysis part of our framework [6], using Model Driven Security (MDS) [14] to automatically generate a SecureUML model for the web application.

ii. Model Transformation and Formal Verification

Once the SecureUML model is constructed, we need to analyze it against the security properties (Figure 1(D)). While UML models provide good support for verifying web application requirements, they need to be converted into a formal state model in order to be automatically checked [7, 8]. several methods in the literature propose tools for the translation from UML diagrams to formal state models that can be checked using existing formal verification tools.

Examples are UML2Alloy [11] and XMI2SMV [12]. We convert our SecureUML model to a formal state model using a similar conversion process. The formal model along with the desired security properties is fed to a formal verification tool such as Alloy, yielding either confirmation that the properties hold, or a counter-example. When a counter-example is generated, the problem is mapped back to the code at the function point level by tracing back to the violated dynamic page. In some cases it may be possible to go deeper, for example using the parameters provided in the URL to identify the block of code causing the violation.

III. EVALUATION AND PRELIMINARY RESULTS

Our approach will be validated on a number of different web applications. Good candidate systems to assess our approach are web applications that are open source, and built using the combination of Apache server, PHP, and MySQL. The proposed framework will be applicable to other technologies as well, simply by adding their grammars to the static analysis and instrumentation stages. The most important requirement is that the web application should have some kind of permission system.

Because our approach is based on static and dynamic analysis, we require source code. Our choice of the combination of PHP, MySQL, and Apache server is based on the popularity of these technologies. According to (Netcraft) [21], Apache web server is the most deployed web server on the internet with a 58.7% market share. PHP has been the most popular server-side scripting language for years and is likely to remain so for some time. As of April 2007, there were more than 20 million websites (domain names) using PHP [22]. MySQL as well is the fastestgrowing database in the industry, with more than 10 million active installations and 50,000 daily downloads [2]. The approach could be applied to other technologies as well.

In our first experiment, we are applying the proposed approach to the PhpBB [23] web application. PhpBB is the world's leading open source forum software. It has a powerful permission system and a number of other key features such as private messaging, search functions, a customizable template and language system, and support for multiple database technologies. So far we have evaluated our prototype tools, SQL2XMI [6] and PHP2XMI [5], on PhpBB 2.0. SQL2XMI is able to automatically reverse engineer an ER class model from the PhpBB source, and PHP2XMI is able to automatically reverse engineer two kinds of sequence diagrams from PhpBB, one that represents the basic page transitions for each role, and a more detailed version that shows the effect of each page transition on the application entities recovered by SQL2XMI based on dynamic read and write operations.

IV. CONCLUSION

The proposed approach is a novel one in web application security verification. Besides being the first approach to tackle the issue of access control verification, the proposed framework is flexible enough to allow for different server side technologies and databases in plug and play fashion. Our approach also yields the potential for application in systems other than web applications. The static and dynamic reverse-engineering front-end of the framework can be reused for other kinds of analysis, and the framework could be used to discover other kinds of security attacks, such as cross-site scripting and SQL injection.

In our first experiment, the framework is being evaluated on one of the most popular PHP web applications, PhpBB, to check that the application is free from any remaining access control vulnerabilities.

REFERENCES

- [1] The Top Ten Most Critical Web Application Security Vulnerabilities, <http://www.owasp.org/documentation/topten>, last access June 27, 2007.
- [2] MySQL AB, MySQL Market Share <http://www.mysql.com/why-mysql/marketshare/>, last access Nov 26, 2008.
- [3] Gail-Joon Ahn and Hongxin Hu. Towards realizing a formal RBAC model in real systems. In SACMAT 2007, 12th ACM Symposium on Access Control Models and Technologies, Sophia Antipolis, France, June 20-22, 2007, pages 215–224.
- [4] Gail-Joon Ahn and Ravi S. Sandhu. Role-based authorization constraints specification. *ACM Trans. Inf. Syst. Secur.*, 3(4):207–226, 2000.
- [5] Manar H. Alalfi, James R. Cordy, and Thomas R. Dean. "Automated Reverse Engineering of UML Sequence Diagrams for Dynamic Web Applications". In *WebTest 2009, 1st International Workshop on Web Testing*, Denver, Colorado - USA April 4, 2009 (in press).
- [6] Manar H. Alalfi, James R. Cordy, and Thomas R. Dean. SQL2XMI: Reverse Engineering of UML-ER Diagrams from Relational Database Schemas. In *WCRE 2008, the 15th Working Conference on Reverse Engineering*, Antwerp, Belgium, October 15-18, pages 187–191.
- [7] Manar H. Alalfi, James R. Cordy, and Thomas R. Dean. A Survey of Analysis Models and Methods in Website Verification and Testing. In *ICWE 2007, 7th International Conference on Web Engineering*, Como, Italy, pages 306–311, 2007.
- [8] Manar H. Alalfi, James R. Cordy, and Thomas R. Dean. Modeling methods for web application verification and testing: State of the art. *Softw. Test., Verif. Reliab.*, 2008 (in press).
- [9] Khaled Alghathbar and Duminda Wijesekera. authUML: a three-phased framework to analyze access control specifications in use cases. In *FMSE 2003, ACM workshop on Formal methods in security engineering*, FMSE 2003,

Washington, DC, USA, October 30, pages 77–86.

[10] Carlo Bellettini, Alessandro Marchetto, and Andrea Trentini. WebUml: reverse engineering of web applications. In SAC 2004, ACM Symposium on Applied Computing, Nicosia, Cyprus, March 14-17, 2004, pages 1662–1669.

[11] Behzad Bordbar and Kyriakos Anastasakis. MDA and Analysis of Web Applications. In TEAA(2005), Trends in Enterprise Application Architecture, VLDB Workshop, Trondheim, Norway,, volume 3888 of LNCS, pages 44–55. Springer.

[12] Daniela Castelluccia, Marina Mongiello, Michele Ruta, and Rodolfo Totaro. WVer: A Model Checking-based Tool to Verify Web Application Design. *Electr. Notes Theor. Comput. Sci.*, 157(1):61–76, 2006.

[13] James R. Cordy. The TXL source transformation language. *Sci. Comput. Program.*, 61(3):190–210, 2006.

[14] D.Basin, J.Doser, and T. Lodderstedt. Model driven security: from UML models to access control infrastructures. *ACM Trans. Softw. Eng. Methodol.*, 15(1):39–91, 01 2006.

[15] Canoo Engineering. Canoo WebTest, <http://webtest.canoo.com>.

[16] Yao-Wen Huang, Chung-Hung Tsai, Tsung-Po Lin, Shih-Kun Huang, D. T. Lee, and S. Y. Kuo. A testing framework for Web application security assessment. *Computer Networks*, 48(5):739–761, 08 2005.

[17] Yao-Wen Huang, Fang Yu, Christian Hang, Chung-Hung Tsai, Der-Tsai Lee, and Sy-Yen Kuo. Securing web

application code by static analysis and runtime protection. In the 13th international conference on World Wide Web, WWW 2004, New York, NY, USA, May 17-20, pages 40–52, 2004.

[18] Sanctum Inc. Web Application Security Testing, AppScan 3.5., <http://www.sanctuminc.com>, last access September 5, 2007.

[19] Daniel Jackson. *Software Abstractions: Logic, Language, and Analysis*. MIT Press. Cambridge, MA., March 2006.

[20] B. Michael, F. Juliana, and G. Patrice. Veriweb: automatically testing dynamic web sites. In WWW 2002, the International World Wide Web Conferences, Honolulu.

[21] Netcraft Ltd. November 2008 web server survey, http://news.netcraft.com/archives/2008/11/19/november_2008_web_server_survey.html, last access Nov 26, 2008.

[22] PHP Group. PHP usage Stats for April 2007, <http://www.php.net/usage.php>, last access June 27, 2007.

[23] phpBB Group. PhpBB, <http://www.phpbb.com/>, last access June 27, 2007.

[24] R. S.Sandhu, E. J.Coyne, H. L.Feinstein, and C. E.Youman. Role-based access control models. *Computer*, 29(2):38, February 1996.

[25] WatirCraft. WATIR, <http://wtr.rubyforge.org>.

[26] Adrian Wiesmann, Andrew van der Stock, and Mark. *A Guide to Building Secure Web Applications and Web Services*. Open Web Application Security Project, OWASP, 2005.

Analysis of Knowledge Management Tools

Muhammad Bilal Qureshi,
Department of Computer Science,
Faculty of Basic & Applied Sciences,
International Islamic University
Islamabad, Pakistan
muhdbilal.qureshi@gmail.com

Abstract-This paper describes experiences in applying knowledge management tools, that are ISYSDesktop8Setup, kmAnywhere 2005 Pro and junior v3.6. This paper describes the performance analysis of three knowledge management tools (KMT) in term of their storage capabilities and their performance on the basis of searching capabilities. From the outlook all three KMT's are meant for large organizations and form a single platform basis for information gathering and management. The distinctive features of these tools are their graphical user interface (GUI) facilities like contacts, calendar, notes, lectures and data storage etc.

Keywords:

Knowledge Management, Searching, Analysis, GUI, Indexing, File format.

I. INTRODUCTION

An effective knowledge management is an asset to any organization .Because it can greatly enhance the returns and reputation of an organization. Knowledge management helps the organization to find, select, organize, disseminate and transfer important information and expertise. This has led to a rise in the number of knowledge management tools available on the software market. So wide variety of knowledge management tools are available. However, the wide range of choice can make it difficult for an organization to select a tool (Ingie Holland, 2003) that suitably meets their requirements.

In this paper we evaluate the above three knowledge management tools. These tools are analyzed based on search criteria.

II. WHAT IS KNOWLEDGE MANAGEMENT?

For the purpose of introduction it is useful to differentiate between raw information and knowledge (Edwards, 1994). Raw information may be widely available to a number of agencies (Brian Newman, Kurt W.Conrad, 1999), but only some organizations will be able to convert the information into relevant knowledge and to use this knowledge to achieve their aims. There are two strategies first and second generation KM strategies. The first generation strategy relies on organizing and controlling the existing knowledge and knowledge sharing within organization, the second generation KM strategies have shifted towards enhancing

Muhammad Shuaib Qureshi
Department of Computer Science,
Faculty of Basic & Applied Sciences,
International Islamic University
Islamabad, Pakistan
qureshi.shuaib@gmail.com
the conditions for innovation and knowledge creation (McElroy, 2000).

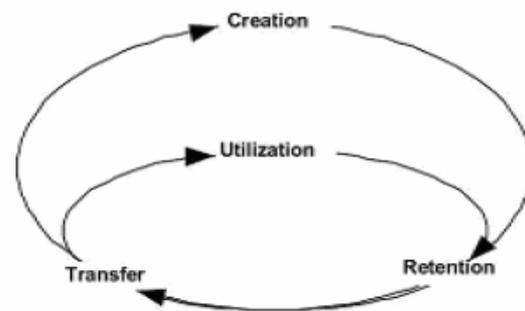


Figure 1. The General Knowledge Model

Knowledge Creation:

This stage accumulates all the activities related to the addition of new knowledge to the system including knowledge development, discovery and capture.

Knowledge Retention:

This stage incorporates all the activities that retain knowledge and permit it to remain in the system once introduced. It also contains those activities that preserve the viability of knowledge within the system.

Knowledge Transfer:

This stage holds all the activities that are related to the transfer of knowledge from one party to another. This process comprises communication, translation, conversion, filtering and rendering.

Knowledge Utilization:

This stage includes all the events concerned with the application of knowledge to business processes (Nayna PateIVlatka Hlupic, 2002).

III. KNOWLEDGE MANAGEMENT TOOLS

Knowledge management tool is software used for planning knowledge management projects which promote sharing and use of knowledge such as ideas, expertise, and best practices (www.content-management-junction.com, Jan 4, 2008).

1) *Why to use knowledge management tools?*

Almost eighty percent of an organization's business content is unstructured. The unstructured business content include information in files, messages, memos, reports, and proposals created in different formats and stored in many locations. The vast amounts of information need to be put in context to be filtered through and be made available to those (www.content-managementjunction.com, Jan 4, 2008) who need it in a format that they want. This enables you to realize the full value of organizational knowledge assets. The Knowledge management tools help enables organizations to tackle all the problems related to knowledge management more effectively at reduced costs. In addition, the Knowledge management tools allow them to leverage the collective knowledge and experience of an organization to accelerate innovation and sharpen competitive advantage. The Knowledge management tools dramatically improve the way organizations manage their knowledge assets. The Knowledge management tools allow user to conduct single, unified searches across multiple unstructured information sources which include websites, file repositories, document management systems, multimedia libraries, etc.

2) *Different knowledge management tools*

There are many knowledge management tools available in the market, but we have selected the following three tools,

1. ISYSDesktop8Setup
2. KmAnywhere 2005 Pro
3. Junior v3.6

IV. FEATURES OF SELECTED TOOLS

1) *ISYSDesktop8Setup*

Retrieval: With ISYSDesktop8Setup, you can preview a result's contents to ensure it's the right one. Preview features include a built-in preview pane, complete with hit highlighting and hit-to-hit navigation; a Tool Tip window, which displays relevant document information and meta data; and Outline Browse, which shows only the relevant portions of a document with your search terms in context.

Display: ISYSDesktop8Setup can display your documents in the ISYS Browser, or let you view them in their original applications, if they are available. You can annotate documents with virtual notes and attach photos, video or sound files, all without altering the original document.

Indexing: ISYSDesktop8Setup recognizes and indexes more than 150 different file formats. You select the computers, folders and files to be incorporated into your index, as well as specify where the index is located, when updates will occur and how frequently.

Searching: ISYSDesktop8Setup supports both simple and sophisticated searching mechanisms to enable you to find the precise document you need as quickly as possible. Search methods include Natural Language Query, Fielded, Phrase Matching and our popular Menu-Assisted Query, for

easily constructing advanced Boolean and proximity queries.

Navigation: ISYSDesktop8Setup automatically categorizes your documents as they are indexed, giving your results more contexts, and allowing users to navigate and refine a results list with a simple click of the mouse.

Discovery: ISYSDesktop8Setup enables users to better understand the who, what and where of a given result list, thanks to ISYS Entities. The only desktop search application offering this feature, ISYS: desktop automatically extracts entities such as names, places, email addresses and more.

2) *KmAnywhere 2005 Pro*

Retrieval: With KmAnywhere 2005 Pro, you can preview a result's contents to ensure it's the right one. Preview features include a builtin preview pane; it only shows those documents which have your search name given to it do not search with in the document.

Display: KmAnywhere 2005 Pro, can display your documents in the Browser, or let you view them in their original applications, if they are available. You can only add images and DOC or PDF files to it.

Searching: KmAnywhere 2005 Pro, has just one type of searching capability. It has a browser showing the options through which we can select either to search from notebooks, emails, contacts or time sheet. It is powered by Essential Skills Consultants.

Navigation: It is achieved through the Navigation pane on the left side of the main GUI which has all the main functions that can be performed and a Start button to get things running in case if you're a beginner.

3) *Junior v3.6*

Retrieval: With junior v3.6, you can preview a result's contents to ensure it's the right one. Preview features include a built-in preview pane; it has the capability of searching within the documents and can find even if a single word is there in any document it'll show it.

Display: The main interface of the junior is not that user friendly as it tends to get hard when you are looking for something. It only adds notepad or plain text files and can cross reference with another document in the same database.

Searching: junior v3.6 has just one type of searching capability. It has a small window which appears at the bottom of the main window and there you can enter the text which you want to search.

Navigation: It is achieved through the tool bar on the top of the main GUI which has all the main functions that can be performed.

V. RELATED WORK

From the initial observation the authors have observed that all these three tools are reasonably good and very flexible in terms of usage and maintenance but a detailed review of these tools show that only one is better than the rest two and that is ISYSDesktop8Setup. For the reason that junior v3.6 is not user friendly and KmAnywhere 2005 Pro also has only file name searching capability not an inside file searching capability. ISYSDesktop8Setup actually first indexes all the files in a manner that user first specifies a target location for indexing and then it is stored with in ISYSDesktop8Setup and whenever a search is given, it shows each file name and every possible combination in which the search can occur. In KmAnywhere 2005 Pro only the file name is searched which can be either in the notes that are specified or time sheet or in the contacts. junior v3.6 has a very simple and straightforward mechanism of searching as there is a small window and there one can enter the search criteria and by pressing ALT+ page up or ALT+ page down can scroll through the records having the given word or phrase. The junior saves all the text files and the related data in a database file, which is hard for normal user to identify that currently which database file needs to be loaded into the junior and which files need to be searched. While in KmAnywhere 2005 Pro, there is concept of notes and contacts and each is stored under the same context. If a contact is added, only that contact is saved under that tab and if a DOC or a PDF file is stored in the KmAnywhere 2005 Pro, it is saved under the notes tab. KmAnywhere 2005 Pro has a very user-friendly environment where one has ease in moving around, searching and adding things in the knowledge base. KmAnywhere 2005 Pro has a windows based approach that each time you open a contact or a search, it is opened in a separate tab making it easy to look back into the stuff and the search results for the given data that has been added to it. On exiting the application, it asks the user to save all the related data which is not been saved yet.

VI. CONCLUSION

From the analysis of the above tools the authors have reached to the conclusion that ISYSDesktop8Setup is the best among these three tools in case of searching capabilities and the manner in which it shows the search results as compared to the rest two. It searches the whole data and shows all the relevant results by highlighting the required results throughout the document and indexes all the files on its database which enables it fast to search and produce accurate results. Extreme clarity in the results is shown and very easy to navigate.

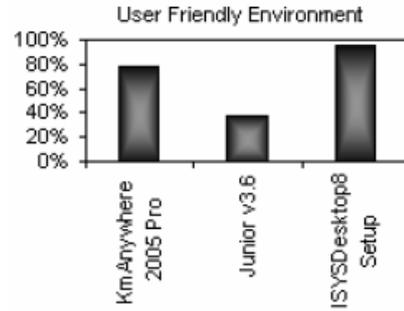


Figure2. Comparison by using User Friendly Environment Feature

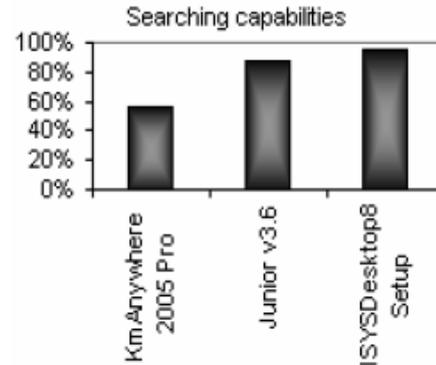


Figure3. Comparison by using Searching Capabilities Feature

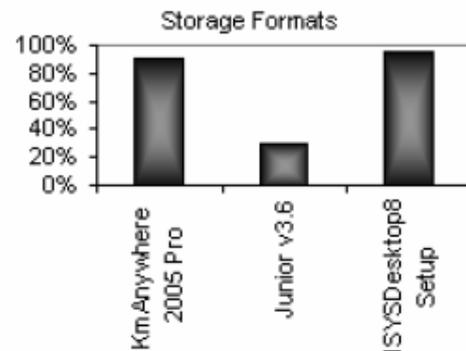


Figure4. Comparison by using Storage Formats Feature

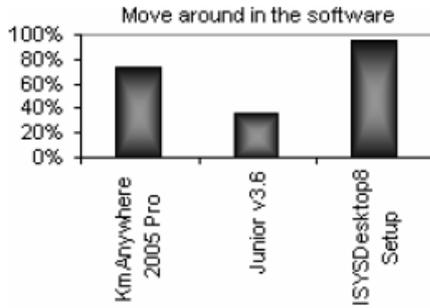


Figure5. Comparison by using Move around the Software Feature

VII. FUTURE WORK

The analysis of these tools leads to the conclusion that one tool is better than the other two and the authors observe that all the tools are not semantically correct. So there is a need to develop a tool that is semantically correct which should provide the best and most desirable results.

REFERENCES

1. Ingie Holland, August 2003, Knowledge Management and Organizational Learning, An Annotated Bibliography, Overseas Development Institute 111 Westminster Bridge Road London, SE1 7JD, UK
2. Brian Newman, Kurt W. Conrad, 1999, A Framework for Characterizing Knowledge Management Methods, Practices, and Technologies, George Washington University.
3. Nayna Patel Vlatka Hlupic. A Methodology for the Selection of Knowledge Management (KM) Tools Information Technology Interfaces, 2002. ITI 2002. Proceedings of the 24th International Conference on Information Technology.
4. Benefits of Knowledge Management Tool. Content Management Junction, www.contentmanagement-junction.com, Jan 4, 2008.
5. Marchand, D. & Davenport, T. 2004. Dominando a gestao da informacao [Mastering information management]. Porto Alegre: Bookman.

Fellows

FICCT (FELLOW OF INTERNATIONAL CONGRESS OF COMPUTER SCIENCE AND TECHNOLOGY)

- 'FICCT' title will be awarded to the person/institution after approval of Editor-in-Chief and Editorial Board. The FICCT can be used following the name. **e.g. Dr. Andrew Knoll, Ph.D., FICCT**
- **Free unlimited Webspace** will be allotted to 'FICCT' along with subDomain to contribute and partake in our activities.
- A **professional email address** will be allotted free with unlimited email space.
- FICCT will be authorized to receive e-Journals and Printed Journal-GJCST for the Lifetime.
- FICCT can send papers for publication without any charges. The paper will be sent to two peer reviewers. The papers will be published after the acceptance of peer reviewers and Editorial Board.
- FICCT will be exempted from the registration fees of Seminar/Symposium/Conference/Workshop conducted internationally of GJCST.
- FICCT will be Honorable Guest of any gathering held.

AICCT (ASSOCIATE OF INTERNATIONAL CONGRESS OF COMPUTER SCIENCE AND TECHNOLOGY)

- Title, AICCT will be awarded to the person/institution after approval of Editor-in-Chief and Editorial Board. The 'AICCT' title can be used following the name. **e.g. Dr. Thomas Knoll, Ph.D., AICCT**
- **Free 2GB Webspace** will be allotted to 'FICCT' along with subDomain to contribute and participate in our activities.
- A **professional email address** will be allotted with free 1GB email space.
- AICCT will be authorized to receive e-Journal GJCST for lifetime.
- AICCT can send two papers per annum for publication without any charges. The papers will be sent to two peer reviewers. The papers will be published after the acceptance of peer reviewers and Editorial Board.

Auxiliary Memberships

ANNUAL MEMBER

- Annual Member will be authorized to receive **e-Journal GJCST for one year.**
- The member will be allotted **free 1 GB Webspace along with subDomain** to contribute and participate in our activities.
- **A professional email address** will be allotted free 500 MB email space.
- Annual Member **can send one paper for publication** without any charges. The paper will be sent to two experts. The paper will be published after the acceptance of peer reviewers and Editorial Board.

PAPER PUBLICATION

- The members can publish paper once. The paper will be sent to two peer reviewer. The paper will be published after the acceptance of peer reviewers and Editorial Board.

Process of submission of Research Paper

The Area or field of specialization may or may not be of any category as mentioned in Abstracting/indexing page of this website. Authors should prefer the mentioned categories. The major advantage of this is that, their research work shall be exposed to all over the world as we are being abstracted and indexed worldwide.

The paper should be in proper format. For sample paper, please refer Guideline menu of this website and for details, please refer Author Guidelines page of this website. The Authors should follow the general rules as mentioned on this page.

The authors should prefer online submission. The paper should be in MS-Word Format (*.DOC) or in Adobe Portable Document format (*.PDF). The file prepared can be uploaded (by online submission process), which is a very easy process. One can follow online submission page of our Website **www.ComputerResearch.org**

1. You must first fill and Submit Signup form.

2. After Guided Registration you will be automatically redirected to the Online Submission Page.

Registration and Uploading of Paper is confirmed by immediate emails. If you are not getting any email from us then please contact us at webmaster@computerresearch.org. These all process goes through “Online Submission” Menu.

We welcome all the Honorable fellows and members of GJCST. All the Authors and co-Authors are supposed to join at least one category as mentioned in Fellows page.

Research Paper will be delivered to our Editorial Board and Peer Reviewers for the review.

You will be contacted as soon as processing the paper is completed.

Preferred Author Guidelines

MANUSCRIT STYLE INSTRUCTION (Must be Strictly followed):

1. Page Size: 8.27 X 11
 - a. Left Margin: 0.65
 - b. Right Margin: 0.65
 - c. Top Margin: 0.75
 - d. Bottom Margin: 0.75
2. Font type of all text should be Times New Roman.
3. Paper Title should be of Font Size 24 with one Column section.
4. Author Name in Font Size of 11 with one column as of Title.
5. Abstract Font size of 9 Bold, "Abstract" word in Italic Bold.
6. Main Text: Font size 10 with justified two columns section
 - a. Two Column with Equal Column with of 3.38 and Gapping of .2
 - b. First Character must be two lines Drop capped.
 - c. Paragraph before Spacing of 1 pt and After of 0 pt.
 - d. Line Spacing of 1 pt
7. Large Images must be in One Column
 - a. Numbering of First Main Headings (Heading 1) must be in Roman Letters, Capital Letter, and Font Size of 10.
 - b. Numbering of Second Main Headings (Heading 2) must be in Alphabets, Italic, and Font Size of 10.

Author Guidelines:

1. General,
2. Ethical Guidelines,
3. Submission of Manuscripts,
4. Manuscript Types Accepted,
5. Manuscript Format and Structure
6. After Acceptance.

1. GENERAL

Before submitting your research paper, one is advised to go through the details as mentioned in following heads. It will be beneficial, while peer reviewer justify your paper for publication.

Scope

The GJCST welcomes the submission of original paper, review paper, survey article relevant to the computer science and technology. The choice of specific field can be done otherwise as following in Abstracting and Indexing Page on this Website. As the GJCST is being abstracted and indexed(in process) by most of the reputed organizations. Topics of only narrow interest will not be accepted unless they have wider potential or consequences.

2. ETHICAL GUIDELINES

Authors should follow the ethical guidelines as mentioned below for publication of research paper and research activities.

Papers are accepted on strict understanding that the material in whole or in part has not been, nor is being, considered for publication elsewhere. If the paper once accepted by GJCST and Editorial Board, will become the *copyright of the GJCST and of Authors of Paper*.

Authorship: The authors and coauthors should have active contribution to conception design, analysis and interpretation of findings. They should critically review the contents

and drafting of the paper. All should approve the final version of the paper before submission

The GJCST follows the definition of authorship set up by the Global Academy of Research and Development. According to the Global Academy of R&D authorship criteria must be based on:

- 1) Substantial contributions to conception and acquisition of data, analysis and interpretation of the findings.
- 2) Drafting the paper and revising it critically regarding important academic content.
- 3) Final approval of the version of the paper to be published.

All authors should have been credited according to their appropriate contribution in research activity and preparing paper. Contributors who do not match the criteria as authors may be mentioned under Acknowledgement. Acknowledgements: Contributors to the research other than authors credited should be mentioned under acknowledgement. The specifications of the source of funding for the research if appropriate can be included. Suppliers of resources may be mentioned along with address.

Appeal of Decision: The Editorial Board's decision on publication of the paper is final and cannot be appealed elsewhere.

Permissions: It is the author's responsibility to have prior permission if all or parts of earlier published illustrations are used in this paper.

Please mention proper reference and appropriate acknowledgements wherever expected.

If all or parts of previously published illustrations are used, permission must be taken from the copyright holder concerned. It is the author's responsibility to take these in writing.

Approval for reproduction/modification of any information (including figures and tables) published elsewhere must be obtained by the authors/copyright holders before submission of the manuscript. Contributors (Authors) are responsible for any copyright fee involved.

3. SUBMISSION OF MANUSCRIPTS

Manuscripts should be uploaded via this online submission page. The online submission is most efficient method of submission of papers, which enables rapid distribution of manuscripts and consequentially speeds up the review procedure. It also enables authors to know the status of their own manuscripts by emailing us. Complete instructions for submitting a paper is available below.

Manuscript submission is a step-by-step procedure and little preparation is required beyond having all parts of your manuscript in a given format and a computer with an Internet connection and a Web browser. Full help and instructions are provided on-screen. As an author, you will be prompted for login and manuscript details as Field of Paper and then to upload your manuscript file(s) according to the instructions.

To avoid postal delays, all association is preferred by e-mail. A finished manuscript submission is confirmed by e-mail immediately and your paper enters the editorial process with no postal delays. When a conclusion is made about the publication of your paper by our Editorial Board, revisions can be submitted online with the same procedure, with an occasion to view and respond to all comments.

Complete support for both authors and co-author is provided.

4 MANUSCRIPT'S CATEGORY: on the basis of potential and nature, the manuscript can be categorized under the following heads:

Original research paper: Such papers are basically reports of high level significant original research work.

Review papers: These are concise, significant but helpful and decisive topics for young researchers.

Research articles: These are handled with small investigation and applications

Research letters: The letters are small concise comments on previously published matters

5 STRUCTURE AND FORMAT OF MANUSCRIPT: the recommend size of original research paper is less than seven thousand words, review papers fewer than seven thousands words also. The research articles and research letters should be fewer than three thousand words, the structure original research paper, sometime review paper should be as follows:

Papers: These are reports of significant research (typically less than 7000 words equivalent, including tables, figures, references), and comprise:
(a) *Title* should be relevant and commensurate the theme of the paper .

(b) A brief Summary, “*Abstract*” (less than 150 words) containing the major results and conclusions.

(c) Up to *ten keywords*, that precisely identify the paper's subject, purpose and focus.

(d) An *Introduction*, giving necessary background excluding subheadings; objectives must be clearly declared.

(e) Resources and techniques with sufficient complete experimental details (wherever possible by reference) to permit repetition; sources of information must be given and numerical methods must be specified by reference, unless non-standard.

(f) Results should be presented concisely, by well-designed tables and/or figures; the same data may not be used in both; suitable statistical data should be given. All data must be obtained with attention to numerical detail in the planning stage. As reproduced design has been recognized to be important to experiments for a considerable time, the Editor has decided that any paper that appears not to have adequate numerical treatments of the data will be returned un-refereed;

(g) Discussion should cover the implications and consequences, not just recapitulating the results; *conclusions* should be summarizing.

(h) Brief Acknowledgements.

(i) References in the proper form.

Authors should very cautiously consider the preparation of papers to ensure that they communicate efficiently. Papers are much more likely to be accepted, if they are cautiously designed and laid out, contain few or no errors, are summarizing, and be conventional to the approach and instructions. They will in addition, be published with much less delays than those that require much technical and editorial correction.

The Editorial Board reserves the right to make literary corrections and to make suggestions to improve brevity.

It is vital, that authors take care in submitting a manuscript that is written in simple language and adheres to published guidelines.

Format

Language: The language of publication is UK English. Authors, for whom English is a second language, must have their manuscript efficiently edited by an English speaking person before submission to make sure that, the English is of high excellence. It is preferable, that manuscripts should be professionally edited.

Standard Usage, Abbreviations and Units: Spelling and hyphenation should be conventional to The Concise Oxford English Dictionary. Statistics and measurements should at all times be given in figures, e.g. 16 min, except for when the number begins a

sentence. When the number does not refer to a unit of measurement it should be spelt in full unless, it is 160 or greater.

Abbreviations supposed to be used carefully. The abbreviated name or expression is supposed to be cited in full at first usage, followed by the conventional abbreviation in parentheses.

Metric SI units are supposed to generally be used excluding where they conflict with current practice or are confusing. For illustration, 1.4 l rather than $1.4 \times 10^{-3} \text{ m}^3$, or 4 mm somewhat than $4 \times 10^{-3} \text{ m}$. Chemical formula and solutions must identify the form used, e.g. anhydrous or hydrated, and the concentration must be in clearly defined units. Common species names should be followed by underlines at the first mention. For following use the generic name should be constricted to a single letter, if it is clear.

Structure

All manuscripts submitted to GJCST, ought to include:

Title: The title page must carry an instructive title that reflects the content, a running title (less than 45 characters together with spaces), names of the authors and co-authors, and the place(s) wherever the work was carried out. The full postal address in addition with the e-mail address of related author must be given. Up to eleven keywords or very brief phrases have to be given to help data retrieval, mining and indexing.

Abstract, used in Original Papers and Reviews:

Optimizing Abstract for Search Engines-

Many researchers searching for information online will use search engines such as Google, Yahoo or similar. By optimizing your paper for search engines, you will amplify the chance of someone finding it. This in turn will make it more likely to be viewed and/or cited in a further work. GJCST have compiled these guidelines to facilitate you to maximize the web-friendliness of the most public part of your paper.

Numerical Methods: Numerical methods used should be clear and, where appropriate, supported by references.

Acknowledgements: Please make these as concise as possible.

References:

References follow the *Harvard scheme* of referencing. References in the text should cite the authors' names followed by the time of their publication, unless there are three or more authors when simply the first author's name is quoted followed by et al. unpublished work has to only be cited where necessary, and only in the text. Copies of references in press in other journals have to be supplied with submitted typescripts. It is

necessary that all citations and references are carefully checked before submission, as mistakes or omissions will cause delays.

References to information on the World Wide Web can be given, but only if the information is available without charge to readers on an official site. Wikipedia and Similar websites are not allowed where anyone can change the information. Authors will be asked to make available electronic copies of the cited information for inclusion on the GJCST homepage at the judgment of the Editorial Board.

The Editorial Board and GJCST recommend that, citation of online published papers and other material should be done via a DOI (digital object identifier). If an author cites anything which does not have a DOI, they run the risk of the cited material not being noticeable.

The Editorial Board and GJCST recommend the use of a tool such as Reference Manager for reference management and formatting.

Tables, Figures and Figure Legends

Tables: Tables should be few in number, cautiously designed, uncrowned, and include only essential data. Each must have an Arabic number, e.g. Table 4, a self-explanatory caption and be on a separate sheet. Vertical lines should not be used.

Figures: Figures be supposed to be submitted as separate files. Always take in a citation in the text for each figure using Arabic numbers, e.g. Fig. 4. Artwork must be submitted online in electronic form by e-mailing them.

Preparation of Electronic Figures for Publication:

Even though low quality images are sufficient for review purposes, print publication requires high quality images to prevent the final product being blurred or fuzzy. Submit (or e-mail) EPS (line art) or TIFF (halftone/photographs) files only. MS PowerPoint and Word Graphics are unsuitable for printed pictures. Do not use pixel-oriented software. Scans (TIFF only) should have a resolution of at least 350 dpi (halftone) or 700 to 1100 dpi (line drawings) in relation to the imitation size. Please give the data for figures in black and white or submit a Color Work Agreement Form. EPS files must be saved with fonts embedded (and with a TIFF preview, if possible).

For scanned images, the scanning resolution (at final image size) ought to be as follows to ensure good reproduction: line art: >650 dpi; halftones (including gel photographs) : >350 dpi; figures containing both halftone and line images: >650 dpi.

Color Charges: It is the rule of the GJCST for authors to pay the full cost for the reproduction of their color artwork. Hence, please note that, if there is color artwork in your manuscript when it is accepted for publication, our publishing require you to complete and return a color work agreement form before your paper can be published.

Figure Legends: Self-explanatory legends of all figures should be incorporated separately under the heading 'Legends to Figures'. In the full-text online edition of the journal, figure legends may possibly be truncated in abbreviated links to the full screen

version. Therefore, the first 100 characters of any legend should notify the reader, about the key aspects of the figure.

6. AFTER ACCEPTANCE

Upon approval of a paper for publication, the manuscript will be forwarded to the dean, who is responsible for the publication of the GJCST.

6.1 Proof Corrections

The corresponding author will receive an e-mail alert containing a link to a website or will be attached. A working e-mail address must therefore be provided for the related author.

Acrobat Reader will be required in order to read this file. This software can be downloaded

(Free of charge) from the following website:

www.adobe.com/products/acrobat/readstep2.html. This will facilitate the file to be opened, read on screen, and printed out in order for any corrections to be added. Further instructions will be sent with the proof.

Proofs must be returned to the dean at dean@computerresearch.org within three days of receipt.

As changes to proofs are costly, we inquire that you only correct typesetting errors. All illustrations are retained by the publisher. Please note that the authors are responsible for all statements made in their work, including changes made by the copy editor.

6.2 Early View of GJCST (Publication Prior to Print)

The GJCST is enclosed by our publishing's Early View service. Early View articles are complete full-text articles sent in advance of their publication. Early View articles are absolute and final. They have been completely reviewed, revised and edited for publication, and the authors' final corrections have been incorporated. Because they are in final form, no changes can be made after sending them. The nature of Early View articles means that they do not yet have volume, issue or page numbers, so Early View articles cannot be cited in the conventional way.

6.3 Author Services

Online production tracking is available for your article through Author Services. Author Services enables authors to track their article - once it has been accepted - through the production process to publication online and in print. Authors can check the status of their articles online and choose to receive automated e-mails at key stages of production. The authors will receive an e-mail with a unique link that enables them to register and have their article automatically added to the system. Please ensure that a complete e-mail address is provided when submitting the manuscript.

6.4 Author Material Archive Policy

Please note that if not specifically requested, publisher will dispose off hardcopy & electronic information submitted, after the two months of publication. If you require the return of any information submitted, please inform the Editorial Board or dean as soon as possible.

6.5 Offprint and Extra Copies

A PDF offprint of the online published article will be provided free of charge to the related author, and may be distributed according to the Publisher's terms and conditions. Additional paper offprint may be ordered by emailing us at: editor@computerresearch.org.

Index

A

Ad hoc On Demand Distance Vector · 122
Ad hoc routing protocols · 121
adaptive immunity · 4
Anonymous Proxy Server · 109
Artificial Intelligence · 3, 6
asynchronous concurrent systems · 5
Average Normalized Modified Retrieval Rank (*ANMRR*) · 162

B

BANDWIDTH-ON-DEMAND · 81
Bayesian networks · 2, 4
BEHAVIA fire model · 138
Biological data · 142
Bivens · 4, 6
Bloor Research Resolution · 23
BPD · 18, 19, 20
BPMN · 7, 18, 19, 20, 21, 22, 23
Business Process Diagram · 18, 20, 22
Business Process Modeling Notation · 18, 19, 23

C

cell transfer delay · 131
Cellular automata paradigm · 137
chromosome · 5
ColorLayout · 8, 11
Complex activity · 21
Conference · 6, 11, 208
Constant Modulus Algorithm · 87
context representation · 5
critic · 105

D

Data Analysis · 2, 3, 6
Data cleaning · 141
Data mining · 4
Denormalized databases · 45
design patterns · 176
Discrete Cosine Transform · 157

DNA · 88
Download time · 12, 16
Draw entire area · 129

E

Eavesdropping · 115
EdgeHistogram · 11
Exception · 18, 19, 21, 22, 23
EXIF · 7, 8, 9, 11

F

flux · 165
font tag errors · 15, 17
full-duplex transmission · 193
functional testing · 178
fuzzy logic · 2, 3, 5, 95

G

Genetic algorithms · 5

H

Hand-off GSM call · 83
HeartbeatComplete · 100
hesitation analysis · 179
HIPAA regulations · 72
Homomorphism · 73
Human expertise · 5

I

IEEE Multimedia · 11
Immature detectors · 4
immediate dropping · 84
Immune based · 4
improved proxy server · 113
Intelligent data analysis · 3
Intrusion · 7, 2, 4, 6
IUD · 46

J

Jakarta Lucene search engine · 152
Java Swing application · 8
JAXEN · 7
JDOM · 7
J-Unit · 77
JViews · 7, 18, 19, 20, 21, 22, 23

K

kernel function · 66
Knowledge management tools · 205

L

least-frequently used · 110
Lee, Solto and Mok's · 4
libtissue · 4
LUCS · 53

M

Machine learning · 3
Markup Validation Service · 12, 13, 17
MLP · 64
Mobile Ad-hoc Network · 187
Modeling · 6, 18, 19, 23
MPEG-7 · 7, 9, 11, 146
multi-party key generation technique · 42

N

Network based systems · 2
NWCBIR · 7, 8, 9, 10, 11

O

Object Management Group · 19
object-oriented programming · 183
opencv · 158
OPNET · 133

P

Parkinson database · 68
peak cell rate · 133
physical components · 116

physical noise source · 128
primer · 91
private key · 41
proxy caching · 108
Pudukkottai · 9

Q

QoS parameters · 134
qualitative measures · 12, 17

R

Ramamohanarao · 5, 6
RDF · 7
Redundancies · 143
Response time · 79
RREP · 122
rule-based IDS · 3

S

ScalableColor · 8, 11
script tag errors (STE) · 17
Semantic Annotation · 7
Shared key authentication · 119
SOM · 56
Speech recognition · 172
State machines · 5
state transition analysis · 5
Stock Photography · 7
SWARM · 139
SYBEX · 17

T

table tag errors · 15, 17
temperature adjustment factor · 106
template · 91
temporal difference · 103
Toggling · 177
Traffic policing · 132
Traffic Scaling Factor · 135
Training · 4

U

Use Case Model · 34, 35
User Interface detection · 177

UUCP · 37

V

Valance Skeleton Point · 168

Vector Machine · 5

W

W3C · 12, 13, 14, 17

Web page errors · 12, 14, 16

Webpage Analyzer · 12, 13

Website Extractor · 12, 13, 14, 17

WEP · 114

Wildcards · 149

Wired Equivalent Privacy · 118

X

XSLT · 154



save our planet

Global Journal of Computer Science and Technology

Visit us on the Web at <http://www.ComputerResearch.org>
or email us at: helpdesk@computerresearch.org



6 586986 142797 32011
GJCST August 2009 CT 1.1