July - December, 2014 Vol. 6 No. 2 ISSN 0973-5658

# BIJIT

*Impact Factor 0.605 and Index Copernicus Value (ICV) 4.75*

# BVICAM's

## International Journal of Information Technology

*Indexed with INSPEC (UK), Index Copernicus (Poland), ProQuest (UK), EBSCO (USA), Google Scholar (USA)*

## CONTENTS

BHARATI VIDYAPEETH
BHARATI VIDYAPEETH POONA
PUNE

Bharati Vidyapeeth's
Institute of Computer Applications and Management
A-4, Paschim Vihar, Rohtak Road, New Delhi-63

Email : bijit@bvicam.ac.in, Website : http://www.bvicam.ac.in

Volume 6, Number 2, July – December, 2014

***BIJIT - BVICAM's International Journal of Information Technology*** is a half yearly publication of Bharati Vidyapeeth's Institute of Computer Applications and Management (BVICAM), A-4, Paschim Vihar, Rohtak Road, New Delhi – 110063 (INDIA).

**ISSN 0973 – 5658**

**Disclaimer**
The opinions expressed and figures provided in the Journal; BIJIT, are the sole responsibility of the authors. The publisher and the editors bear no responsibility in this regard. Any and all such liabilities are disclaimed

All disputes are subject to Delhi jurisdiction only.

Address for Correspondence:
**Prof. M. N. Hoda**
Editor-in-Chief, BIJIT
Director, Bharati Vidyapeeth's Institute of Computer Applications and Management (BVICAM),
A-4, Paschim Vihar, Rohtak Road, New Delhi – 110063 (INDIA). Tel.: +91 – 11 – 25275055
Fax: +91 – 11 – 25255056; E-Mail: bijit@bvicam.ac.in, Visit us at www.bvicam.ac.in/bijit

Published and printed by Prof. M. N. Hoda, Editor-in-Chief, BIJIT and Director, Bharati Vidyapeeth's Institute of Computer Applications and Management (BVICAM), A-4, Paschim Vihar, New Delhi – 110063 (INDIA). Tel.: +91 – 11 – 25275055, Fax: +91 – 11 – 25255056. E-Mail: bijit@bvicam.ac.in; mca@bvicam.ac.in; Visit us at www.bvicam.ac.in/bijit

# *Our Indexing at International Level*

The **INSPEC, IET (UK),** formerly IEE (UK), database is an invaluable information resource for all scientists and engineers, that contains 13 million abstracts and specialized indexing to the world's quality research literature in the fields of physics and engineering. *For further details, click at* http://www.theiet.org/resources/inspec/

**Index Copernicus International (Poland)** is a journal indexing, ranking and abstracting site. This service helps a journal to grow from a local level to a global one as well as providing complete web-based solution for small editorial teams. ICV 2012 for the BIJIT is 4.75. *For further details, click at* http://jml2012.indexcopernicus.com/BVICAMs+International+Journal+of+Information+Technology,p4852,3.html

**ProQuest (UK)** connects people with vetted, reliable information. Key to serious research, the company has forged a 70-year reputation as a gateway to the world's knowledge – from dissertations to governmental and cultural archives to news, in all its forms. *For further details, click at* http://www.proquest.co.uk/en-UK/default.shtml

***EBSCOhost Electronic Journals Service (EJS)*** *is a gateway to thousands of e-journals containing millions of articles from hundreds of different publishers, all at one web site. For further details, click at* http://www.ebscohost.com/titleLists/tnh-coverage.htm

***Open J-Gate*** *is an electronic gateway to global journal literature in open access domain. Launched in 2006, Open J-Gate is aimed to promote OAI. For further details, click at* http://informindia.co.in/education/J-Gate-Engineering/JET-List.pdf

***DOAJ*** *aims at increasing the visibility and ease of use of open access scientific and scholarly journals, thereby promoting their increased usage and impact. For further details, click at*
http://www.doaj.org/doaj?func=issues&jId=87529&uiLanguage=en

***Google Scholar*** *provides a simple way to broadly search for scholarly literature and repositories from across different parts of the world. For further details, click at* http://scholar.google.com/scholar?hl=en&q=BIJIT%2BBVICAM&btnG=

***Cabell's Directory*** *of Publishing Opportunities contains a wealth of information designed to help researchers and academics, match their manuscripts with the scholarly journals which are most likely to publish those manuscripts. For further details, click at* https://ssl.cabells.com/index.aspx

***Academic Journals Database*** *is a universal index of periodical literature covering basic research from all fields of knowledge. For further details, click at* http://journaldatabase.org/journal/issn0973-5658

***Indian Citation Index (ICI)*** *is an abstracts and citation database, with multidisciplinary objective information/knowledge contents from about 1000 top Indian scholarly journals For further details, click at*
http://www.indiancitationindex.com/htms/release_notes.htm

***and many more..., for more details click at*** **http://www.bvicam.ac.in/BIJIT/indexing.asp**

# Editorial Board

# *Editorial*

*It is a matter of both honor and pleasure for us to put forth the twelfth issue of BIJIT; the BVICAM's International Journal of Information Technology. It presents a compilation of ten papers that span a broad variety of research topics in various emerging areas of Information Technology and Computer Science. Some application oriented papers, having novelty in application, have also been included in this issue, hoping that usage of these would further enrich the knowledge base and facilitate the overall economic growth. This issue again shows our commitment in realizing our vision "to achieve a standard comparable to the best in the field and finally become a symbol of quality".*

*As a matter of policy of the Journal, all the manuscripts received and considered for the Journal, by the editorial board, are double blind peer reviewed independently by at-least two referees. Our panel of expert referees posses a sound academic background and have a rich publication record in various prestigious journals representing Universities, Research Laboratories and other institutions of repute, which, we intend to further augment from time to time. Finalizing the constitution of the panel of referees, for double blind peer review(s) of the considered manuscripts, was a painstaking process, but it helped us to ensure that the best of the considered manuscripts are showcased and that too after undergoing multiple cycles of review, as required.*

*The Ten papers, that were finally published, were chosen out of seventy nine papers that we received from all over the world for this issue. We understand that the confirmation of final acceptance, to the authors / contributors, sometime is delayed, but we also hope that you concur with us in the fact that quality review is a time taking process and is further delayed if the reviewers are senior researchers in their respective fields and hence, are hard pressed for time.*

We further take pride in informing our authors, contributors, subscribers and reviewers that the journal has been indexed with some of the world's leading indexing / bibliographic agencies like **INSPEC** of IET (UK) formerly IEE (UK), **Index Copernicus International** (Poland) with **IC Value 4.75** for 2012, **ProQuest** (UK), **EBSCO** (USA), **Open J-Gate** (USA), **DOAJ** (Sweden), **Google Scholar, WorldCat** (USA), **Cabell's Directory** of Computer Science and Business Information System (USA), **Academic Journals Database**, **Open Science Directory, Indian Citation Index, etc.** and listed in the libraries of the world's leading Universities like **Stanford University, Florida Institute of Technology, University of South Australia, University of Zurich**, etc. Related links are available at http://www.bvicam.ac.in/bijit/indexing.asp. Based upon the papers published in the year 2012, its **Impact Factor** was found to be **0.605**. These encouraging results will certainly further increase the citations of the papers published in this journal thereby enhancing the overall research impact.

We wish to express our sincere gratitude to our panel of experts in steering the considered manuscripts through multiple cycles of review and bringing out the best from the contributing authors. We thank our esteemed authors for having shown confidence in BIJIT and considering it a platform to showcase and share their original research work. We would also wish to thank the authors whose papers were not published in this issue of the Journal, probably because of the minor shortcomings. However, we would like to encourage them to actively contribute for the forthcoming issues.

The undertaken Quality Assurance Process involved a series of well defined activities that, we hope, went a long way in ensuring the quality of the publication. Still, there is always a scope for improvement, and so, we request the contributors and readers to kindly mail us their criticism, suggestions and feedback at bijit@bvicam.ac.in and help us in further enhancing the quality of forthcoming issues.

**Editors**

# CONTENTS

# Performance Optimization of Benchmark Functions Using VTS-ABC Algorithm

**Twinkle Gupta**[1] and **Dharmender Kumar**[2]

*Submitted in February, 2014; Accepted in September, 2014*

***Abstract - A new variant based on tournament selection called VTS-ABC algorithm is provided in this paper. Its performance is compared with standard ABC algorithm with different size of data on several Benchmark functions and results show that VTS-ABC provides better quality of solution than original ABC algorithm in every case.***

***Index Terms – Artificial Bee Colony Algorithms, Nature-Inspired Meta-heuristics, Optimizations, Swarm Intelligence Algorithms, Tournament selection.***

**NOMENCLATURE**

ABC – Artificial Bee Colony
ACO – Ant Colony Optimization
BFS – Blocking Flow-Shop Scheduling
DE – Differential Evolution
EA – Evolutionary Algorithm
GA – Genetic Algorithm
MCN – Maximum Cycle Number
PSO – Particle Swarm Optimization
TS – Tournament size
TSP – Travelling Salesman Problem

## 1.0 INTRODUCTION

For optimization problems, various algorithms have been designed which are base donnature-inspired concepts [1].Evolutionary algorithms(EA) and swarm optimization algorithms are two different classes in which nature inspired algorithms are classified. Evolutionary algorithms like Genetic algorithms (GA) and Differential evolution (DE) attempt to carry out the phenomenon of natural evolution [2]. However, a swarm like ant colony, a flock of birds can be described as collection of interacting agents and their intelligence liein their way of interactions with other individuals and the environment [3]. Swarm optimization includes Particle swarm optimization (PSO) model on social behavior of bird flocking [4], Ant colony optimization (ACO) model on swarm of ants and Artificial Bee Colony (ABC) model on the intelligent foraging behaviour of honey bees [5]. Some important characteristics of ABC algorithm which makes it more attractive the another optimizational algorithms are:

1) Employs only three control parameters (population size, maximum cycle number and limit) [6].
2) Fastconvergencespeed.
3) Quite simple, flexible and robust [7] [8].
4) Easyintegrationwithotheroptimizationalgorithms.

Therefore, ABC algorithm is a very popular nature inspired meta-heuristic algorithm used to solve various kinds of optimization problems. In recent years, ABC has earned so much popularity and used widely in various application such as: Constrained optimization, Image processing, Clustering, Engineering Design, Blocking flow shop scheduling (BFS), TSP, Bioinformatics, Scheduling and many others [9]-[18].Similar to other stochastic population-based approaches like GA, Ant Colony etc. ABC algorithm also applied Roulette Wheel selection mechanism which chooses best solution always with high selection pressure and leads the algorithm into premature convergence. With ever-growing size of dataset, optimization of algorithm has become a big concern. This calls for a need of better algorithm.

The aim of this paper is to create such an algorithm named VTS-ABC algorithm. This new variant is based on tournament selection mechanism and selects variable tournament size each time in order to select the employed bees sharing their information with onlooker bees. Onlooker bees select solution from selected tournament size of solutions with less selection pressure so that high fitness solutions can't dominate and give better quality of solutions with large data set as well. A worst solution is also replaced by better solution generated randomly in each cycle.

Rest of the paper is divided in different sections as follows: Introduction to standard ABC algorithm is described in section 2. Section 3 describes the proposed VTS-ABC algorithm. Experiments and its simulation results to show performance on several Benchmark functions are described in section 4 and in the last; Conclusion of the paper is discussed.

## 2. ARTIFICIAL BEE COLONY ALGORITHM

In 2005, Karaboga firstly proposed Artificial Bee Colony algorithm for optimizing numerical problems [19] which includes employed bees, onlooker bees and scouts. The bee carrying out search randomly is known as a scout. The bee going to the food source visited by it before and sharing its information with onlooker bees is known as employed bee and the bee waiting on the dance area called onlooker bee. ABC algorithm as a collective intelligence searching model has three essential components: Employed bees, Unemployed bees (onlooker and scout bees) and Food sources. In the view of optimization problem, a food source represents a possible solution. The position of a good food source indicates the solution providing better results to the given optimization problem. The quality of nectar of a food source represents the fitness value of the associated solution.

[1] *Master of Technology (CSE), Guru Jambheshwar University of Science and Technology, Hisar, Haryana, India.*
[2] *Associate Professor (CSE), Guru Jambheshwar University of Science and Technology, Hisar, Haryana, India.*
*E-mail:* [1] *twinkle2803.gupta@gmail.com and* [2] *dharmindia24@gmail.com*

Initially, a randomly distributed food source position of SNsize, the size of employed bees or onlooker bees is generated. Each solution $x_i$ is a D-dimensional vector that represents the number of optimized parameters and produced usingthe equation 1:

$$x_{i,j} = x_{min,j} + rand(0,1)\left(x_{max,j} - x_{min,j}\right) \quad (1)$$

where, $x_{max}$ and $x_{min}$ are the upper and lower bound of the parameter $x_i$ , respectively and j denotes the dimension. The fitness of food sources to find the global optimal is calculated by the following formula:

$$fit_m(x_m) = \left( \begin{array}{cc} \dfrac{1}{1+f_m(x_m)} & , f_m(x_m) > 0 \\ 1 + |f_m(x_m)| & , f_m(x_m) < 0 \end{array} \right) \qquad (2)$$

where, $f_m(x_m)$ is the objective function value of $x_m$. Then the employed bee phase starts. In this phase, each employed bee $x_i$ finds a new food source $v_i$in its neighborhood using the equation 3:

$$x_{i,j}(t+1) = x_{i,j}(t) + \emptyset\left(x_{i,j}(t) - x_{k,j}(t)\right) \qquad (3)$$

where, t: Cycle number; $x_k$: Randomly chosen employed bee and k is not equal to i ; $\emptyset(\,)$: A series of random variable in the range [-1, 1]. The fitness of new solution produced is compared with that of current solution and memorizes the better one by means of a greedy selection mechanism.

Employed bees share their information about food sources with onlooker bees waiting in the hive and onlooker bees probabilistically choose their food sources using fitness based selection technique such as roulette wheel selection shown in equation 4:

$$P_i = \frac{F(\theta_i)}{\sum_{k=1}^{S} F(\theta_k)} \qquad (4)$$

where, $P_i$: Probability of selecting the $i^{th}$ employed bee, S: Size of employed bees, $\theta_i$: Position of the $i^{th}$ employed bee and $F(\theta_i)$ : Fitness value. Afterthatonlookerbeescarried outrandomly searchintheirneighborhood similar to employed bees and memorize the better one.

Employed bees whose solutions can't be improved through a predetermined number of cycles, called limit, become scouts and their solutions are abandoned. Then, they find a new random food source position using the following equation 5:

$$x_{i,j} = x_{jmin} + r \cdot \left(x_{jmax} - x_{jmin}\right) \quad (5)$$

Where, r: A random number between 0 and 1 and these steps are repeated through a predetermined number of cycles called Maximum Cycle Number (MCN).

## 3. PROPOSED WORK: VTS-ABC ALGORITHM

In every meta-heuristic algorithm mainly two factors need to be balanced for global optimization outcome i.e. Exploration and Exploitation but ABC is a poor balance of these two factors. Various variants of ABC have been modelled for its improvement in different phases by number of researchers like Sharma and Pant have proposed a variant of ABC called RABC for solving the numerical optimization problem [20] and Tsai et al. have presented an interactive ABC optimization algorithm to solve combinational optimization problem [21] in which the concept of universal gravitational force for the movement of onlooker bees is introduced to enhance the exploration ability of the ABC algorithm. D. Kumar and B. Kumar also reviewed various papers on ABC and give a modified RABC algorithm based on topology for optimization of benchmark functions [22] [23].

Intelligence of ABC algorithm mainly depends upon the communication between individual agents. Employed beesshare their information with onlooker bees waiting in the hive and flow of this information from one individual to another depends on the selection mechanism used. Different selection schemes select different individuals to share the information which affect the communication ability of individuals and primarily the outcome of the algorithm. ABC algorithm uses Roulette wheel selection mechanism in which each onlooker bee selects the food source based on certain probability. Each onlooker bee selects the best food source with high selection pressure and lead to premature convergence. To overcome this problem, its new variant is proposed in which Tournament Selection method is applied based on Cycle number and number of employed bees.

In Tournament selection, a tournament size (TS) is chosen to select the number of employed bees sharing the information with onlooker bees. For better exploration, TS=2 i.e. Binary Tournament is applied in early stages and for better exploitation, variable tournament size is applied based on the current cycle number (CYL) and size of employed bee in middle stages. As the stages grow, this method works similar to Roulette wheel method in the end. Hence, the selection pressure is less in early stages and more in final stages which provide us better quality of solution. As variable size of tournament is used at different stages of the algorithm, hence the algorithm named VTS-ABC (Variable Tournament Size Artificial Bee Colony) algorithm. Method used for calculating TS is shown in equation 6 and equation 7:

If SN >= 20

$$TS = \left( \begin{array}{c} (SN) * \dfrac{i}{10}, if MCN * \dfrac{i-1}{10} \le CYL \le MCN * \dfrac{i}{10} \\ and\ i = 1,2 \ldots 10 \end{array} \right. \qquad (6)$$

If SN<20

$$TS = \left\{ \begin{array}{l} 2\,, if CYL \le \dfrac{MCN}{5} \\ TS+1\,, if \dfrac{MCN}{5} * i < CYL < \dfrac{MCN}{5} * (i+1) \\ and TS < SN and SN < 10 \\ TS + b\,, if \dfrac{MCN * i}{5} < CYL < \dfrac{MCN * (i+1)}{5} \\ and TS < SN and\ 10 \le SN \le 20 \\ SN, if CYL < MCN \end{array} \right. \qquad (7)$$

Where $b = \frac{SN - mod(SN,5)}{5}$

Here, two equations are shown for calculating tournament size of tournament selection method. The purpose of using these two equations is to increase the speed of algorithm. When the

size of employed bee i.e. given population of food source positions is small like 10, a solution can be easily found by changing the tournament size by 1 but as the size grows i.e. when best food source position is to be found in large set of population for example when SN=40 or more than 40, increasing size of tournament by 1 and 2 only is a very tedious task as it will take more time to run the algorithm. Hence, in order to increase speed of algorithm, the tournament size based on current cycle and size of population is increased.
One more concept is applied to increase its convergence speed. At each iteration or cycle, a new solution is generated randomly similar to scout and its fitness value is calculated. Greedy selection mechanism is applied between new solution and worst one and the better solution is memorized. Hence, it helps in finding good quality of solution as well as improving the convergence speed and provides better balance between exploration and exploitation.

## 4. EXPERIMENTS AND SIMULATION RESULTS

### 4.1 Benchmark Functions

The Benchmark Functions used to compare the performance of VTS-ABC algorithm with original ABC algorithm are illustrated below:

1) Sphere Function:

$$ObjVal = \sum_{j=1:p} (X^2)'$$

2) Schwefel Function:

$$ObjVal = \sum_{j=1:p} ObjVal + (X(1) - X(j)^2)^2 + (X(1) - 1)^2$$

3) Griewank Function:

$$ObjVal = \left(\sum\left(\frac{X^2}{4000}\right)'\right)' - \prod\left(Cos\left(\frac{X}{\sqrt{\alpha}}\right)'\right)' + 1$$

Where $\alpha = repmat(1{:}p, [n\ 1])$

4) Ackley Function:

$$ObjVal = -20 * e^{-0.2*\sqrt{\frac{1}{p}*(\sum(X\hat{}2)')'}} - e^{\frac{1}{p}*(\sum(Cos(2*\pi*X))')'} + 20 + e^1$$

Here, ObjVal is the function value calculated for each food source position. A food source is represented by X and population size is taken of n*p matrix where n is the no. of possible food source positions and p represents the dimension of each position.

### 4.2 Performance Measures & Simulation Result

The experimental results of VTS-ABC and ABC algorithm in MATLAB are taken under the parameter of size of food source positions (n*p) i.e. different size of population with different dimensions are taken to run and compare both algorithms. MCN is set as 2000 and each algorithm is run for 3 iteration i.e. Runtime=3. Limit for scouts is set equals to 300. In order to provide the quantitative assessment of the performance of an optimization algorithm, Mean of Global Minimum i.e. mean of minimum objective function value at each cycle of all iterations are taken as performance measure whose values are shown in table1and figure 1-4.

| Benchmark function | Algorithm | 20*10 | 100*100 | 150*100 |
|---|---|---|---|---|
| Sphere | ABC | 0.340462 | 13.0503 | 0.0124608 |
| | VTS-ABC | 0.0988222 | 6.70776 | 0.011053 |
| Schwefel | ABC | 1.23231 | 0.107861 | 19.0437 |
| | VTS-ABC | 0.729075 | 0.107592 | 14.3503 |
| Griewank | ABC | 0.0619326 | 0.000526703 | 0.447714 |
| | VTS-ABC | 0.0146616 | 0.00043907 | 0.189238 |
| Ackley | ABC | 1.56513 | 0.0648988 | 2.57993 |
| | VTS-ABC | 0.946886 | 0.0604899 | 2.13692 |

**Table1: Mean of Global minimum on different size of data**

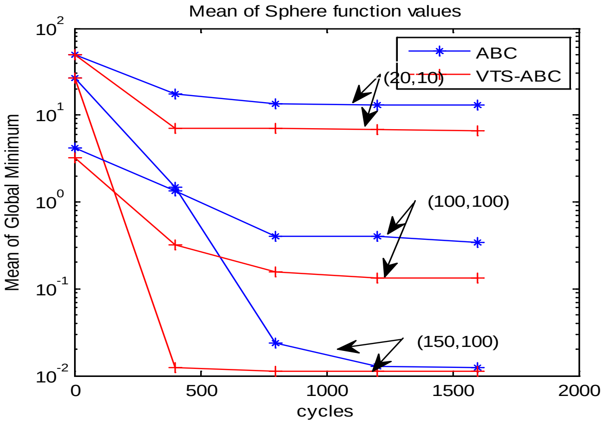


**Figure 1: Mean of Sphere function values on different size of data**

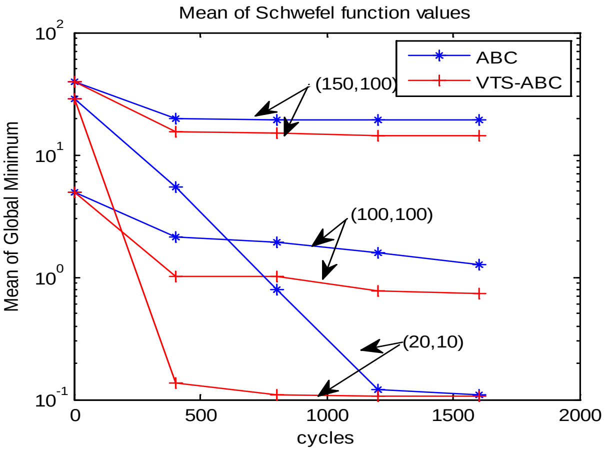


**Figure 2: Mean of Schwefel function values on different size of data**

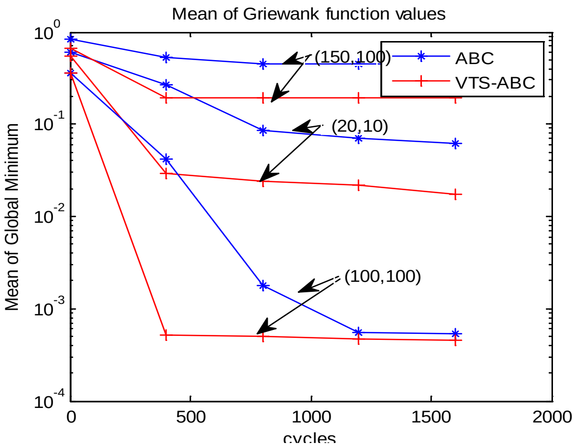


**Figure 3: Mean of Griewank function values on different size of data**

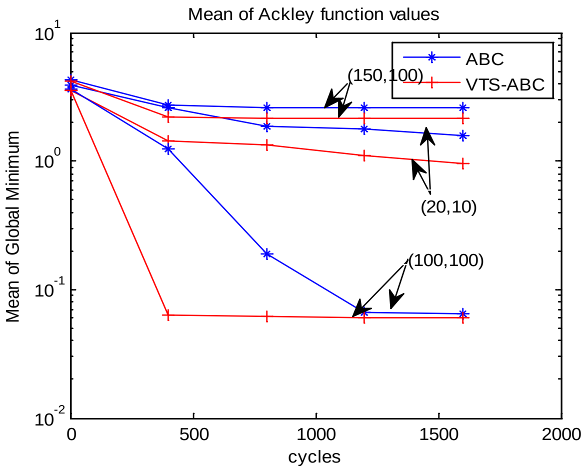


**Figure 4: Mean of Ackley function values on different size of data**

Figure 1 to 4 show simulation results of ABC and VTS-ABC algorithm with different size of data on Sphere, Schwefel, Griewank, Ackley respectively and reveal that VTS-ABC algorithm provides us better quality of solution than original ABC algorithm by minimizing objective function value or producing higher fitness solutions.

## 5. DISCUSSION AND CONCLUSION

In this paper, a new algorithm VTS-ABC is presented. In this algorithm, firstly variable tournament size (TS) is applied to select the food source position for onlooker bees which helps to achieve diversity in solution. Then to increase convergence speed, a new solution is generated in each cycle which replaced the worst one. In order to demonstrate the performance of proposed algorithm, it is applied on several Benchmark functions with different size of data set as input. Simulation results show that it provides better quality of solution than original ABC algorithm in every case. Therefore, it can be applied in different fields of optimization with large and higher dimensions data set efficiently.

## 6.0 REFERENCES

[1]. Yugal Kumar and Dharmender Kumar, “Parametric Analysis of Nature Inspired Optimization Techniques” International Journal of Computer Applications, vol. 32, no. 3, pp. 42-49, Oct. 2011.

[2]. P. J. Angeline, J. B. Pollack and G.M. Saunders, “An evolutionary algorithm that constructs recurrent neural networks,” Neural Networks in IEEE Transactions on, vol. 5, no. 1, 1994, pp. 54-65.

[3]. J. Kennedy and R. Eberhart, “Particle swarm optimization,” in Proceedings of IEEE international conference on neural networks, 1995, vol. 4, pp. 1942–1948.

[4]. E. Bonabeau, M. Dorgio, and G. Theraulaz, “Swarm intelligence: from neural network to artificial intelligence,” NY: oxford university press, New York, 1999.

[5]. D. Karaboga, “An idea based on honey bee swarm for numerical optimization,” Techn.Rep. TR06, Erciyes Univ. Press, Erciyes, 2005.

[6]. D. Karaboga and B. Akay, “A comparative study of artificial bee colony algorithm,” Applied Mathematics and Computation, vol. 214, no. 1, pp. 108–132, 2009.

[7]. R. S. Rao, S. V. L. Narasimham, and M. Ramalingaraju, “Optimization of distribution network configuration for loss reduction using artificial bee colony algorithm,” International Journal of Electrical Power and Energy Systems Engineering, vol. 1, no.2, pp. 116–122, 2008.

[8]. A. Singh, “An artificial bee colony algorithm for the leaf-constrained minimum spanning tree problem,” Applied Soft Computing, vol. 9, no. 2, pp. 625–631, Mar. 2009.

[9]. D. Karaboga and B. Basturk, “Artificial bee colony (ABC) optimization algorithm for solving constrained optimization problems,” in Foundations of Fuzzy Logic and Soft Computing, Springer, 2007, pp. 789–798.

[10]. C. Chidambaram and H. S. Lopes, “A new approach for template matching in digital images using an Artificial Bee Colony Algorithm,” in World Congress on Nature Biologically Inspired Computing, 2009. NaBIC 2009, IEEE, 2009, pp. 146–151.

[11]. N. K. Kaur Mann, “Review Paper on Clustering Techniques,” Global Journal of Computer Science and Technology, vol. 13, no. 5, 2013.

[12]. S. Okdem, D. Karaboga, and C. Ozturk, “An application of Wireless Sensor Network routing based on Artificial Bee Colony Algorithm,” in 2011 IEEE Congress on Evolutionary Computation (CEC), 2011, pp. 326–330.

[13]. T. K. Sharma, M. Pant, and J. C. Bansal, “Some modifications to enhance the performance of Artificial Bee Colony,” in 2012 IEEE Congress on Evolutionary Computation (CEC), 2012, pp. 1–8.

[14]. L. Bao and J. Zeng, “Comparison and analysis of the

selection mechanism in the artificial bee colony algorithm," in Hybrid Intelligent Systems, 2009. HIS'09. Ninth International Conference on, 2009, vol. 1, pp. 411–41.

[15]. C. M. V. Benítez and H. S. Lopes, "Parallel Artificial Bee Colony Algorithm Approaches for Protein Structure Prediction Using the 3DHP-SC Model," in Intelligent Distributed Computing IV, M. Essaaidi, M. Malgeri, and C. Badica, Eds. Springer Berlin Heidelberg, 2010, pp. 255–264.

[16]. D. L. González-Álvarez, M. A. Vega-Rodríguez, J. A. Gómez-Pulido, and J. M. Sánchez-Pérez, "Finding Motifs in DNA Sequences Applying a Multiobjective Artificial Bee Colony (MOABC) Algorithm," in Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics, C. Pizzuti, M. D. Ritchie, and M. Giacobini, Eds. Springer Berlin Heidelberg, 2011, pp. 89–100.

[17]. L. Wang, G. Zhou, Y. Xu, S. Wang, and M. Liu, "An effective artificial bee colony algorithm for the flexible job-shop scheduling problem," Int J Adv Manuf Technol, vol. 60, no. 1–4, pp. 303–315, Apr. 2012.

[18]. S.-W. Lin and K.-C. Ying, "Increasing the total net revenue for single machine order acceptance and scheduling problems using an artificial bee colony algorithm," J Oper Res Soc, vol. 64, no. 2, pp. 293–311, Feb. 2013.

[19]. D. Karaboga, "An idea based on honey bee swarm for numerical optimization," Techn.Rep. TR06, Erciyes Univ. Press, Erciyes, 2005.

[20]. T. K. Sharma, M. Pant, and J. C. Bansal, "Some modifications to enhance the performance of Artificial Bee Colony," in 2012 IEEE Congress on Evolutionary Computation (CEC), 2012, pp. 1–8.

[21]. TSai, Pei-Wei, et al. , "Enhanced artificial bee colony optimization." International Journal of Innovative Computing, Information and Control ,vol. 5, no. 12, 2009, pp.5081-5092.

[22]. B. K. Verma and D. Kumar, "A review on Artificial Bee Colony algorithm," International Journal of Engineering & Technology, vol. 2, no. 3, pp. 175–186, 2013.

[23]. D. Kumar and B. Kumar, "Optimization of Benchmark Functions Using Artificial Bee Colony (ABC) Algorithm," IOSR Journal of Engineering, vol. 3, no. 10, pp. 09-14, October 2013.

# Optimization of Component Based Software Engineering Model Using Neural Network

**Gaurav Kumar**[1] and **Pradeep Kumar Bhatia**[2]

*Submitted in November, 2013; Accepted in September, 2014*

***Abstract - The goal of Component Based Software Engineering (CBSE) is to deliver high quality, more reliable and more maintainable software systems in a shorter time and within limited budget by reusing and combining existing quality components. A high quality system can be achieved by using quality components, framework and integration process that plays a significant role. So, techniques and methods used for quality assurance and assessment of a component based system is different from those of the traditional software engineering methodology. In this paper, we are presenting a model for optimizing Chidamber and Kemerer (CK) metric values of component-based software. A deep analysis of a series of CK metrics of the software components design patterns is done and metric values are drawn from them. By using unsupervised neural network- Self Organizing Map, we have proposed a model that provides an optimized model for Software Component engineering model based on reusability that depends on CK metric values. Average, standard deviated and optimized values for the CK metric are compared and evaluated to show the optimized reusability of component based model.***

***Index Terms – Chidamber and Kemerer (CK) metric; Component Based Software Engineering (CBSE); Neural Network (NN); Self Organizing Map (SOM).***

## 1.0 INTRODUCTION

Reusability of software can be enhanced by using the structured approaches of Component-Based Software Engineering (CBSE). CBSE includes various object-oriented concepts such as Reusability through Inheritance, encapsulation, abstraction and polymorphism. Features of CBSE includes increase in productivity, improvement in quality, reduced time to market, broad range of reusability and effective management of complexity. The main characteristics of CBSE are:

- CBSE considers a component as a reusable entity.
- CBSE supports the development of system as the integration of components.
- CBSE provides facilities for upgrading and maintaining a system by simply changing the components that needs to be upgraded or replaced.

Software component with specified interfaces can be deployed independently and each component communicates with other component(s) by using its public interfaces. A software component is a self contained software element that can be specified formally, composed without modification according to deployment and composition standard, deployed independently from its environment without needs of other specific components.

A component interfaces provides functional properties and can be divided into 2 parts: *Behavior part* specifies behavior of a component; *Signature part* specifies operations provided by a component that are understandable by both interface provider (component) and user (other components or other software that interact with provider). A component has two kinds of interfaces that can be distinguish as *imported interface* which describes those services that a component requires from its environments, e*xported interface* which describes those services that a component provides to its environment.

Chidamber and Kemerer (CK) metric used in the proposed model includes weighted sum of all the methods in a class called *Weighted Methods per Class* (WMC); count of the number of other classes to which a given class is coupled called *Coupling Between Object classes* (CBO); in the inheritance hierarchy, length of the longest path from a given class to the root class called *Depth of the Inheritance Tree* (DIT); a count of the number of immediate child classes called *Number of Children* (NOC); count of the methods that can be invoked in response to a message received by an object of a particular class called *Response for a Class* (RFC);count of the number of classes used in the component (NC).

## 2.0 LITRATURE SURVEY

A brief literature review of work done in the field of Component Based Software Engineering is discussed here:

Kilsup Lee and Sung Jong Lee (2005) proposed a quantitative software quality evaluation model with respect to the Component Based Development (CBD) methodology. The evaluation is done using checklists that helps to acquire high quality software. The model proposed includes international standards (quality characteristic, sub-characteristics, quality metrics, and quality evaluation process) for software quality model and evaluation process.

Alexandre Alvaro et. al. (2006) analyzed and evaluated components using consistent and well-defined characteristics, sub-characteristics and quality attributes related metrics.

Kung-Kiu Lau and Zheng Wang (2007) classified and evaluated taxonomy w.r.t. parameters, its key characteristics based on commonly accepted parameters for Component Based Development.

Anita Gupta et. al. (2008) discussed an industrial case study involving a reusable Java-class framework and an application to use that framework. The impact of software changes on

[1,2] *Department of Computer Science & Engineering, Guru Jambheshwar University of Science & Technology, Hisar, Haryana, India*
*E-mail: [1]er.gkgupta@gmail.com,and [2]pkbhatia.gju@gmail.com*

different development characteristics (e.g. impact of reuse and impact of refactoring) are analyzed. Perfective and corrective changes in both reusable and non-reusable software are analyzed.

Mubarak Mohammad and Vasu Alagar (2008) proposed a CBSE approach that defines the trust worthy quality attributes as first class structural elements where formalism is integrated into various stages of the development process. A development framework of comprehensive tool support and justification of their role in assuring trustworthiness during the different stages of software development is discussed.

R. Senthil et. al. (2008) described and evaluated N-tier architecture for a component based application against the external and internal quality factors to establish an enhanced component model (ECM). A case study has been carried out to establish the result that the application so developed is scalable and robust as one can migrate from one data source to another using this quality model.

Yoonjung Choi et. al. (2008) presented a Component Quality Model which includes metrics for component quality evaluation, basic guidelines for evaluations, and reporting formats of evaluations. An improvement is made in the Component Quality Model for the proper tailoring when applied in embedded system domain. Two different projects are evaluated to use as guidelines and goals for component quality improvements.

V. Lakshmi Narasimhan et. al. (2009) carried out a systematic analysis and comparison of three suites of metrics and several key inferences have been drawn from them. The metric values provided are helpful to study the behavior of metrics under various quality factors. Inferences on complexity, reusability, testability, and stability of the underlying components have been drawn from various metrics evaluations.

María A. Reyes et. al. (2009) analyzes and proposed a systematic approach that allows assessing and improving products and processes of the conceptual elements behind Component-Based Software Engineering (CBSE) for its quality evaluation and integrating the product perspective as well as process perspective.

Anju Shri et. al. (2010) used Tuned CK metric suit as input to obtain the structural analysis of Object oriented based software components. Reusability of object oriented software components is evaluated using a hybrid K-Means and Decision tree approach. The proposed reusability model produces high precision results.

G. Shanmugasundaram et. al. (2011) proposed an evolutionary model using maturity level of reuse metrics to show the relationship between component based systems, object oriented systems, and service oriented systems using reuse metrics.

Aldeida Aleti and Indika Meedeniya (2011) proposed a Bayesian Heuristic for Component Deployment Optimization (BHCDO) which builds deployment architectures by using a Bayesian learning mechanism. BHCDO efficiently automates the search for component deployment design alternatives and outperforms state of the art optimization methods. BHCDO does not require any parameter tuning to have a good performance as required in other approximate optimization methods.

Mostefai Mohammed Amine and Mohamed Ahmed-Nacer (2011) proposed Knowledge Management System (KMS) implementation in a CBSE-oriented organization that uses short iterations, less resources and budget, continuous integration and intensive customer collaboration to eliminate risks and misconceptions.

Abhikriti Narwal (2012) conducted a survey and interviews with Software developers, Testers, Research Engineers of many major Software Companies to show usage of Complexity metrics in Component-based Software Systems may improve the quality of Software. Complex components are hard to understand and take much time to execute than simple components and are very difficult to maintain.

Amr Rekaby and Ayat Osama (2012) proposed a model containing activities involved in component-based development lifecycle that can decrease the effort of the projects by 40% after few months of application.

Sandeep Srivastava (2012) discussed the relation between software metrics and maintainability which characterize the ease of the maintenance process when applied to a specific product. It is determined that up to what point and in which cases we can rely on software metrics in order to define the maintainability of a software product.

Simrandeep Singh Thapar et. al. (2012) discussed usage of Component Based Software Development (CBSD) approach as a success factor for business that provides benefits like reusability, on-time delivery, high quality, and less cost. The major reason of focus of software organizations to implement quality management in software development is the quality expectations of customers at purchase time from the software.

Anupama Kaur et. al. (2012) proposed a neural network based approach to establish the relationship between different attributes for evaluating reusability. Structural attributes of function oriented software components are shown using software metrics i.e. McCabe's cyclometric complexity measure for complexity measurement, regularity Metric, Halstead Software Science Indicator for Volume indication, Reuse Frequency metric and Coupling Metric.

Ashish Oberoi and Deepti Arora (2014) analyzed CK metric values and used Self Organizing Map for evaluation of CK metric of component quality model. They provided CK metric value to improve the performance using design patterns.

Neha Goyal and Deepali Gupta (2014) proposed a model that uses unsupervised neural network to calculate reusability of a component based software with the help of Rational Rose.

## 3.0 PROPOSED WOK

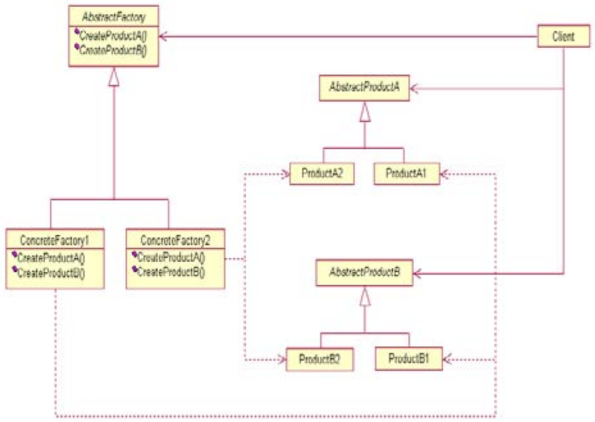
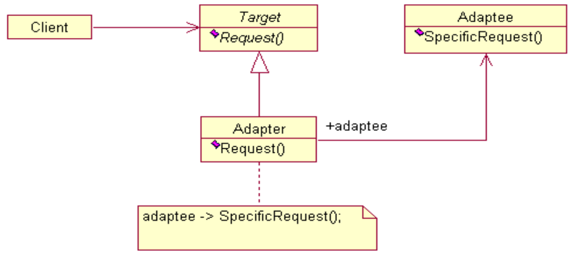
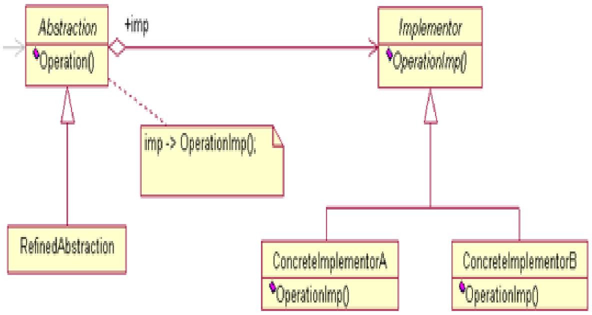
Design patterns are considered as separate components for the problem formulation. All Implementation has been done by using MATLAB Version 2013-Neural Network Tool Box. Data Structure, Algorithms, functions and implementation done for doing this work are discussed here. Ashish Oberoi and Deepti Arora (2014) derived quality model using same strategy but no comparison or evaluation has been done, on the basis of which optimum values was retrieved.

Optimization of Component based Software Engineering model is done through analyzing a series of design patterns which are worldwide accepted as the reuse design terminology for object oriented designing and hence component based designing. In the proposed model, firstly entire design pattern are analyzed obtained from Gamma et. al. (1995) by formulating CK metric analysis. Based on average, standard deviation on metrics values obtained and weight values got through unsupervised neural network Self Organizing Map (SOM); optimized values for components are achieved. Here we are using an unsupervised method because we don't know in advance output values for the corresponding design patterns. MatLab will be used for the implementation purpose. The problem is divided into two phases:

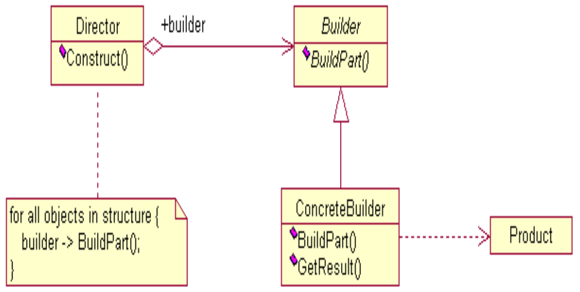
1. Analysis of Software Design Patterns through CK metrics analysis.
2. Implementation through Neural Network using MatLab

### 3.1. CK Metric Analysis of Software Design Patterns

Design patterns are usually used in the software design phase to create abstractions to provide independency in components and to handle future changes and maintaining architectural integrity. In this phase, CK metric values (WMC, DIT, NOC, RFC, CBO) of each component's every class is computed for design patterns. To get the value of metric NC, number of classes in each component is found at run time and added to the final CK metric values. Analysis of CK metric values for design patterns are shown in table 1.

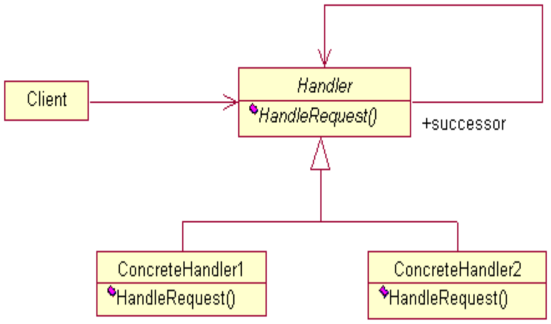
<table>
<tr><th>Design Pattern</th><th>Metric Values</th></tr>
<tr><th colspan="2">Abstract Factory</th></tr>
<tr><td>
</td><td>
<table>
<tr><th>Class/Metrics</th><th>NOM</th><th>DIT</th><th>NOC</th><th>CBO</th><th>RFC</th></tr>
<tr><td>Abstract Factory</td><td>2</td><td>0</td><td>2</td><td>0</td><td>2</td></tr>
<tr><td>Concrete Factory</td><td>2</td><td>1</td><td>0</td><td>2</td><td>2</td></tr>
<tr><td>Concrete Factory</td><td>2</td><td>1</td><td>0</td><td>2</td><td>2</td></tr>
<tr><td>Abstract Product</td><td>0</td><td>0</td><td>2</td><td>0</td><td>0</td></tr>
<tr><td>Product A2</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>Product A1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>Abstract Product</td><td>0</td><td>0</td><td>2</td><td>0</td><td>0</td></tr>
<tr><td>Product B2</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>Product B1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td></tr>
</table></td></tr>
<tr><th colspan="2">Adapter</th></tr>
<tr><td>
</td><td>
<table>
<tr><th>Class/Metrics</th><th>NOM</th><th>DIT</th><th>NOC</th><th>CBO</th><th>RFC</th></tr>
<tr><td>Target</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>Adapter</td><td>1</td><td>1</td><td>0</td><td>0</td><td>2</td></tr>
<tr><td>Adaptee</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr>
</table></td></tr>
<tr><th colspan="2">Bridge</th></tr>
<tr><td>
</td><td>
<table>
<tr><th>Class/Metrics</th><th>NOM</th><th>DIT</th><th>NOC</th><th>CBO</th><th>RFC</th></tr>
<tr><td>Abstaction</td><td>1</td><td>0</td><td>1</td><td>0</td><td>2</td></tr>
<tr><td>Refined Abstraction</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>Implementor</td><td>1</td><td>0</td><td>2</td><td>0</td><td>1</td></tr>
<tr><td>Concrete Implementor A</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>Concrete Implementor B</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td></tr>
</table></td></tr>
</table>

**Builder**



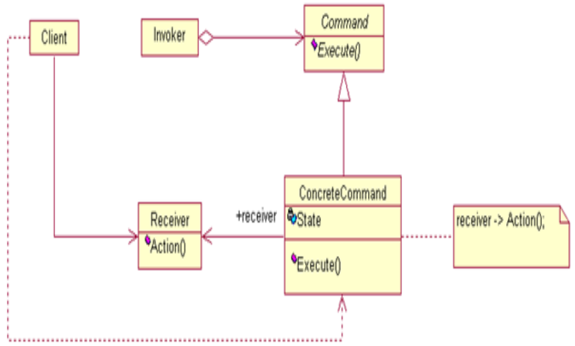
| Class/Metrics | NOM | DIT | NOC | CBO | RFC |
|---|---|---|---|---|---|
| Director | 1 | 0 | 0 | 0 | 2 |
| Builder | 1 | 0 | 1 | 0 | 1 |
| Concrete Builder | 2 | 1 | 0 | 1 | 2 |
| Product | 0 | 0 | 0 | 0 | 0 |

**Chain of Responsibility**



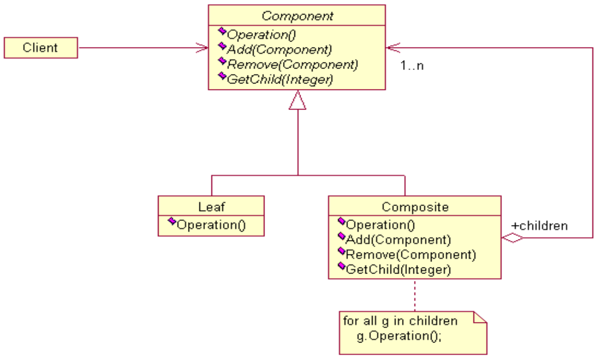
| Class/ Metrics | NOM | DIT | NOC | CBO | RFC |
|---|---|---|---|---|---|
| Handler | 1 | 0 | 2 | 0 | 2 |
| Concrete Handler 1 | 1 | 1 | 0 | 0 | 1 |
| Concrete Handler 2 | 1 | 1 | 0 | 0 | 1 |

**Command**



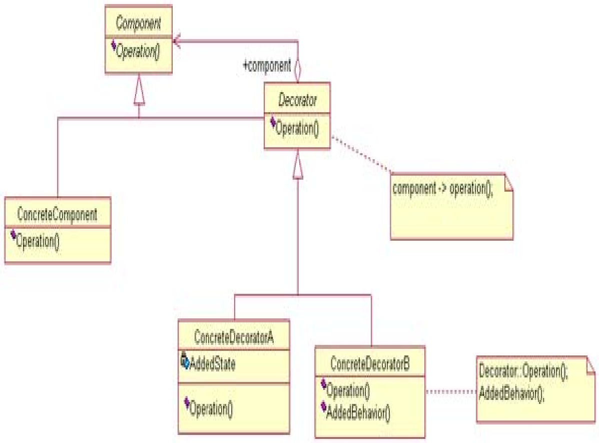
| Class/ Metrics | NOM | DIT | NOC | CBO | RFC |
|---|---|---|---|---|---|
| Invoker | 0 | 0 | 0 | 0 | 0 |
| Command | 1 | 0 | 1 | 0 | 1 |
| Concrete Command | 1 | 1 | 0 | 0 | 2 |
| Receiver | 1 | 0 | 0 | 0 | 1 |

**Composite**



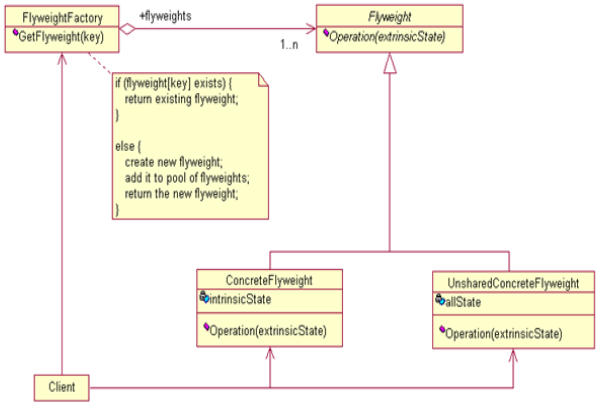
| Class/ Metrics | NOM | DIT | NOC | CBO | RFC |
|---|---|---|---|---|---|
| Component | 4 | 0 | 2 | 0 | 4 |
| Leaf | 1 | 1 | 0 | 0 | 1 |
| Composite | 4 | 1 | 0 | 0 | 5 |

**Decorator**



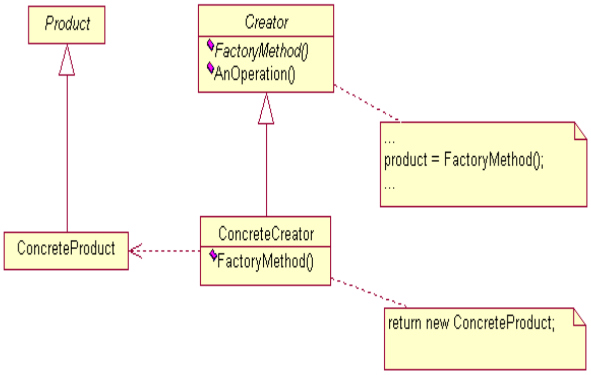
| Class/ Metrics | NOM | DIT | NOC | CBO | RFC |
|---|---|---|---|---|---|
| Component | 1 | 0 | 2 | 0 | 1 |
| Concrete Component | 1 | 1 | 0 | 0 | 1 |
| Decorator | 1 | 1 | 2 | 0 | 2 |
| Concrete Decorator A | 1 | 2 | 0 | 0 | 1 |
| Concrete Decorator B | 2 | 2 | 0 | 0 | 3 |

**Flyweight**


FlyweightFactory
GetFlyweight(key)
+flyweights
1..n
Flyweight
Operation(extrinsicState)
if (flyweight[key] exists) {
return existing flyweight;
}
else {
create new flyweight;
add it to pool of flyweights;
return the new flyweight;
}
ConcreteFlyweight
intrinsicState
Operation(extrinsicState)
UnsharedConcreteFlyweight
allState
Operation(extrinsicState)
Client

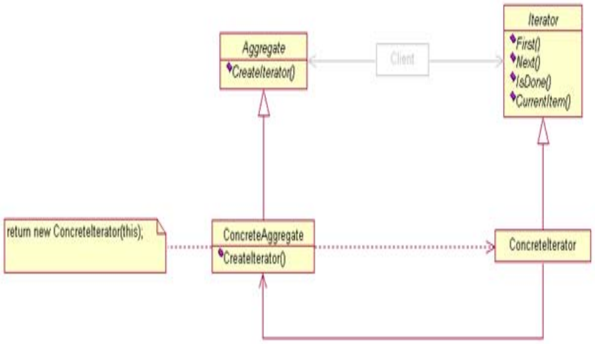
| Class/ Metrics | NOM | DIT | NOC | CBO | RFC |
|---|---|---|---|---|---|
| Fly Weight Factory | 1 | 0 | 0 | 0 | 1 |
| Fly weight | 1 | 0 | 2 | 0 | 1 |
| Concrete Fly weight | 1 | 1 | 0 | 0 | 1 |
| Unshared Concrete Fly weight | 1 | 1 | 0 | 0 | 1 |

**Factory Method**



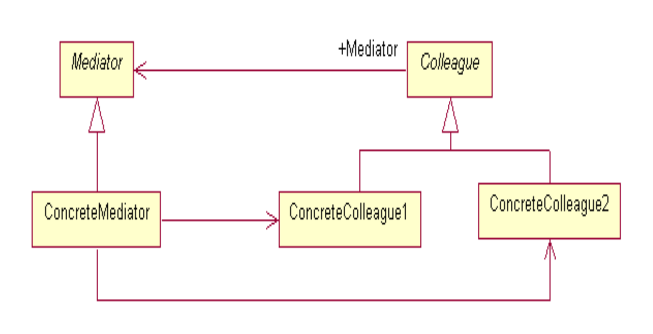
| Class/ Metrics | NOM | DIT | NOC | CBO | RFC |
|---|---|---|---|---|---|
| Product | 0 | 0 | 2 | 0 | 0 |
| Concrete Product | 0 | 1 | 0 | 0 | 0 |
| Creator | 2 | 0 | 1 | 0 | 2 |
| Concrete Creator | 1 | 1 | 0 | 1 | 1 |

**Iterator**



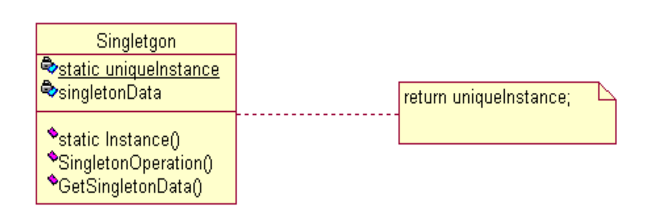
| Class/ Metrics | NOM | DIT | NOC | CBO | RFC |
|---|---|---|---|---|---|
| Aggregate | 1 | 0 | 1 | 0 | 1 |
| Concrete Aggregate | 1 | 1 | 0 | 1 | 2 |
| Iterator | 4 | 0 | 1 | 0 | 4 |
| Concrete Iterator | 0 | 1 | 0 | 0 | 0 |

**Mediator**



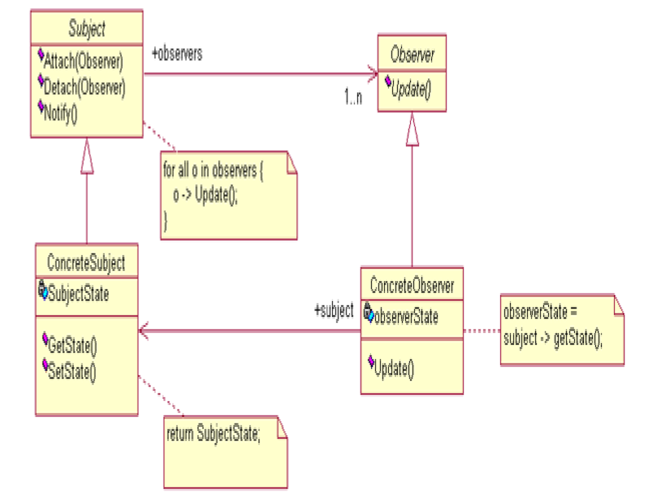
| Class/ Metrics | NOM | DIT | NOC | CBO | RFC |
|---|---|---|---|---|---|
| Mediator | 0 | 0 | 1 | 0 | 0 |
| Colleague | 0 | 0 | 2 | 0 | 0 |
| Concrete Mediator | 0 | 1 | 0 | 0 | 0 |
| Concrete Colleague 1 | 0 | 1 | 0 | 0 | 0 |
| Concrete Colleague 2 | 0 | 1 | 0 | 0 | 0 |

**Singleton**



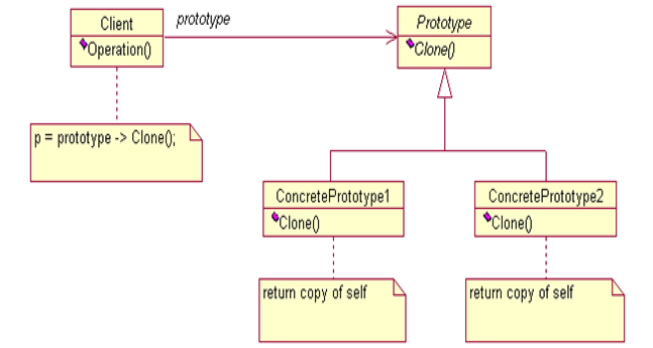
| Class/ Metrics | NOM | DIT | NOC | CBO | RFC |
|---|---|---|---|---|---|
| Singleton | 3 | 0 | 0 | 0 | 3 |

**Observer**



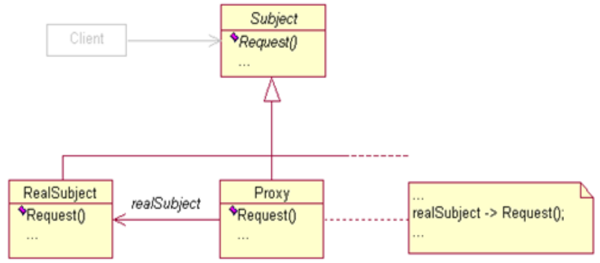
| Class/ Metrics | NOM | DIT | NOC | CBO | RFC |
|---|---|---|---|---|---|
| Subject | 3 | 0 | 1 | 0 | 4 |
| Concrete Subject | 2 | 1 | 0 | 0 | 2 |
| Observer | 1 | 0 | 1 | 0 | 1 |
| Concrete Observer | 1 | 1 | 0 | 0 | 2 |

**Prototype**



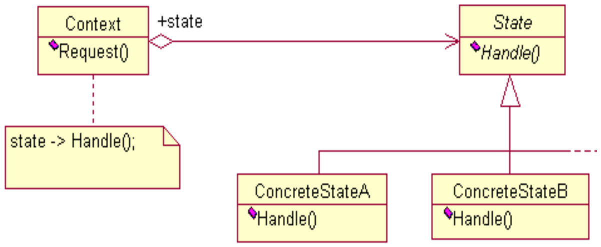
| Class/ Metrics | NOM | DIT | NOC | CBO | RFC |
|---|---|---|---|---|---|
| Prototype | 1 | 0 | 2 | 0 | 1 |
| Concrete Prototype 1 | 1 | 1 | 0 | 0 | 1 |
| Concrete Prototype 2 | 1 | 1 | 0 | 0 | 1 |

**Proxy**



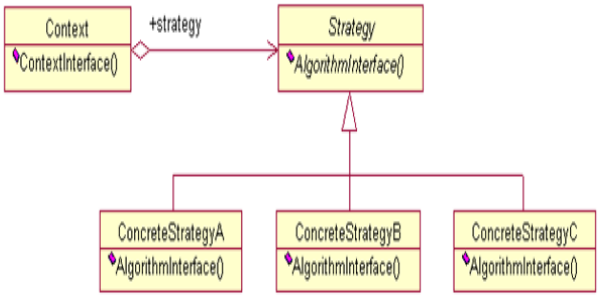
| Class/ Metrics | NOM | DIT | NOC | CBO | RFC |
|---|---|---|---|---|---|
| Subject | 1 | 0 | 2 | 0 | 1 |
| Real Subject | 1 | 1 | 0 | 0 | 1 |
| Proxy | 1 | 1 | 0 | 0 | 2 |

**State**



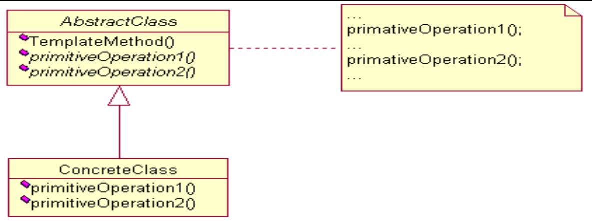
| Class/ Metrics | NOM | DIT | NOC | CBO | RFC |
|---|---|---|---|---|---|
| Context | 1 | 0 | 0 | 0 | 2 |
| State | 1 | 0 | 2 | 0 | 1 |
| Concrete State A | 1 | 1 | 0 | 0 | 1 |
| Concrete State B | 1 | 1 | 0 | 0 | 1 |

**Strategy**



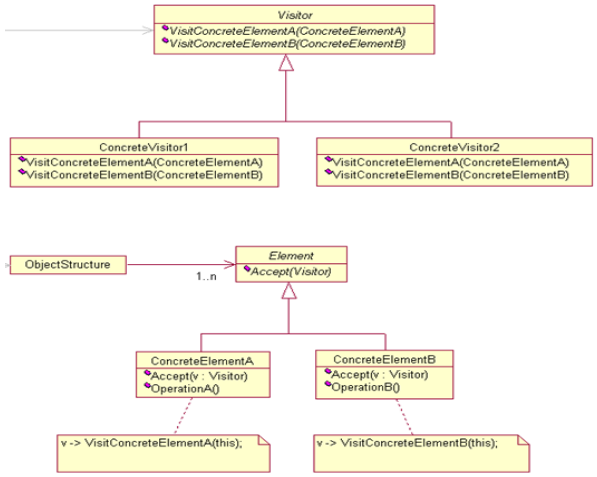
| Class/ Metrics | NOM | DIT | NOC | CBO | RFC |
|---|---|---|---|---|---|
| Context | 1 | 0 | 0 | 0 | 1 |
| Strategy | 1 | 0 | 3 | 0 | 1 |
| Concrete Strategy A | 1 | 1 | 0 | 0 | 1 |
| Concrete Strategy B | 1 | 1 | 0 | 0 | 1 |
| Concrete Strategy C | 1 | 1 | 0 | 0 | 1 |

**Template Method**



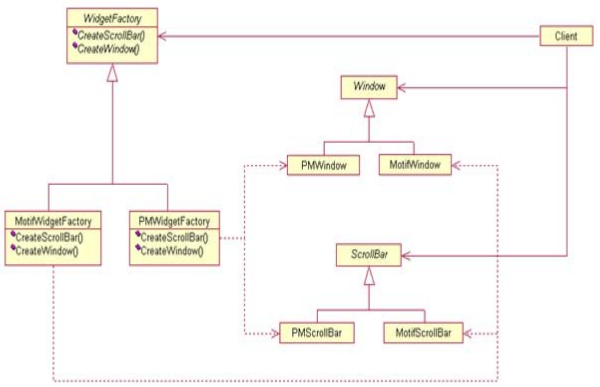
| Class/ Metrics | NOM | DIT | NOC | CBO | RFC |
|---|---|---|---|---|---|
| Abstract Class | 3 | 0 | 1 | 0 | 5 |
| Concrete Class | 2 | 1 | 0 | 0 | 2 |

**Visitor**


Visitor
VisitConcreteElementA(ConcreteElementA)
VisitConcreteElementB(ConcreteElementB)
ConcreteVisitor1
VisitConcreteElementA(ConcreteElementA)
VisitConcreteElementB(ConcreteElementB)
ConcreteVisitor2
VisitConcreteElementA(ConcreteElementA)
VisitConcreteElementB(ConcreteElementB)
ObjectStructure
1..n
Element
Accept(Visitor)
ConcreteElementA
Accept(v : Visitor)
OperationA()
ConcreteElementB
Accept(v : Visitor)
OperationB()
v -> VisitConcreteElementA(this);
v -> VisitConcreteElementB(this);

| Class/ Metrics | NOM | DIT | NOC | CBO | RFC |
|---|---|---|---|---|---|
| Visitor | 2 | 0 | 2 | 0 | 2 |
| Concrete Visitor 1 | 2 | 1 | 0 | 0 | 2 |
| Concrete Visitor 2 | 2 | 1 | 0 | 0 | 2 |
| Object Structure | 0 | 0 | 0 | 0 | 0 |
| Element | 1 | 1 | 2 | 0 | 1 |
| Concrete Element A | 2 | 1 | 0 | 0 | 3 |
| Concrete Element B | 2 | 1 | 0 | 0 | 3 |

**Widget Factory**



| Class/ Metrics | NOM | DIT | NOC | CBO | RFC |
|---|---|---|---|---|---|
| Widget Factory | 2 | 0 | 2 | 0 | 2 |
| Motif Widget Factory | 2 | 1 | 0 | 2 | 2 |
| PM Widget Factory | 2 | 1 | 0 | 2 | 2 |
| Window | 0 | 0 | 2 | 0 | 0 |
| PM Window | 0 | 1 | 0 | 0 | 0 |
| Motif Window | 0 | 1 | 0 | 0 | 0 |
| Scroll Bar | 0 | 0 | 2 | 0 | 0 |
| PM Scroll Bar | 0 | 1 | 0 | 0 | 0 |
| Motif Scroll Bar | 0 | 1 | 0 | 0 | 0 |

**Table 1: Design Pattern with Metric Values**

### 3.2. Implementation using Neural Network

Unsupervised neural network i.e. Self Organizing Map is created using newsom() of Neural Network Toolbox of MatLab. CK metric values of design patterns obtained through first step are feed as input in the created network. The parameters taken are: trainbuwb method as training method, no. of training data is 21x6=126, number of epoch taken to converge are 500. After the training, free parameter value i.e. weight is summed up for each metric. Weighted sum is divided by total number of metrics that returns optimum value for each metric which in turn establishes landmark for reusability of component based model. Also, average and standard deviation values are retrieved to compare with optimum values. For finding average and standard deviation values, mean() and std() of MatLab have been used.

## 4.0 RESULT ANALYSIS

Graphs showing comparison between Average values, Standard Deviation values and experimental values evaluated through SOM based analysis is shown in figure 1(a-c). CK metric is shown on x-axis and Values w.r.t. CK metric is shown on y-axis.

Variations in average, standard deviation and experimental values are shown in table 2. By getting average of all the variations as shown in table 2, optimum values can be found to improve the reusability of component based software engineering model.

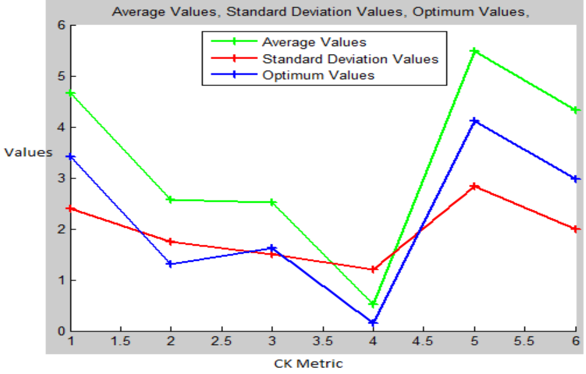


**Figure 1(a)**

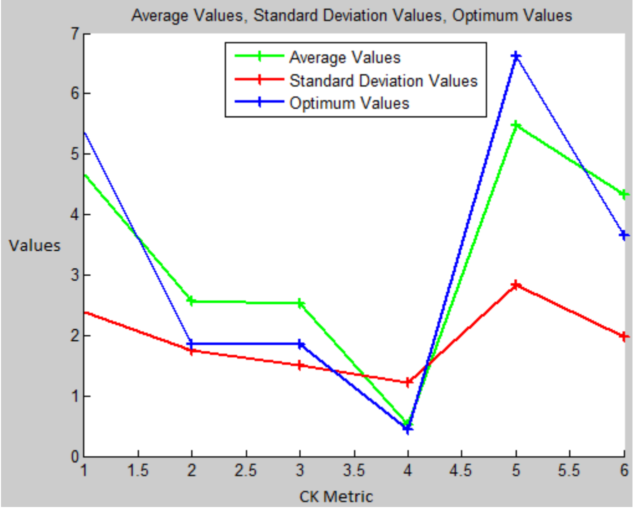


**Figure 1(b)**

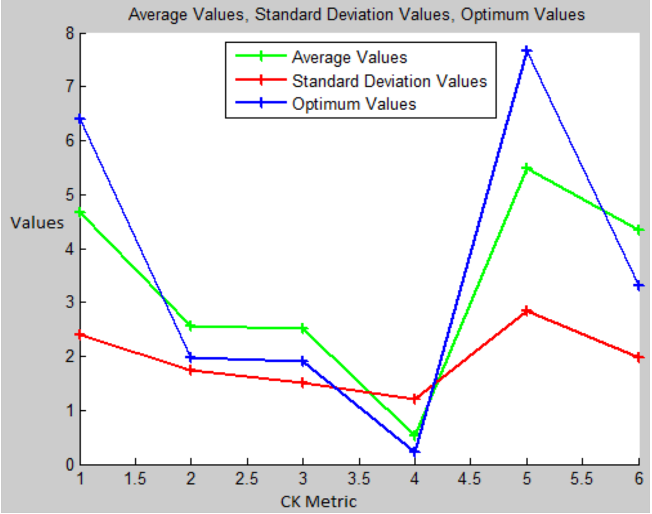


**Figure 1(c)**

**Figure 1: Variation 1 between Average, Standard Deviation, and Optimum value for design patterns.**

| S. No. | CK Metric | Variation 1 | Variation 2 | Variation 3 | Optimum Values |
|---|---|---|---|---|---|
| 1 | WMC | 4 | 6 | 7 | 5.666 |
| 2 | DIT | 2 | 2 | 2 | 2 |
| 3 | RFC | 2 | 2 | 2 | 2 |
| 4 | NOC | 1 | 1 | 1 | 1 |
| 5 | NC | 5 | 7 | 8 | 6.666 |
| 6 | CBO | 3 | 4 | 4 | 3.666 |

**Table 2: Variations between Average, Standard Deviation and Optimum Values**

## 5.0 CONCLUSION

CBSE is a knowledge-intensive activity that helps in producing better quality software systems by playing significant role in achieving programmer's productivity, system flexibility, and overall system quality. A model has been proposed for optimizing CK metric values with respect to the Component Based Software Engineering (CBSE) methodology using design patterns. UML properties have been applied to find out the various metrics from each and every class of various types of design patterns (components). While adding up the metric values for each component, number of classes is calculated at run time and concatenated with the final metrics value. These metric values are feed to un-supervised neural network. Graphs have been plotted with the help of average, standard deviation and optimized values to show the variation. Optimum values achieved on the basis of CK metric variations provide an optimized model for Software Component Engineering model.

## 6.0 LIMITATIONS & FUTURE SCOPE

For the evaluation of software components, a component quality model is required that reuse not only the functional parts, but also achieve easier and more accurate predictability of the system behavior. The proposed model used for optimization of component based software engineering can give better results if good amount of data is provided with realistic values e.g. historical project values of a software company that have used component based development. If that data is used as an input and results in terms of quality factors like reusability, maintainability, complexity, testability etc. are given as output. A supervised neural network may give better results as compared to unsupervised neural network which is used in the proposed model.

## 7.0 REFERENCES

[1]. Ashish Oberoi, Deepti Arora, "Quality Model For Analysis And Implentation Of CK Metrics Through Neural Networks", National Conference on Advances in Engineering and Technology, pp. 46-51, March 2014.

[2]. Neha Goyal, Deepali Gupta, "Reusability Calculation of Object Oriented Software Model by Analyzing CK Metric", International Journal of Advanced Research in Computer Engineering & Technology, Vol. 3, Issue 7, pp. 2466-2470, July 2014.

[3]. Rajni Jain, Satma M C, Alka Aroa, Sudeep Marwaha, R C Goyal, "Online Rule Generation Software Process Model", BIJIT - BVICAM's International Journal of Information Technology, Vol. 5, No. 1, pp. 505-511, Jan. – June, 2013.

[4]. MatLab R2013 Neural Network Tool Box Product Help.

[5]. Abhikriti Narwal, "Empirical Evaluation of Metrics for Component Based Software Systems", International Journal of Latest Research in Science and Technology, Vol. 1, Issue 4, pp.373-378, Nov.- Dec. 2012.

[6]. Amr Rekaby, Ayat Osama, "Introducing Integrated Component-Based Development Lifecycle and Model", International Journal of Software Engineering & Applications, Vol. 3, No. 6, pp. 87-99, Nov. 2012.

[7]. Sandeep Srivastava, "Software metrics and Maintainability Relationship with CK Metrics", International Journal of Innovations in Engineering and Technology, Vol. 1 Issue 2, pp. 76-82, Aug. 2012.

[8]. Anupama Kaur, Himanshu Monga, Mnupreet Kaur, Parvinder S. Sandhu, "Identification and Performance Evaluation of Reusable Software Components Based Neural Network", International Journal of Research in Engineering and Technology, Vol. 1, No. 2, pp. 100-104, March 2012.

[9]. Simrandeep Singh Thapar, Paramjeet Singh, Shaveta Rani, "Challenges to the Development of Standard Software Quality Model", International Journal of Computer Applications, Vol. 49, No.10, pp. 1-7, July 2012.

[10]. G. Shanmugasundaram, V. Prasanna Venkatesan, C. Punitha Devi, "Reusability metrics - An Evolution based Study on Object Oriented System, Component based System and Service Oriented System", Journal Of Computing, Volume 3, Issue 9, pp. 30-38, Sept. 2011.

[11]. Aldeida Aleti, Indika Meedeniya, "Component Deployment Optimisation with Bayesian Learning", ACM Journal, pp. 11-20, June 2011.

[12]. Mostefai Mohammed Amine, Mohamed Ahmed-Nacer, "An Agile Methodology For Implementing Knowledge Management Systems : A Case Study In Component-

Based Software Engineering", International Journal of Software Engineering and Its Applications, Vol. 5, No. 4, pp. 159-170, 2011.

[13]. P. C. Jha, Shivani Bali and P. K. Kapur, "Fuzzy Approach for Selecting Optimal COTS Based Software Products Under Consensus Recovery Block Scheme", BIJIT - BVICAM's International Journal of Information Technology, Vol. 3, No. 1, pp. 318-323, Jan. – June 2011.

[14]. Anju Shri, Parvinder S. Sandhu, Vikas Gupta, Sanyam Anand, "Prediction of Reusability of Object Oriented Software Systems using Clustering Approach", World Academy of Science, Engineering and Technology, Vol. 43, pp. 853-856, 2010.

[15]. G. M. Tere and B. T. Jadhav, "Design Patterns for Successful Service Oriented Architecture Implementation", BIJIT - BVICAM's International Journal of Information Technology, Vol. 2, No. 2, pp. 245-249, July – Dec. 2010.

[16]. Sonia Manhas,Rajeev Vashisht,Reeta Bhardwaj, "Framework for Evaluating Reusability of Procedure Oriented System using Metrics based Approach", International Journal of Computer Applications, Vol. 9, No. 10, pp. 14-19, Nov. 2010.

[17]. V. Lakshmi Narasimhan, P. T. Parthasarathy, M. Das, "Evaluation of a Suite of Metrics for Component Based Software Engineering (CBSE)", Issues in Informing Science and Information Technology, Vol. 6, pp. 731-740, 2009.

[18]. María A. Reyes, Maryoly Ortega, María Pérez, Anna Grimán Luis E. Mendoza and Kenyer Domínguez, "Toward A Quality Model for CBSE", International Conference on Enterprise Information Systems, pp. 101-106, 2009.

[19]. R. Senthil, D. S. Kushwaha, A. K. Misra, "An Extended Component Model and its evaluation for Reliability & Quality", Journal of Object Technology, Vol. 7, No. 7, pp. 109-129, Sept. 2008.

[20]. Yoonjung Choi, Sungwook Lee, Houp Song, Jingoo Park, SunHee Kim, "Practical S/W Component Quality Evaluation Model", ICACT, pp. 259-264, Feb. 2008.

[21]. Mubarak Mohammad, Vasu Alagar, "A Component-Based Software Engineering Approach for Developing Trustworthy Systems", ACTS Report Series, Feb. 2008.

[22]. Anita Gupta, Reidar Conradi, Forrest Shull, Daniela Cruzes, "Experience Report on the Effect of Software Development Characteristics on Change Distribution", Springer Journal, pp. 158–173, 2008.

[23]. Kung-Kiu Lau, Zheng Wang, "Software Component Models", IEEE Transactions On Software Engineering, Vol. 33, No. 10, pp. 709-724, Oct. 2007.

[24]. Net Objective, "Design Patterns: From Analysis to Implementation", Manuals for design patterns explained: A New perspective for Object Oriented Design, 2007.

[25]. Alexandre Alvaro, Eduardo Santana de Almeida, Silvio Lemos Meira, "A Software Component Quality Model: A Preliminary Evaluation", IEEE Proc. of the 32nd EUROMICRO Conference on Software Engineering and Advanced Applications, 2006.

[26]. Parvinder S. Sandhu, Hardeep Singh, "Automatic Reusability Appraisal of Software Components using Neuro-fuzzy Approach", International Journal of Information Technology,Vol. 3,No. 3, pp. 209-215, 2006

[27]. Kilsup Lee, Sung Jong Lee, "A Quantitative Software Quality Evaluation Model for the Artifacts of Component Based Development", Proc. of the 6th IEEE International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, 2005.

[28]. Sajjad Mahmood, Richard Lai, Yong Soo Kim, Ji Hong Kim, Seok Cheon Park, Hae Suk Oh, "A survey of component based system quality assurance and assessment", Elsevier Information and Software Technology, Vol. 47, pp. 693-707, 2005.

[29]. Ramanath Subramanyam and M.S. Krishnan, "Empirical Analysis of CK Metrics for Object-Oriented Design Complexity: Implications for Software Defects", IEEE Transactions on Software Engineering, Vol. 29, No. 4, pp. 297-310, April 2003.

[30]. Simon Haykin, Neural Networks: A Comprehensive Foundation, Pearson Education, 2002.

[31]. Margaretha W. Price, Donald M. Needham, Steven A. Demurjian, "Producing Reusable Object-Oriented Components: A Domain-and-Organization-Specific Perspective", ACM Proc.of the symposium on Software reusability: putting software reuse in context, pp. 41-50, May 18-20, 2001.

[32]. Tullio Vernazza, Giampiero Granatella, Giancarlo Succi, Luigi Benedicenti, Martin Mintchev, "Defining metrics for software components", Proc. of the World Multiconference on Systemics, Cybernetics and Informatics, Vol. 11, pp. 16-23, July 2000.

[33]. John Grundy, Warwick Mugridge, John Hosking, "Constructing Component-based Software Engineering Environments: Issues and Experiences", Elsevier Journal of Information and Software Technology, Vol. 42, No. 2, pp. 117-128, Jan. 2000.

[34]. Neville I. Churcher, Martin J. Shepperd, "Comments on - A Metrics Suite for Object Oriented Design", IEEE Transactions on Software Engineering, Vol. 21, No. 3, pp. 263-265, March 1995.

[35]. R. Geoff Dromey, "A Model for Software Product Quality", IEEE Transactions on Software Engineering, Vol. 21, No. 2, pp. 146-162, Feb. 1995.

[36]. E. Gamma, R. Helm, R. Johnson, J. Vlissides, "Design Patterns: Elements of Reusable Object-Oriented Software", Addison Wesley, 1995

[37]. Shyam R. Chidamber, Chris F. Kemerer, "A metrics suite for Object Oriented Design", IEEE Transactions on Software Engineering, Vol. 20, No. 6, pp. 476-493, June 1994.

[38]. Sidhu Pravneet, “Quality Metrics Implementation In Component Based Software Engineering Using AI Back Propagation Algorithm Software Component”, International Journal of Engineering and Management Sciences, Vol. 3, No. 2, pp. 109-114, 1994.

[39]. James C. Browne, Taejae Lee, John Werth, “Experimental Evaluation of a Reusability-Oriented Parallel Programming Environment”, IEEE Transactions on Software Engineering, Vol. 16. No. 2, pp. 111-120, Feb. 1990.

# A Methodology to Find the Cycle in a Directed Graph Using Linked List

**Shubham Rungta**[1], **Samiksha Srivastava**[2], **Uday Shankar Yadav**[3]and **Rohit Rastogi**[4]

*Submitted in July, 2013; Accepted in September, 2014*

***Abstract - In computer science, cycle detection is the algorithmic problem of finding a cycle in a sequence of iterated function values. The analysis of cycles in network has different application in the design and development in communication systems such as the investigation of topological features and consideration of reliability and fault tolerance. There are various problems related to the analysis of cycles in network among which the most important one is the detection of cycles in graph. In this paper, we proposed SUS_dcycle method which is a detection algorithm for detecting cycle in a directed graph, with the help of linked list in order to discover new lists in run time. This algorithm is used to detect the cycle in any type of directed graph. The proposed algorithm differs from other existing algorithms through its ability to count the total number of cycles present in any type of directed graphs. Also the study of earlier works says that this is a novel approach for the prescribed task and complex problems may use it as a subroutine application for effective results. In advanced computing, time-space trade-off is an important factor to efficiently deal with the problems. This method may solve the above said purpose.***

***Index Terms – Directed graph, Cycle, Linked list, Graph theory, and Data structure.***

## 1.0 INTRODUCTION

### 1.1 Cycle

A cycle is repeating part in the sequence. In computer science, cycle detection is the algorithmic problem of finding a cycle in a sequence of iterated function values [1]. Suppose in a function f(x), if x repeats the same sequence of values once again, then there exist a cycle.

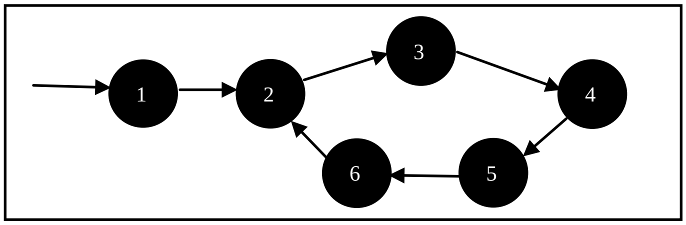


**Figure 1: Example of a cycle with 1, 2, 3, 4, 5, and 6 as vertices of the graph.**

Here f(x) is the function and [x: x is 1, 2, 3, 4, 5, 6, y… in sequence, where y is 2, 3, 4, 5, and 6 in sequence and is repeated], here cycle exists because x repeats the value 2, 3, 4, 5, 6 repeatedly. This paper proposes a SUS's cycle detection algorithm to detect cycles and to find number of cycles in any directed graph whether it is simple or multi digraph. This algorithm is also helpful in fetching out the number of cycles in the whole graph.

## 2.0 EARLIER WORKS IN THIS FIELD

**2.1 Floyd's cycle-finding algorithm**, also called the "tortoise and the hare" algorithm, is a pointer algorithm that uses only two pointers, which move through the sequence at different speeds. The algorithm is named for Robert W. Floyd, who invented it in the late 1960s.[9]

The key insight in the algorithm is that, for any integers $i \geq \mu$ and $k \geq 0$, $x_i = x_{i+k\lambda}$, where λ is the length of the loop to be found. In particular, whenever $i = m\lambda \geq \mu$, it follows that $x_i = x_{2i}$. Thus, the algorithm only needs to check for repeated values of this special form, one twice as far from the start of the sequence as the other, to find a period ν of a repetition that is a multiple of λ. Once ν is found, the algorithm retraces the sequence from its start to find the first repeated value $x_\mu$ in the sequence, using the fact that λ divides ν and therefore that $x_\mu = x_{\mu+2\nu}$. Finally, once the value of μ is known it is trivial to find the length λ of the shortest repeating cycle, by searching for the first position $\mu + \lambda$ for which $x_{\mu+\lambda} = x_\mu$.

The algorithm thus maintains two pointers into the given sequence, one (the tortoise) at $x_i$, and the other (the hare) at $x_{2i}$. At each step of the algorithm, it increases i by one, moving the tortoise one step forward and the hare two steps forward in the sequence, and then compares the sequence values at these two pointers. The smallest value of i> 0 for which the tortoise and hare point to equal values is the desired value ν.

**2.2Richard P. Brent** et al. described an alternative cycle detection algorithm that, like the tortoise and hare algorithm, requires only two pointers into the sequence.[10] However, it is based on a different principle: searching for the smallest power $2^i$ that is larger than both λ and μ. For i = 0, 1, 2, etc., the algorithm compares $x_{2^i-1}$ with each subsequent sequence value up to the next power of two, stopping when it finds a match. It has two advantages compared to the tortoise and hare algorithm: it finds the correct length λ of the cycle directly, rather than needing to search for it in a subsequent stage, and its steps involve only one evaluation of *f* rather the indices of saved sequence than three.

Brent [10] already describes variations of his technique in which values are powers of a number R other than two. By choosing R to be a number close to one, and storing the sequence values

[1, 2, 3,] *ABES Engineering College, Ghaziabad (U.P.), India.*
[4] *Sr. Asst. Professor, CSE Deptt. ABES Engineering College, Ghaziabad (U.P.), India.*
*E-mail: [1]shubhamrungta93@gmail.com,*
*[2]samikshasrivastava607@gmail.com,*
*uday4792@gmail.com and [4]rohit.rastogi@abes.ac.in*

at indices that are near a sequence of consecutive powers of R, a cycle detection algorithm can use a number of function evaluations that is within an arbitrarily small factor of the optimum λ+μ.[12] [13]

**2.3Sedgewick, Szymanski, and Yao** [14] provide a method that uses M memory cells and requires in the worst case only $(\lambda + \mu)(1 + cM^{-1/2})$function evaluations, for some constant c, which they show to be optimal. The technique involves maintaining a numerical parameter d, storing in a table only those positions in the sequence that are multiples of d, and clearing the table and doubling d whenever too many values have been stored.

Several authors have described distinguished point methods that store function values in a table based on a criterion involving the values, rather than (as in the method of Sedgewick et al.) based on their positions. For instance, values equal to zero modulo some value d might be stored.[15][16] More simply, **Nivasch**[11] credits D. P. Woodruff with the suggestion of storing a random sample of previously seen values, making an appropriate random choice at each step so that the sample remains random.

**2.4 Nivasch** [11] describes an algorithm that does not use a fixed amount of memory, but for which the expected amount of memory used (under the assumption that the input function is random) is logarithmic in the sequence length. An item is stored in the memory table, with this technique, when no later item has a smaller value. As Nivasch shows, the items with this technique can be maintained using a stack data structure, and each successive sequence value need be compared only to the top of the stack. The algorithm terminates when the repeated sequence element with smallest value is found. Running the same algorithm with multiple stacks, using random permutations of the values to reorder the values within each stack, allows a time–space tradeoff similar to the previous algorithms. However, even the version of this algorithm with a single stack is not a pointer algorithm, due to the comparisons needed to determine which of two values is smaller.

Any cycle detection algorithm that stores at most M values from the input sequence must perform at least $(\lambda+\mu)(1+\frac{1}{M-1})$function evaluations.[17] [18]

### 3.0 PROPOSED IDEA

### 3.1 Proposed SUS_dcycle Algorithm

**a)** START

**b)** SUS_dcycle algorithm has parameters as information field, variables with their data types and identifiers for starting, processing the paths, temporary allocation and exchanging of the data, variables for counting the cycles in test graph, holding temporary data and functional mechanism to free the unused space.

**c)** Initialize the variables as per the mechanism chosen and allocate the data in them.

**d)** Define the vertices information and edges as per the mechanism/ approach chosen.

**e)** Put each and every node in a list say ‘LIST0’.

**f)** Take out any node from LIST0 as a starting node (if any node exists.)

**g)** Get/start with the first vertex, let A and hold its storage location.

**h)** Now, if there are n directed paths from the first vertex A, then the possible new paths from here are n-1, so create the n-1 data structures (chosen by you) to hold these possible paths. Also remove these discovered node from List0.And whenever a new node discovered during traversal remove it from LIST0.

**i)** Define a mechanism to store the information of all next vertex traversed from the previous vertex and hold it anyhow.

**j)** Loop starts up to all the existing and uncovered paths.

**k)** In a particular path p, compare the new vertex presently being traversed with the all of previously traversed vertices starting from the first vertex in p and check whether the present information is being repeated.

**l)** If yes, then
   We are sure that a cycle exists.
   Store the vertices information comprising this cycle and may print their values as per need.
   Increment the counter by one.

**m)** If no vertex is repeated, then
   We can declare that there is not a cycle in that traversed path p.

**n)** So, go to next possible path starting from first vertex.
   Continue this process till all the paths are covered.
   Loop Ends

**o)** After working with every path from Starting node (‘A’). Check LIST0 if any node is present there, if yes then once again take out any node from remaining node and follow step 8 to 14.

**p)** Print the counter value as the no. of cycles in the test graph and as per need can print the vertices contained in those cycles respectively.

**q)** END

### 3.2 Proposed SUS_dcycle Algorithmwith the Linked List Implementation

**SUS_dcycle (INFO, LINK, START1, START2, LIST0, LIST1, LIST2, PTR1, PTR2, PTR3, ITEM, Counter, FREE(x), TEMP)**

**INFO-**Stores the information field of the node in linked list.

**LINK-**Address field of the node that contains the address of the next node in the linked list.

**LIST0-** List to store all the nodes of a graph. Use of this list is to find out the unreachable nodes (if exist) from a starting node (that we choose randomly from LIST0) in a digraph.

**LIST1-**Linked list to store the base address of all the linked lists formed during runtime.

**LIST2-**Linked list to store nodes discover during traversal.

**START1-**Pointer which points to the first node of linked LIST1.
**START2-**Pointer which points to the first node of linked LIST2.
**Counter-**A global variable to count the number of cycle.
**SAVE**, **PTR**, **PTR2**, **PTR3**,**PTR4**, **and TEMP**: They are the pointer variables.
**ITEM-**It contains the info character.
**FREE(x)-**This function will remove the node x.
**Start with:**

1. Put each and every node in a LIST0.
2. Take out any node from LIST0 as a starting node (if any node exists.).
3. Whenever a new node discovered during traversal just remove it from LIST0.

**Step1:** Insert the first node in linked list LIST2. (Let the first node be A.)

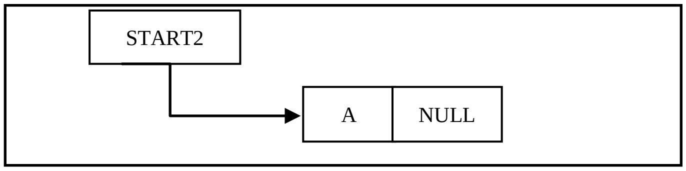


**Fig: 2 Insertion of first**

**Step2:** Put the base address of the LIST2 in LIST1.

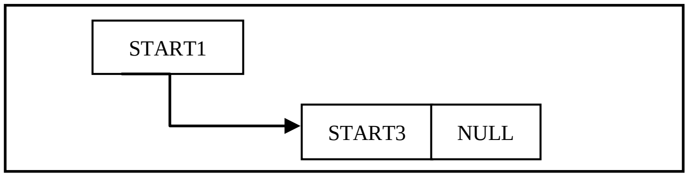


**Figure 3: Insertion**

**Step3:**Now, if there are n directed paths from A, then total number of new linked list will be n-1 and they all will be duplicate of LIST2.

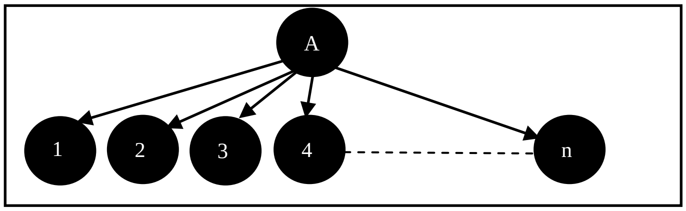


**Figure 4: Example of n directed path from A**

**Step-4:** Now put each next node of the graph directed from the last node into separate linked lists.

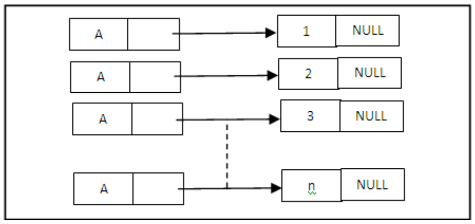


**Figure 5: Insert 1 in LIST2 and others in the copies in LIST2**

**Step-5:** Put the base address of each newly created list into the LIST1 in a consecutive manner.

// Iteration of outer loop
**Step-6:** Repeat Step (7) to (10) while (START1! = NULL)
/*Compare the element at last node of LIST2 with all its previous nodes starting from starting node.*/

**Step 7:** [Initialize the value of pointer PTR and SAVE with the value of pointer START2.]

```
        PTR<-START2
        SAVE<-START2
```

**Step 8**: [Repeat Steps (a) to (b) until (PTR! =NULL)]

```
        a) PTR<-LINK [PTR]
        b) TEMP<-INFO [PTR]
```

**Step 9:** [Initialize PTR2 with START2]

```
        PTR2<-START2
```

/*Compare the last element with all the elements of LIST2.*/

**Step10:** [Repeat the following steps (a) to (b) and 11 to 13 until (PTR2! =NULL)]

```
        SAVE=PTR2
      IF (TEMP=INFO [SAVE])
      THEN
        i.  Counter<- Counter+1
```

/*If the counter is incremented then drop the wholeLIST2 list if cycle exist in addition, the node with the base address of the dropped list is removed from List1 and START1 points the next node to the deleted node in List1*/

ii. [Display the detected cycle by repeating the following step until PTR3! =NULL]

```
          a)  ITEM=INFO [START2]
          b)  DISPLAY [ITEM]
          c)  PTR3=LINK [START2]
[In addition freeing the displayed nodes]
          d)  FREE (START2)
          e)  START2=PTR3
```

iii. [Remove the node containing the base address ofLIST2 list in which cycle occurs or null node appears]

```
a) PTR4=LINK [START1]
b) FREE (START1)
          c) START1=PTR4
      ELSE
```

```
        PTR2=LINK [PTR2]
ENDIF
```

**Step 11:** [Enter the next node directed by the last node in LIST2 using Step (3)-(12) until NULL node encountered] [If NULL is encountered then GOTO STEP 11-(ii)]
After working with every path from Starting node ('A'). Check LIST0 if any node is present there, if yes then once again take out any node from remaining node and follow **Step** 1 to 11.

If no node is left in LIST0 then GOTO Step12.

**Step12:** [The final value of counter will result in total number of cycles in the test graph].

```
        DISPLAY [Counter]
```

**Step 13:** END

### 3.3 Example

Let's take a directed graph with n (V) =5

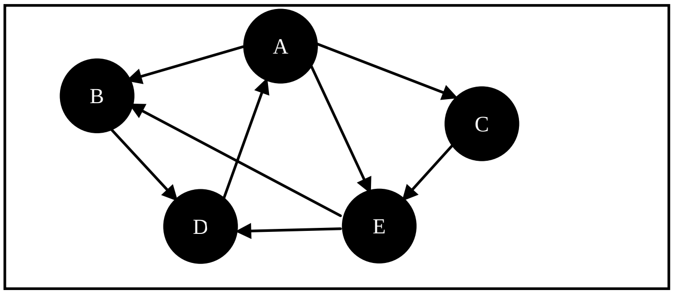


**Figure 6: A directed graph with A, B, C, D and E as its vertices.**

Let us take two-linked list LIST1, LIST2 withSTART1, START2 as their pointers to their base address respectively.
And LIST0 for storing each and every node of digraph (Figure 6)
**Start with:**

| | | | | |
|---|---|---|---|---|
| A | B | E | C | D |

**Figure 7: LIST0 storing all the nodes of digraph (Figure 6)**

Take out any node from LIST0 as a starting node (if any node exists.).
Remember, whenever a new node discovered during traversal just remove it from LIST0.

**Step-1:** Insert the first node in linked list LIST2. Let it be node A. Remove 'A' from LIST0.

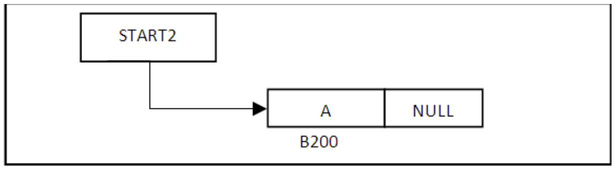


**Figure 8: Linked list LIST2 with A in its first info field.**

In addition, put the base address of this linked list LIST2 in LIST1 in its info field.

In addition, put the base address of this linked list LIST2 in LIST1 in its info field.

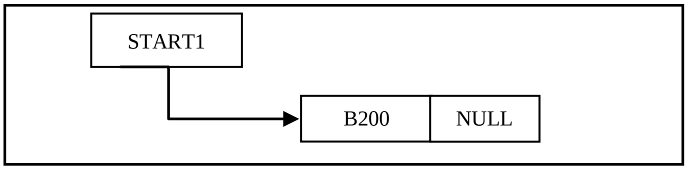


**Figure 9: START1 points to the base address of LIST1 with base address of LIST2 in its info field.**

**Step-2:** Now, there are four directed paths from A i.e. B,C,Eand Dso, total number of new linked lists will be three (4-1=3).And they will be duplicate of lastly implemented linked list. Now Put each next node of the graph directed from the last node (B,C,E&D) into separate linked lists and put the base address of each newly created linked lists into the LIST1 in a consecutive manner. Also, Remove B, C, E & D from LIST0.

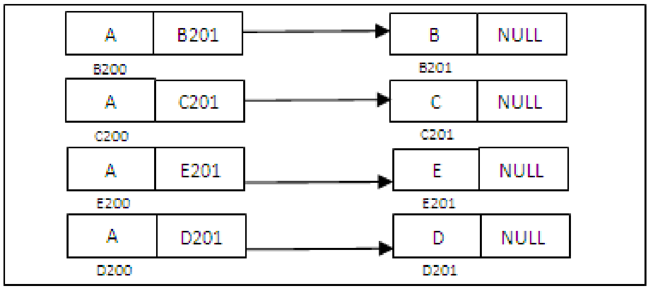


**Figure 10: Insert B in LIST2 and C, E and D in the newly created copies of LIST2.**

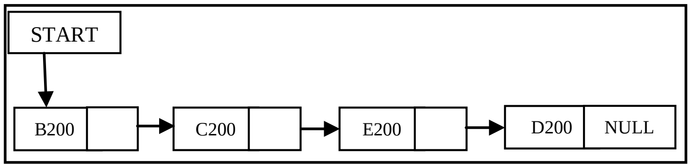


**Figure 11: Insertion of base address of the above-created lists in LIST1**

**Step-3:**Repeat step (4)-(9) until START1! = NULL
**Step-4:** Take PTR as a temporary variable.
PTR=INFO [START1] (Here, PTR = B200)
**Step-5:**Now as PTR points to the LIST2, compare the element at last node of LIST2 with all its previous nodes from starting.
**Step-6:** Since, cycle is not detected; insertion of node directed by B in LIST2 will take place. Since B directs only one node (D), there is no need to discover new lists, if there will be two say x & y then new nodes will be formed with A linked with B and B linked with x and second list will be A linked with B and B linked with y.

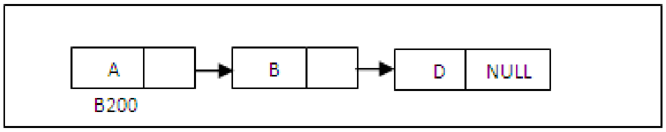


**Figure 12: Insertion of D node in the LIST2**

**Step-7:** use Step-5 and 6 for further traversing path.

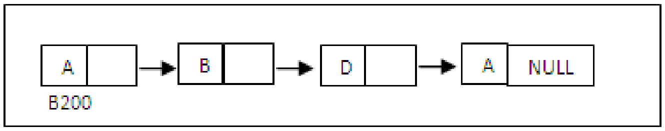


**Figure 13: Insertion of node A in List2**

**Step-8:** Use step-5 and check it has cycle or not.
Here, there is a cycle hence counter incremented and the list displayed and then deleted. In addition, if first info field of the LIST1 detects NULL then there is no cycle, counter does not incremented but list deleted.

**Step-9:** Now, PTR should points to next node of LIST1.
PTR =LINK [PTR]

**Step-10:** Since, there are no nodes left in LIST0, displaying counter results in number of cycle in the graph.

HERE, there are 7 cycles in the graph (Figure 5)
Therefore, using the algorithm we can find the node that forms the cycles and number of cycle in any digraph.

### 3.4 Complexity

### 3.4.1 Worst Case

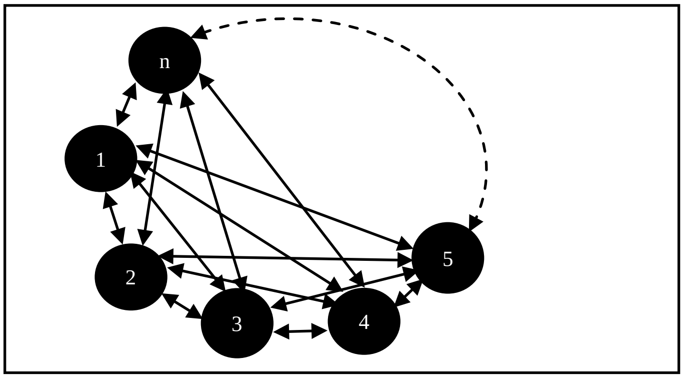


**Figure 14: Example of complete directed graph in order to calculate the worst case of algorithm.**

Let n be the number of vertices in the directed complete graph.
=>T (n) =Pointers assigning+ New lists+ Cycle detection
=>T (n) =O (n-1) +O ((n-1) ^ (n-1)) +O (n-1)
=>T (n) =O ($n^n$)

### 3.4.2 Best Case

=>T (n) = Pointers assigning+ Cycle detection
=>T (n) =O (1) +O (n-1)
=>T (n) =O (n)

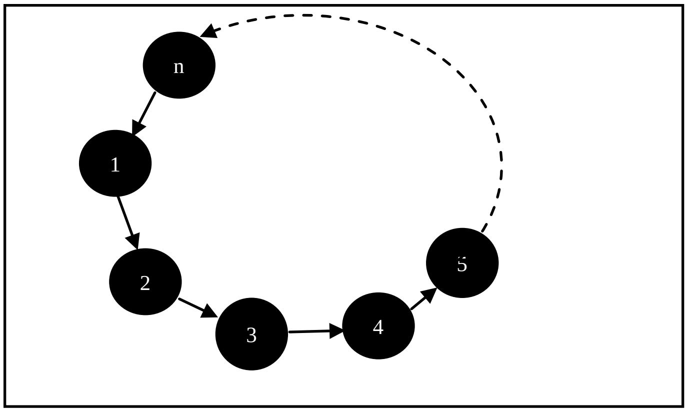


**Figure 15: Example of the directed graph in order to find the best case of algorithm**

## 4.0 APPLICATIONS OF CYCLE

a) Cycle detection may be helpful as a way of discovering infinite loops in certain types of computer programs [2].
b) Use of wait-for graphs to detect deadlocks in concurrent system [3].
c) Periodic configurations in cellular automaton simulations may be found by applying cycle detection algorithms to the sequence of automaton states [4].
d) In cryptographic applications, the ability to find two distinct values $x_{\mu--1}$ and $x_{\lambda+\mu--1}$ mapped by some cryptographic function $f$ to the same value $x_{\mu}$ may indicate a weakness in $f$. For instance, Quisquater and Delescaille [5] apply cycle detection algorithms in the search for a message and a pair of Data Encryption Standard keys that map that message to the same encrypted value; Kaliski, Rivest, and Sherman [6] also use cycle detection algorithms to attack DES. The technique may also use to find a collision in a cryptographic hash function.
e) Analysis of electrical networks, periodic scheduling, analysis of chemical and biological pathways.

## 5.0 RECOMMENDATION

The proposed SUS's cycle detection method is not only an easier method to detect cycle in any digraph but also very helpful in finding number of cycles in the graph. So, thisalgorithmcan be used in detecting infinite loops in various computer programs, analysis of electrical networks, periodic scheduling and in many more places where there is a need to detect cycle.

## 6.0 LIMITATION

In this paper, the cycle detection done with the help of linked list. However, this method is easier to implement, in worst case its complexity reaches to O (n^n), which is much higher and because of the formation of new lists in run time it needs large space to act upon. Although, this algorithm removes that linked lists in which traversal is completed, computers with large space used here to execute the proposed algorithm.

## 7.0 FUTURE SCOPE

The above-proposed algorithmhelps in detecting cycle in any digraph takes much space to execute and its complexity is much higher in case of worst case. Therefore, using this method and logic, in future new logics may define to overcome the complexity and space problems.

## 8.0 CONCLUSION

In this paper, we have developed a new technique to detect the number of cycles in a directed graph and showed the entire traversed node that forms cycle by displaying it at the time of using counter that incremented at the time of detecting cycle. In addition, when the cycle detects, the same time traversed list is deleted hence, that saved the space. Insertion of nodes from directed graph inserts in the singly linked list. The procedure of this algorithm is much easier to implement and execute for digraph and directed multigraph. This algorithm can be beneficial in detecting infinite loops in certain computer program [2].The proposed algorithm expected to be of great interest in theory and practice alike.

## 9.0 NOVELTY IN THIS PAPER

In this paper, we have not only presented the new way to detect the cycle in any simple or strongly connected digraph but also presented the new way to count number of cycles in the graph. We used here an efficient data structure named linked list to form new nodes in run-time. It is also used in storing the base address of the newly form linked list in run-time. Hence, we can say that all the application of this algorithm executes in run-time.

## 10.0 ACKNOWLEDGEMENT

The completion of this paper would not have been possible without the help and guidance of our rev. HOD Prof. Dr.R.Radhakrishnan who was always there to support us and correct us at every step. We would also like to thank our Prof. Rohit Rastogi who guided us on this work. Next, we would like to thank our friends, family and seniors for their motivation and encouragement. Last but definitely not the least we would thank the almighty god without whose grace this paper would not have achieved success.

## 11.0 REFERENCES

[1]. Piotr Puczynski, "The cycle detection algorithms", Wroclaw University of Technology, Faculty of Management.

[2]. Van Gelder, Allen (1987), "Efficient loop detection in Prolog using the tortoise-and-hare technique", Journal of Logic Programming 4 (1): 23–31, doi: 10.1016/0743-1066(87)90020-3.

[3]. Silberschatz, Abraham; Peter Galvin, Greg Gagne (2003). Operating System Concepts. John Wiley & Sons, INC. p. 260. ISBN 0-471-25060-0.

[4]. Nivasch, Gabriel (2004), "Cycle detection using a stack", Information Processing Letters 90 (3): 135–140, doi: 10.1016/j.ipl.2004.01.016.

[5]. Quisquater, J.-J.; Delescaille, J.-P., "How easy is collision search? Application to DES", Advances in Cryptology – EUROCRYPT '89, Workshop on the Theory and Application of Cryptographic Techniques, Lecture Notes in Computer Science 434, Springer-Verlag, pp. 429–434.

[6]. Kaliski, Burton S., Jr.; Rivest, Ronald; Sherman, Alan T. (1988), "Is the Data Encryption Standard a group? (Results of cycling experiments on DES)", Journal of Cryptology 1 (1): 3–36, doi: 10.1007/BF00206323.

[7]. H.D. Rozenfeld et al. Statistics of cycles: how loopy is your network? J.Phys. A: Math.Gen. 38:4589, 2005.

[8]. M. Medard and S. S. Lumetta. Network reliability and fault tolerance. In J. Proakis, editor, Wiley Encyclopaedia of Engineering.

[9]. Floyd describes algorithms for listing all simple cycles in a directed graph in a 1967 paper: Floyd, R.W. (1967), "Non-deterministic Algorithms", J. ACM14 (4): 636–644, doi:10.1145/321420.321422.

[10]. Brent, R. P. (1980), "An improved Monte Carlo factorization algorithm", BIT20 (2): 176–184, doi: 10.1007/BF01933190.

[11]. Nivasch, Gabriel (2004), "Cycle detection using a stack", Information Processing Letters90 (3): 135–140, doi: 10.1016/j.ipl.2004.01.016.

[12]. Schnorr, Claus P.; Lenstra, Hendrik W. (1984), "A Monte Carlo Factoring Algorithm With Linear Storage", Mathematics of Computation (American Mathematical Society) 43 (167): 289–311, doi:10.2307/2007414, JSTOR 2007414.

[13]. Teske, Edlyn (1998), "A space-efficient algorithm for group structure computation", Mathematics of Computation67 (224): 1637–1663, doi: 10.1090/S0025-5718-98-00968-5.

[14]. Sedgewick, Robert; Szymanski, Thomas G.; Yao, Andrew C.-C. (1982), "The complexity of finding cycles in periodic functions", SIAM Journal on Computing11 (2): 376–390, doi: 10.1137/0211030.

[15]. Van Oorschot, Paul C.; Wiener, Michael J. (1999), "Parallel collision search with cryptanalytic applications", Journal of Cryptology12 (1): 1–28, doi: 10.1007/PL00003816.

[16]. [16] Quisquater, J.-J.; Delescaille, J.-P., "How easy is collision search? Application to DES", Advances in Cryptology – EUROCRYPT '89, Workshop on the Theory and Application of Cryptographic Techniques, Lecture Notes in Computer Science 434, Springer-Verlag, pp. 429–434.

[17]. Fich, Faith Ellen (1981), "Lower bounds for the cycle detection problem", Proc. 13th ACM Symp. Theory of Computation, pp. 96–105, doi: 10.1145/800076.802462.

[18]. Allender, Eric W.; Klawe, Maria M. (1985), "Improved lower bounds for the cycle detection problem",

Theoretical Computer Science36 (2–3): 231–237, doi: 10.1016/0304-3975(85)90044-1.

[19]. Pollard, J. M. (1975), "A Monte Carlo method for factorization", BIT15 (3): 331–334, doi: 10.1007/BF01933667.

[20]. Pollard, J. M. (1978), "Monte Carlo methods for index computation (mod p)", Math. Comp. (American Mathematical Society) 32 (143): 918–924, doi: 10.2307/2006496, JSTOR 2006496.

[21]. Kaliski, Burton S., Jr.; Rivest, Ronald L.; Sherman, Alan T. (1988), "Is the Data Encryption Standard a group? (Results of cycling experiments on DES)", Journal of Cryptology1 (1): 3–36, doi: 10.1007/BF00206323.

**Mr. Shubham Rungta** is from Ghughli, a town in district Maharajganj (U.P-India). He had received his high school education from Don Bosco School, Nainital (Uttarakhand-India) and intermediate from Gorakhpur (U.P.). At present, he is an IV year student of Computer Science Engineering in ABES Engineering College, Ghaziabad (U.P. - India). His area of interests includes several languages such as C, JAVA, Web Technologies and several subjects such as DBMS, Data Structure, Graph theory, Software Engineering. He is an author certified with two top most journals namely, Springer and IEEE. He is a philanthropist as an active member of NGO named Help Us to Help Child (HUHC).

**Ms. Samiksha Srivastava** is currently pursuing her graduation in B.Tech in Computer Science and Engineering (IV year)from ABES Engineering College, Ghaziabad (U.P.-India), affiliated to Uttar Pradesh Technical University.Her field of interest includes network security,DBMS, Website Designing, Date Compression and Data Structures. She is an active member of NGO named.

**Mr. Uday Shankar Yadav** is currently an IV year student of Computer Science Engineering in ABES Engineering College, Ghaziabad (U.P. - India), presently affiliated to Uttar Pradesh Technical University. His area of interests includes DBMS, Data Mining, Pattern Recognition, Date Compression, Soft Computing, and Data Structure. Currently, he completely focused upon the field of Graph theory.

**Mr. Rohit Rastogi**received his B.E. degree in Computer Science and Engineering from C.C.S.Univ. Meerut in 2003, the M.E. degree in Computer Science from NITTTR-Chandigarh (National Institute of Technical Teachers Training and Research-affiliated to MHRD, Govt. of India), Punjab Univ. Chandigarh in 2010.

He was Asst. Professor at IMS College, Ghaziabad in computer Sc. Dept. His research interests include Data ware Housing and Data Mining, Design Analysis of Algorithm, Theory of Computation & Formal Languages and Data Bases.

He is a Sr. Asst. Professor of CSE Dept. in ABES Engineering. College, Ghaziabad (U.P.-India), affiliated to Gautam Buddha Tech. University and Mahamaya Tech. University (earlier Uttar Pradesh Tech. University) at present and is engaged in Clustering of Mixed Variety of Data and Attributes with real life application applied by Genetic Algorithm, Pattern Recognition and Artificial Intelligence.

He has served as the technical reviewer of 7 papers in 3rd International Conference on Computing, Communications and Informatics (IC3-2014) at GCET, Greater Noida, NOIDA, India on September, 24-27, 2014 And Currently working as the reviewer for the SPICES-2015 at NIT Kerala, Kojhicode for international conf. of Signal Processing and Communication…

Also currently working as the reviewer in the technical reviewer committee for theINDIA-2015 is Second International Conference on Information System Design and Intelligent Applications organized by Faculty of Engineering, Technology and Management, University of Kalyani, Kalyani-741235, West Bengal, India.

He has authored/co-authored, participated and presented research papers in various Science and Management areas in around 40 International Journals and International conferences including prestigious IEEE and Springer and 10 national conferences including SRM Univ., Amity Univ. and Bharti Vidyapeetha etc. He has guided five ME students in their thesis work and students of UG and PG in around 100 research papers. He has developed many commercial applications and projects and supervised around 30 B.E. students at graduation level projects.

At present, he is a Sr. Asst. Professor of CSE Dept. in ABES Engineering. College, Ghaziabad (U.P.-India), affiliated to Gautam Buddha Tech. University and Mahamaya Tech. University (earlier Uttar Pradesh Tech. University).

His research interests include Data ware Housing and Data Mining, Design Analysis of Algorithm, Theory of Computation & Formal Languages and Data Bases. At present, He is engaged in Clustering of Mixed Variety of Data and Attributes with real life application applied by Genetic Algorithm, Pattern Recognition and Artificial Intelligence.

Also, He is preparing some interesting algorithms on Swarm Intelligence approaches like PSO, ACO and BCO etc.

BIJIT - BVICAM's International Journal of Information Technology
Bharati Vidyapeeth's Institute of Computer Applications and Management (BVICAM), New Delhi (INDIA)

# Secure and Efficient Voting Based Localization Scheme for Wireless Sensor Networks

**Nirmala M. B**[1], **A. S. Manjunath** [2] and **Rajani. M**[3]

*Submitted in March, 2014; Accepted in September, 2014*

***Abstract - Many sensor network applications require sensor node to obtain their locations correctly. Various techniques have been proposed to locate regular sensors based on some special nodes called anchor nodes, which are supposed to know their locations. Providing a certain degree of localization accuracy at the presence of malicious beacons becomes a very challenging task. In this paper, a secure and efficient voting based localization scheme is proposed to mitigate the above impact. In this scheme voting based technique gives a search region in which sensor nodes are present, and then in search region trilateration is applied to know the position of sensor nodes. The communication between anchor and sensor nodes is authenticated and secured by encryption. The proposed scheme can provide very good localization accuracy with the reduced computational cost in presence of malicious nodes. This scheme is resistant to various attacks.***

***Index Terms – Wireless Sensor Networks (WSNs), Secure Localizations, Voting Based Method, Trilateration.***

## NOMENCLATURE

Wireless Sensor Networks (WSNs), Angle of Arrival (AoA), Time Difference Of Arrival (TDoA), Time of Arrival (ToA), Received Signal Strength Indicator (RSSI), cluster-based Minimum Mean Square Estimation (CMMSE), Attack-resistant Minimum Mean Square Estimation (ARMMSE), Least Median square(LMds).

## 1.0 INTRODUCTION

Wireless Sensor Networks (WSNs) is a significant technology attracting considerable research. It is experiencing an explosive growth similar to the internet, this is largely due to the attractive flexibility of anytime, anywhere network access enjoyed by both users and service provider. Knowledge of position of the sensing nodes in a Wireless Sensor Network is a necessary part of many sensor network operations and applications.In hostile environment knowing the position of the sensor is very difficult. The process of determining the position of the sensor nodes in WSNs is defined as localization (location estimation). Sensor node uses anchor node to calculate its location. Anchor nodes are aware of their position through GPS or before deployment and exchange its location information with sensor nodes. The basic idea in D.Liu et al.,[1] is, nodes measure distances to their neighbours and share their position information with them to compute their positions. Sensor node whose position has been uniquely determined can act as a new anchor node to localize other nodes by sharing its position with its neighbours. This iterative process continues until all nodes are localized.

Secure localization as discussed in Jianqing at et al.,[2] is necessary as sensor nodes may be deployed in hostile environments where malicious adversaries attempt to spoof the locations of the sensors by attacking the localization process. For example, an attacker may alter the distance estimations of a sensor to several reference points, or replay beacons from one part of the network to some distant part of the network, thus providing false localization information. Hence, the location estimation is performed in a secured way, even in the presence of attacks. Furthermore, adversaries can compromise the sensor devices and force them to report a false location to the data collection points. Therefore, a secure positioning mechanism is required.

Localization has an endless array of potential applications in both military and civilian applications as discussed in John et al.,[3], including land-mine detection, battlefield surveillance, target tracking, environmental monitoring etc, as discussed [21][23][24][25]. There are many advantages of knowing the location information of sensor nodes. Location information is needed to identify the location of an event of interest like the location of enemy tanks in a battlefield, the location of a fire, target-tracking applications for locating survivors in debris, or enemy tanks in a battlefield.

In this paper a secured efficient localization scheme is proposed based on voting and trilateration method for location discovery. In sensor networks voting method provides us the portable region where unknown node is present. After finding the search area trilateration is applied to find the accurate position. Trilateration is a process of determining absolute position or relative location of point by measurement of distance using the geometry of circle, spheres or triangles. In contrast to triangulation it does not involve the measurement of angles. In two-dimensional geometry, it is known that if a point lies on two circles then the circle centers and the two radii provide sufficient information to find one location. In three-dimensional geometry, when it is known that a point lies on the surfaces of three spheres, then the centers of the three spheres along with their radii provide sufficient information to find the possible locations. There are many other methods available to compute the actual location like Triangulation using AoA as references and Multilateration based on the TDoA where overhead is more compared to Trilateration.

In section 2 literature survey on voting based scheme and other schemes is discussed. Section 3 gives the detail discussion of

[1, 2, 3] *Department of Computer Science, Siddaganga Institute of Technology, Tumkur 572103, Karnataka, India*
*E-mail: [1]nirmalamb@gmail.com, [2]asmanju@gmail.com and [3]rajani10.manju@gmail.com*

the proposed scheme. Section 4 discusses the various type of attacks, analysis of threats to overcome these attacks. Section 5 discusses about computational complexity of our proposed scheme compare to other secure localization scheme.

## 2.0 RELATED WORK

A number of secure localization schemes have been proposed to estimate the location of sensor and protect the anchor nodes, Some of them defeat attacks by detecting and blocking malicious beacons as discussed in Chin et al.,[16], Jinfang et al., [17], Ning Yu[18]. As in Avinash et al.,[11] there are many approaches in localization a) Direct approaches: This is also known as absolute localization. The direct approach itself can be classified into two types: Manual configuration and GPS-based localization. The manual configuration method is very expensive. It is neither practical nor scalable for large scale WSNs and in particular, does not adapt well for WSNs with node mobility. On other hand, in the GPS-based localization method, each sensor is equipped with a GPS receiver. This method adapts well for WSNs with node mobility and it is not economically feasible to equip each sensor with a GPS receiver since WSNs are deployed with hundreds of thousands of sensors. b) Indirect approaches: The indirect approach of localization is also known as relative localization. In this approach, a small subset of nodes in the network, called the anchor nodes is used. It is classified into the following two categories Range-based and Range-free localization. Range-based localization depends on the assumption that the absolute distance between a sender and a receiver can be estimated by one or more features of the communication signal from the sender to the receiver like AoA, RSSI, ToA and TDoA. Range-free localization never tries to estimate the absolute point to point distance based on received signal strength. This greatly simplifies the design and cost effective.

Some schemes utilize clustering algorithm in localization systems to mitigate the impact of malicious attacks. Wang et al. proposed a CMMSE [4] which uses an MMSE to identify and construct a consistent location reference set for the final location estimation. However, the random selection of initial location references makes CMMSE obtain different results in different runs, and might cause more rounds of execution failure. Along the same line, Misra et al. proposed CluRoL [4], which clusters intersections of reference circles to filter out malicious beacon signals but CluRoL is very slow, requires high computation and storage overheads.

A LMdS approach was proposed in [5] to solve the localization problem for scenarios where less than 50% of the nodes are malicious. This method shares similarity with the random sample consensus (RANSAC) algorithm [6], as it uses several subsets of nodes to identify candidate locations, and then chooses the solution that minimizes the median of the residues. These methods localize the nodes with small error as long as the fraction of malicious nodes is not too large. However, the memory requirement and computational cost of running these algorithms is high and can be difficult to meet in resource limited applications.

Loukas lazos et al.[7] present a distributed SeRLoc based on a two-tier network architecture that allows sensor to passively determine their location without interacting with other sensors. The paper also shows that SeRLoc is robust against known attacks on WSNs such as the wormhole attack, the Sybil attack and compromise of network entities. But in this sensor estimates its location as the center of gravity of the overlapping region, which is difficult to estimate.

Monte Carlo based approach for localization was proposed in [8], a fixed number of candidate sample locations that satisfy a constraint on the maximum velocity of the nodes are randomly generated. Samples that are inconsistent with the measurements obtained from anchor nodes are filtered out and

a final estimate of location is found by averaging the remaining samples. The localization accuracy of the algorithm is low. These algorithms did not consider the presence of malicious anchor nodes in the network.

D. Liu, p. Ning et al.,[1] proposed a ARMMSE in which paper two methods to tolerate malicious attacks against beacon-based location discovery in sensor networks have been introduced. The first method filters out malicious beacon signals on the basis of the "consistency" among multiple beacon signals, while the second method tolerates malicious beacon signals by adopting an iteratively refined voting scheme. Both methods can survive even if the attacks bypass authentication, provided that the benign beacon signals constitute the majority of the "consistent" beacon signals. In an extreme case, if all the beacon nodes are compromised, these techniques will fail.

Chen et al.,[19], Sohail et al.,[20] propose localization algorithms based on genetic algorithm and bio inspired computing respectively where computation cost is high.

Our proposed scheme takes a distinct approach by protecting the location privacy of sensor nodes, preventing inaccurate and false location information. Decreasing the computation cost by reducing communication overhead and reduces the location estimation error with no extra localization equipment being employed.

## 3.0 PROPOSED SECURE LOCALIZATION SCHEME

This section gives the detailed description about the proposed secure localization scheme. proposed secure localization scheme is based on voting and trilateration method. Voting based method provides a search region where the sensor node exists. Once the region of sensor node existence is found, trilateration is applied to find the exact location of a sensor node.

Our proposed scheme is purely based on a set of location references, however this scheme is range-independent localization scheme. The location references are taken from set of anchor nodes, so there is no extra communication overhead involved when compared to the other range based localization schemes as discussed in Avinash et al.,[11]. We propose a new key establishment mechanism to establish a symmetric key between the sensor node and anchor nodes to transmit the location information securely to the sensor node. Voting based method finds the overlapping region, if more than three anchor nodes are in the overlapping region of the sensor node, any three anchor nodes are selected and trilateration is applied to calculate the sensor node location. The anchor nodes encrypt the location information of their's and send it to sensor node. Sensor node uses three anchor node locations to compute its position. Network model assumption is given in section 3.1.

### 3.1 Network model

Sensor network consists of sensor nodes. We assume that a set of sensor nodes $S_i = S_{1,}\ldots,S_n$ and a set of anchor nodes $A_j = A_{1,}\ldots,A_m$. The number of anchor nodes $m$ deployed is less than 1/4th the of sensor nodes $n$. We assume that the anchor nodes know their positions accurately (since they are GPS enabled or by other means). Sensor nodes depend on anchor nodes to compute their positions. All the sensor nodes are deployed in the region where its communication range lies within the range of three or more anchor nodes. We consider the anchor nodes which are static and the sensor nodes can be mobile or static. The voting based method and trilateration method is discussed in 3.2 and security scheme in 3.3.

### 3.2 Location estimation

In this section we discuss about our proposed location estimation scheme based on voting scheme and trilateration. In our proposed scheme the location is calculated based on the anchor nodes location information. The anchor nodes broadcast the location information to the sensor nodes. Based on the number of anchor nodes from which the sensor nodes is able to receive the information vote is collected. To illustrate this we consider an example as shown in figure (1). This figure explains both voting based method and trilateration method used to calculate the location of a sensor node.

Fig(a) shows the set of anchor nodes and sensor nodes deployed in an hostile area. Where sensor nodes have to calculate their location with the help of anchor nodes who know their location information in prior. The figure also shows the communication range of each anchor node. Fig (b) chooses the intersection range of three anchor nodes $a_1$, $a_2$, $a_3$ in which the sensor node $s_1$ lies. Around this intersection region an $N{\times}N$ grid is formed and split them into a $N{\times}N$ cells as shown in fig(c). each cell will have the communication range of selected anchor nodes. The anchor nodes which have maximum intersection are considered, take the intersection of communication range of those anchor nodes and split them in to number of $N \times N$ cells. Each cell will have communication range of selected anchor node. Take each location reference as vote. Votes in each cell indicate the number of anchor nodes with in the communication range. Initially all cells will have the vote zero. If any anchor node communication range lies in that cell, the vote count is increased by 1. Fig(d) shows the vote count of each cell and vote count for sensor node $s_1$ which lies within the communication range of three anchor nodes $a_1$, $a_2$, $a_{3.}$ Its vote count is three. Now the sensor node $s_1$ tries to calculate its location using trilateration as show in fig(e). Here three anchor nodes $a_1$, $a_2$, $a_3$ will be considered. Calculate the distance between any two anchor nodes. To simplify the calculations, the equations are formulated so that the nodes (centers of the spheres) are on the $z = 0$ plane. and also the formulation is such that one center is at the origin, and one other is on the *x*- axis, using this calculate *x*,*y* and *z* value, this gives sensor node position.

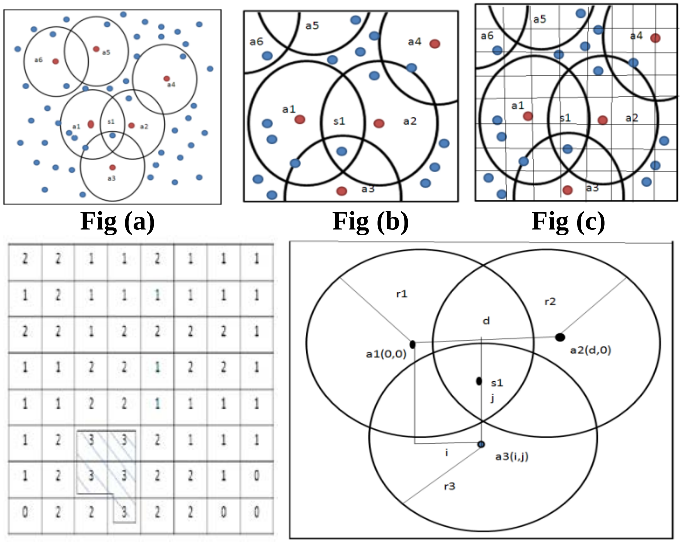


Fig (a) Fig (b) Fig (c)

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 2 | 2 | 1 | 1 | 2 | 1 | 1 | 1 |
| 1 | 2 | 1 | 1 | 1 | 1 | 1 | 1 |
| 2 | 2 | 1 | 2 | 2 | 2 | 2 | 1 |
| 1 | 1 | 2 | 2 | 1 | 2 | 2 | 1 |
| 1 | 1 | 2 | 2 | 1 | 1 | 1 | 1 |
| 1 | 2 | 3 | 3 | 2 | 1 | 1 | 1 |
| 1 | 2 | 3 | 3 | 2 | 2 | 1 | 0 |
| 0 | 2 | 2 | 3 | 2 | 2 | 0 | 0 |

Fig (d) Fig (e)

**Figure 1: (a) Secure network k. (b) Intersection of anchor node. (c) N × N grid in anchor node intersection region. (d) Applying voting technique. (e) Trilateration method.**

### 3.3 Security scheme

This section describes the security scheme used to secure the localization information. We assume that before deployment, the sensor node and the anchor are stored with a key $k_0$. Each sensor node is preloaded with its id i,e $s_{id}$ and a cryptographic hash function $h(\bullet)$ Immediately after the deployment of the anchor nodes and sensor nodes. The sensor node send $S_{id}$ i,e sensor id and random number $r_n$ generated by sensor node, encrypted with symmetric key $k_0$. Anchor node decrypt the $s_{id}$ and $r_\mathrm{n}$ with the key $k_0$. when the sensor node wants to know its position, sensor node will generate a new secrete key, encrypt the key $s_{ki}$ with $r_\mathrm{n}$ send it to anchor node. Anchor node initiate the communication, then sensor node send $E_{sk}(s_{id}$ and $h(\mathrm{r_n}))$ to anchor node. Anchor node decrypts $(s_{id}$ *and* $h(r_n))$ and compare $h(r_n)$ with previously stored value. If the received hash is same as the computed hash, then the sensor node is authenticated and the anchor node will send the location information to sensor node encrypted with $s_{ki}$. Figure 2 explains this security mechanism used to secure the localization information.

## 4.0 ATTACKS

Node compromise is the most fundamental attack in WSN that leads to other kinds of attacks[22]. It occurs when an attacker. gains control of a node in the WSN. With compromised node, an attacker can alter the node to listen information in the WSN, revoke legitimate nodes, input malicious data and cause internal attacks, e.g., DoS attack.

A replay attack is the easiest and most commonly used by attackers. Specifically, when an attacker’s capability is limited, i.e., the attacker cannot compromise more than 1 node. In a replay attack, the attacker merely jams the transmission between a sender and a receiver and later replays the same message, posing as the sender. If an adversary manages to capture a node and extract the authentication/encryption keys, it can produce a large number of replicas having the same identity (ID) from the captured node and integrate them into the WSN at chose locations, which is called the node replication attack.

Security scheme for location information

Initialization:

1. Sensor node $s_{id}$ chooses a random number sends $E_{k0}(s_{id} + r_n)$ to anchor.
2. Anchor node which are in the communication range of $s_{id}$, $D_{k0}(E_{k0}(s_{id} + r_n))$ stores the $s_{id}$ and $r_n$ .

Key exchange phase:

1. Later whenever the sensor node wants to know the location information, generates key $s_k$ encrypt with $r_n$ and send it to anchor node which have been selected for location estimation based on voting.
2. Anchor node sends acknowledgment for the received message.
3. Sensor send ($E_{sk}(s_{id}$ and $h(\mathrm{r_n})$) to anchor node.
4. Anchor node $D_{sk}(E_{sk}(s_{id}$ and $h(\mathrm{r_n})$), computes $h(\mathrm{r_n})$ and compare with the previously stored $h'(\mathrm{r_n})$ values. If both are same then, it encrupts $E_{sk}(L_a)$ sends it to anchor node.

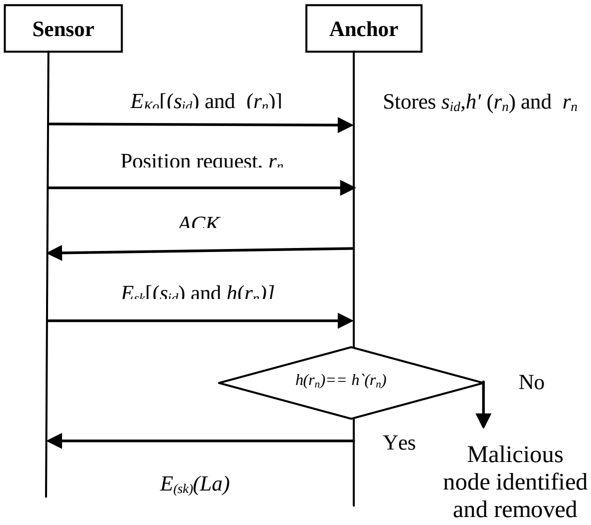

Sensor
Anchor
$E_{K0}[(s_{id})$ and $(r_n)]$
Stores $s_{id}$,$h'$ $(r_n)$ and $r_n$
Position request, $r_n$
ACK
$E_{sk}[(s_{id})$ and $h(r_n)]$
$h(r_n)$== $h`(r_n)$
No
Yes
Malicious node identified and removed
$E_{(sk)}(La)$

**Figure 2: security mechanism for secure location information exchange**

ALGORITHM: Secure voting based localization scheme.

1. Anchor nodes $A_i$ where $i$=1,2,…$m$ within the communication range of the sensor nodes, broadcast the message.
2. Initially sensor nodes set the vote count to zero i.e $v$=0.
3. As it hear the anchor nodes it count gets incremented, it has to hear from at least *3* anchor nodes. If 3 anchor nodes, are in the overlapping region, then the vote count is 3.
4. Sensor nodes generates key $s_k$, $E_{rn}(s_k)$ , encrypts key $s_k$ member with random number and sends it to anchor nodes.
5. Anchor nodes send the acknowledgment for the received message.
6. Sensor nodes sends $E_{sk}(s_{id}+h(r_n))$ to anchor nodes.
7. Anchor nodes sends decrypts$(s_{id}+h(r_n))$ with key $s_k$ which was previous send, computes $h(r_n)$ with the previously stored $r_n$ value and computes $h(r_n)$ with encrypts $E_{sk}(L_A)$ to sensor nodes.
8. Sensor nodes decrypts $L_A$ which as co-ordinands values of $L_i(x_i,y_i)$, $L_{i+1}(x_{i+1},y_{i+1})$, $Li(x_{i+2},y_{i+2})$,…
9. Apply trilateration

a) Consider that all three centers are in the plane $z = 0$. $a_1(0,0)$ is at origin, $a_2(d,0)$ at $x$ axis.

b) To find sensor node position calculate($x$, $y$, $z$)

$$x = \frac{r_1^2 - r_2^2 + d^2}{2d}$$

$$y = \frac{r_1^2 - r_3^2 + i^2 + j^2}{2j} - \frac{i}{j}x$$

$$z = \pm\sqrt{r_1^2 - x^2 - y^2}$$

the adversary replicates one or more sensor nodes, it can execute the malicious operations. For instance, the replicas may inject false localization information into the WSN.

In a sybil attack, a node claims multiple identities in the network. When launched on localization, localizing nodes can receive multiple location references from a single node leading to incorrect location estimation. The Wormhole Attack establishes a direct link between two points in the network. The wormhole attack is very difficult to detect, since it can be launched without compromising any host.

In proposed scheme the authentication is used to identify the authenticated and malicious nodes. In our scheme as hashed random numbers are exchanged whenever sensor node encounters the anchor nodes for communication. Anchor node after receiving the random number verifies it with earlier saved value. If those two values are same then only it sends its location information to sensor nodes. Thus the scheme allows communication between the authenticated nodes thereby preventing above attacks.

As the location information is encrypted with the secret key location information will be secured and it will be difficult for attacker to hack. Table 1 gives summary of various security attacks addressed by our proposed secure voting based scheme scheme compared to other existing schemes.

## 5.0 PERFORMANCE ANALYSIS

The LMdS approach requires a certain minimum number of subsets of nodes $M_1$, which increases as the percentage of malicious nodes increases, in order to ensure that one estimate is the correct estimate with very high probability. An LS estimate needs to be found for each of these subsets, which is computationally expensive. The computation complexity associated with the LMdS method is calculated using the linear least squares (LLS) algorithm described in [9]. LMdS algorithm first performs $M_1$ LLS on different subsets of size $n$ giving a computational complexity of□( $M_1n$). Comparing the computational complexity of the secure voting based method with CluRoL. Proposed scheme has a computational complexity of $O(n^2)$, which is much less than that of $O(n^4 \log n)$ where $n$ is the number of location references provided. This shows that scheme voting based method is efficient compared to CluRoL. Comparing the our scheme and gradient descent based scheme, we can see that they have similar run time. But gradient descent works well only when all received signals converges. If distance between the sensor node and the anchor node increases then the localization error also increases with high computational cost. In secure voting based method as it requires fewer reference points computational complexity is low. Table 2 shows the comparison of computational complexity of various algorithms.

Figure 3 illustrates the key storage overhead PVFS[14] and voting based scheme. PVFS requires storage of four times more keys in its key assignment process compared to voting based scheme. Voting based method requires fewer location references in its localization process, hence the keys required is also minimal.

We compare our secure voting based scheme experimentally with LMds and Gradient descent approach. Simulation is carried out with varying network size of 100 to 500 sensor nodes and 10 to 50 anchor nodes with a deployment region of 600m × 600m. The deployment region is divided into a 10 square grid with each cell of size 60m × 60m.

Figure 4 shows the run time required to achieve a desired localization accuracy comparing the localization errors with gradient descent approach and least mean squares. Localization error of our proposed method is approximately eight times lower compared to LMdS method. Comparing our secure voting based scheme with gradient descent based scheme, they have similar localization accuracy but in gradient descent based approach as the distance increases the localization error also increases. So proposed secure voting based scheme is efficient compared to other schemes.

Figure 5 explains the time taken for localization of different network sizes varying from 100 to 500 nodes by varying the number of anchor nodes. Simulation result shows that proposed

| Algorithm | Localization Attacks | | | | |
|---|---|---|---|---|---|
| | Wormhole | Sybil | Replication | Node compromise | Replay |
| SeRLoc | Y | Y | N | N | N |
| Attack Resistant Location Estimation | N | Y | N | N | N |
| Our proposed Secure Efficient Voting Based Localization Scheme | Y | Y | Y | Y | Y |

**Table 1: Summary of security attacks addressed by each algorithm**

| Method | Complexity |
|---|---|
| Least median Square | $\theta(M_1n)$ |
| CluRoL | $O(n^4 \log n)$ |
| Gradient descent | $\theta(Mn)$ |
| Voting based scheme | $\theta(N_1^2n)$ |

**Table 2: Comparison of run time complexity of algorithms**

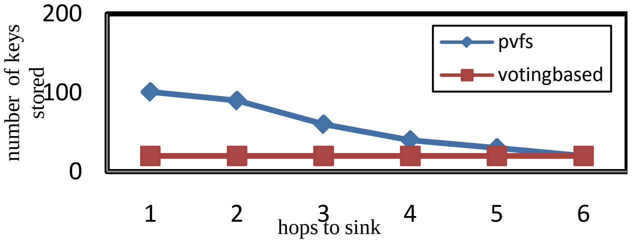


**Figure 3: Key storage overhead**

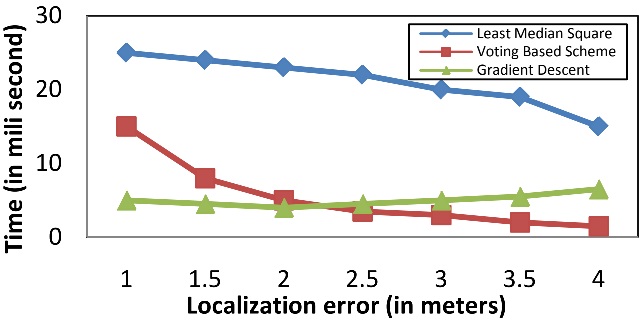


**Figure 4: Comparison of localization error for different localization schemes**

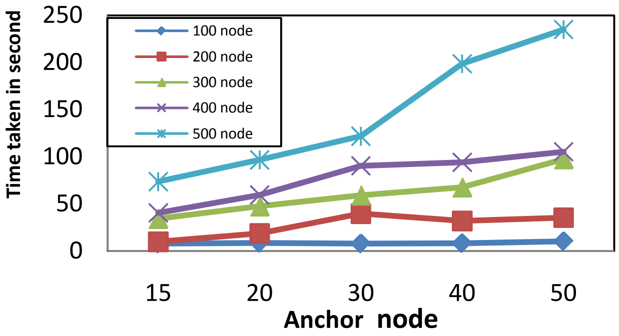


**Figure 5: Time taken for localization**

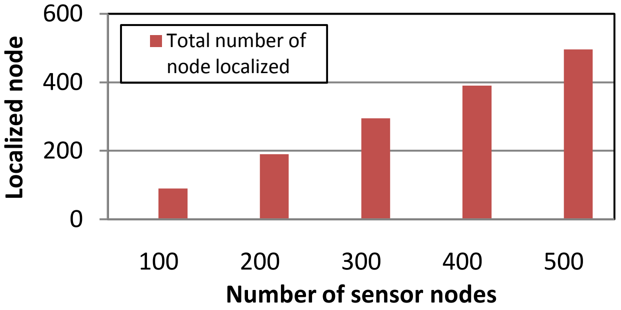


**Figure 6: Total number of nodes localized.**

scheme works efficiently upto 400 nodes. Above 400 nodes all the nodes will be localized but time taken for localization increases. Figure 6 gives the total number of nodes localized which is approximately 97% for varying network sizes.

## 6.0 CONCLUSION

In this paper, we proposed a secure and computationally efficient scheme for localization in wireless sensor networks. Voting based method is used to find localizing area of the node with low estimation error even for complex networks. Later trilateration is applied to find their position with the assistance of a small number of trusted entities. Authentication effectively prevents the attacks since it can filter the false information, which is caused by malicious sensor or anchor nodes that disturb the localization process. As the localization process involves fewer reference points the communication cost is reduced compared to other schemes.

## 7.0 REFERENCES

[1]. D. Liu, P. Ning, A. Liu, C. Wang, and W. K. Du, "Attack Resistant Location Estimation In Wireless Sensor Networks," ACM trans. Inf. Syst. security, vol. 11, no. 4, pp. 1–39, 2008.

[2]. Jianqing Ma, Shiyong Zhangand Yiping Zhong "Seloc: Secure Localization For Wireless Sensor And Actor Network," ACM Transactions on Sensor Networks, IEEE 2006.

[3]. John R And Lowell, "Military Applications Of Localization,Tracking, And Targeting",IEEE Wireless Communications-April 2011.

[4]. Wenbo Yang And Wen Tao Zhu " Voting-On-Grid Clustering For Secure Localizationin Wireless Sensor Networks", IEEE ICC 2010.

[5]. Z. Li, W. Trappe, Y. Zhang, and B. Nath, "Robust Statistical Methods For Securing Wireless Localization In Sensor Networks" in Proc. 4th Int Symp. Inf. Process. Sens. Netw. (IPSN), Los Angeles, CA, 2005, p. 12.

[6]. M. A. Fischler and R. C. Bolles, "Random Sample Consensus: A Paradigm For Model Fitting With Applications To Image Analysis And Automated Cartography" , Commun. ACM, vol. 24, no. 6, pp. 381–395, 1981.

[7]. Loukas Lazos And Radha Poovendran, "Serloc: Secure Range-Independent Localization For Wireless Sensor Networks" , Wise'04, October 1, 2004, Philadelphia, Pennsylvania, USA.

[8]. L. Hu And D. Evans, "Localization For Mobile Sensor Networks", In Proc. 10th ACM ANNU. Int. Conf. Mobile Comput. Netw. (Mobicom), Philadelphia, Pa, 2004, Pp. 45–57.

[9]. R. Szewczyk, E. Osterweil, J. Polastre, M. Hamilton, A. Mainwaring, D. Estrin, "Habitat Moand Nitoring With Sensor Networks", Commun. Vol. 47, No. 6, Pp. 34–40, Jun. 2004.

[10]. A. Savvides, C.-C.Han, Andm. B. Strivastava, "Dynamic FineGrained Localization In Ad-Hoc Networks Of Sensors", In Proc. 7th ACM ANNU. Int. Conf. Mobile Comput. Netw. (Mobicom), Rome, Italy, 2001, Pp.166–179.

[11]. Avinash Srinivasan And Jie Wu, "A Survey On Secure Localization In Wireless Sensor Networks,"Florida Atlantic University, Boca Raton, Fl, USA.

[12]. Amit Gupta, Shashikala Tapaswi, " Recurrent Grid Based Voting Approach For Location Estimation In Wireless Sensor Networks," IEEE Doi 10.1109/Uic-Atc.2009.43.

[13]. Ravi Garg , Avinashl.Varna And Minwu "An Efficient Gradient Descent Approach To Secure Localization In Resource Constrained Wireless Sensor Networks," IEEE Transactions On Information Forensics And Security, Vol. 7, No. 2, April 2012, 717.

[14]. Feng Li And Jie Wu "A Probabilistic Voting-Based Filtering Scheme In Wireless Sensor Networks", IWCMC 06, July 3–6, 2006, Vancouver, British Columbia, Canada.

[15]. J.T.Chiang, J. J. Haas, Andy.-C. Hu, “Secure And Precise Location Verificationcusing Distance Bounding And Simultaneous Multilateration,” In Proc. 2nd ACM Conf. Wireless Netw. Security, Zurich, Switzerland, 2009, Pp. 181–192.

[16]. Chin-Mu Yu, Yao-Trg Tsou, Chun-Shien Lu and Sy-Yen Kno, Localized Algorithms for Detection of Node Replication Attacks in Mobile Sensor Networks, I086 Transaction of Information Forensics and Security, VOL 8 No.5, may 2013.

[17]. Jinfang Jieng, Guangjie Han, Chddan Zhu, Yuhui Dong, Na Zhang, Secure Localization in Wireless Sensor Networks: A survey, Journal of communication, VOL 6, NO.6, September 2011.

[18]. Ning Yu, Liru Zhong and Yongji Ren. BRS-Based Robust Secure Localization Algorithm for Wireless Sensor Networks, Volume 2013, doi:10.1155/2013/107024

[19]. Jie Chen, An Improved Downhill Simplex-Genetic Multiple-Source Localization in Wireless Sensor Networks. Journal of Computational Information Systems 7:11(2011) 4007-4014.

[20]. Sohail Jabbar, Rabia Iram, Abid Ali Minhas, Imran Shafi and Shahzad Khalid, Intelligent Optimization of Wireless Sensor Networks through Bio-inspired Computing; survey and Future Directions. International Journal of Distributed Sensor Networks, Volume 2013, Article ID 421084, 13page.

[21]. Ashwani Kush and C Hwang “Hash Security for Ad hoc Routing” in BIJIT - BVICAM’s International Journal of Information Technology January – June, 2011; Vol. 3 No. 1; ISSN 0973 – 5658.

[22]. B B Jayasingh and B Swathi “A Novel Metric for Detection of Jellyfish Reorder Attack on Ad Hoc Network” in BIJIT - BVICAM’s International Journal of Information Technology Jan – June, 2010; Vol. 2 No. 1; ISSN 0973 – 5658.

[23]. B V Ramanamurthy, K Srinivas Babu and Mohammed Sharfuddin “Dynamic Data Updates for Mobile Devices by Using 802.11 Wireless Communication” in BIJIT - BVICAM’s International Journal of Information Technology Jan – Jun, 2011; Vol. 3 No. 1; ISSN 0973 – 5658.

[24]. Pranav M Pawar, Smita Shukla, Pranav Kulkarni and Adishri Pujari “Simulation and Proportional Evaluation of AODV and DSR in Different Environment of WSN” in BIJIT - BVICAM’s International Journal of Information Technology Jan – Jun, 2011; Vol. 3 No. 1; ISSN 0973 – 5658.

[25]. Sulata Mitra and Arkadeep Goswami “Load Balancing in Integrated MANET, WLAN and Cellular Network” in BIJIT - BVICAM’s International Journal of Information Technology Jan – Jun, 2011; Vol. 3 No. 1; ISSN 0973 – 5658**.**

BIJIT - BVICAM's International Journal of Information Technology
Bharati Vidyapeeth's Institute of Computer Applications and Management (BVICAM), New Delhi (INDIA)

# Comparative Analysis of Data Aggregation Algorithms Under Various Architectural Models in Wireless Sensor Networks

**Anitha C L**[1] and **R. Sumathi**[2]

*Submitted in March, 2014; Accepted in September, 2014*

***Abstract - Wireless sensor network has emerged as a promising technique that revolutionary the way of sensing information. Dense deployed sensor nodes in a specific region are likely to transfer redundant data to the base station. This increases the communication overhead and affects network lifetime. Since energy conservation is the key issue in wireless sensor network, data aggregation should be incorporated in order to save energy. The main aim of data aggregation technique is to collect and aggregate data in an energy efficient manner so that network lifetime is enhanced. In this paper, authors present state of the research by summarizing the work on data aggregation algorithms that has already been published and by highlighting the performance characteristics that are being addressed. The performance comparison of clustered based data aggregation, chain based data aggregation, tree based data aggregation and grid based data aggregation algorithms have been analyzed using NS-2 for various parameters.***

***Index Terms: Wireless Sensor Networks, Data aggregation.***

## 1.0 INTRODUCTION

Wireless Sensor Networks (WSNs) have a large number of sensor nodes with an ability to communicate among themselves and also to an external sink or base-station [1, 2]. The sensors could be scattered randomly in harsh environments such as a battlefield or deterministically placed at specified locations as shown in figure 1.Wireless sensors are equipped with limited range of sensing, computational, storage and communication resources. Extensive utilization of communication resources can potentially reduce the battery life of a wireless sensor. Hence energy conservation must be considered as a most basic constraint while designing a WSN as it governs the network lifetime.  A lifetime of WSN depends on the lifetime of sensor nodes. After the deployment of sensor devices, it is impossible to charge or replace battery present in the network.WSN's can be used for a wide variety of monitoring and research application, inventory maintenance, health care, military, object recognition and tracking  and environmental phenomena. During monitoring sensor nodes collect sensory information which is highly redundant and correlated. Since sensor nodes are energy constrained, it is inefficient for all the sensors transmit the data directly to the base station. To conserve energy this redundant information is aggregated and it is transmitted to the base station as illustrated in figure 2. Data aggregation is defined as the process of aggregating the data from multiple sensors to eliminate the redundant transmission and provide consolidated information to the base station [4], [8]. Eventually, the lifetime of the sensor nodes can be increased.

In this paper, the authors made an attempt to present various architectural models that exist under hierarchical networks which are used for data aggregation in WSN and also the performance analysis of various algorithms of each architecture is considered. Some of the network parameter has taken to compare the performance of each algorithm under clustered based data aggregation, chain based data aggregation, tree based data aggregation and grid based data aggregation algorithms.

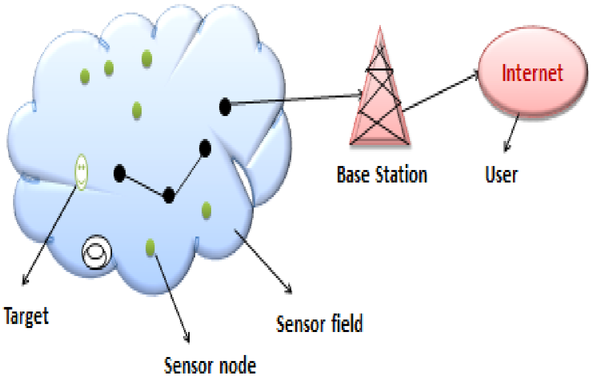


**Figure 1: A typical Wireless Sensor Network**

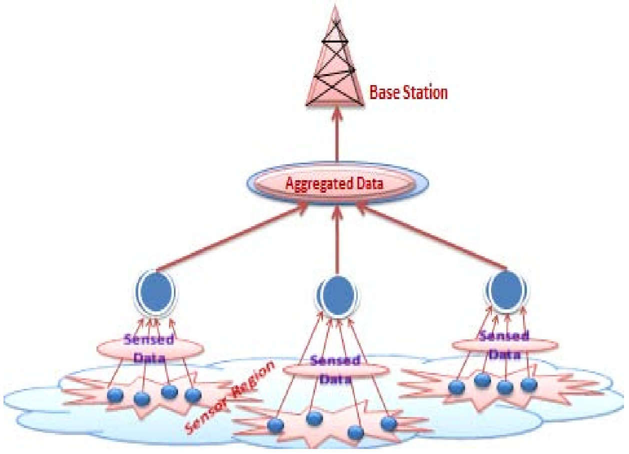


**Figure 2: Data Aggregation**

[1] *Research Scholar, Department of Computer Science and Engineering, Kalpataru Institute of Technology, Tiptur Tumkur, Karnataka, India.*
[2] *Department of Computer Science and Engineering, Siddaganga Institute of Technology, Tumkur 572103, Karnataka, India.*
*E-mail:* [1]clanitha@gmail.com *and* [2]rsumathi@sit.ac.in

The remainder of the paper is organized as follows: Section 2 briefly reviews a survey on previous approaches focusing on their disadvantages. Section 3 presents different architectural models of data aggregation. Section 4 describes the performance analysis of data aggregation techniques. The simulation results of various data aggregation algorithms are compared and analyze in section 5. Finally, Section 6 concludes the paper.

## 2.0 RELATED WORK

During the past few years, many different protocols for data WSN aggregation have been proposed. Literature [1] proposes a detailed survey on various aspects of WSNs and different data aggregation techniques. All of them focus on optimizing performance measures such as network lifetime, data latency, data accuracy and energy consumption.

In a WSN application for tracking multiple mobile targets [2], large amounts of sensing data can be generated by a number of sensors. Generated data must be controlled with an efficient data aggregation technique so that number of data transmissions can be reduced by using one such clustering based data aggregation algorithm which shows effectiveness in restricted type of sensing scenarios, while posing great problems when trying to adapt to various environmental changes.

Power Efficient Gathering Sensor Information System (PEGASIS) [4, 5] is a chain based power efficient routing protocol. This protocol is applicable to homogeneous sensors. PEGASIS assumes that all the sensor nodes have the same level of energy and they are likely to die at the same time. Since all nodes are immobile and have global knowledge of the network, the chain can be constructed easily by using a greedy algorithm. In this approach, each sensor node will have the information about hop neighbors. Sensed information will be passed across to the next hop neighbor and hop neighbour transmit the packet to the next hop neighbour until it reaches the base station.

Sensor Protocol for Information via Negotiation (SPIN) [3, 5, and 6] is an adaptive protocol that uses data activity and resource adjustable algorithms. SPIN follows Proactive type flat architectural approach. In SPIN algorithm all the nodes are close to the base station. The nodes which are closer will sense and gather identical information. In SPIN algorithm all sensor nodes act as a base station. SPIN solves these shortcomings of conventional approaches using data negotiation and resource-adaptive algorithms. The user can query to any node to gather sensed information. Data transmitted within the sensor nodes are called as metadata. Before transmission, meta-data will be passed across all the sensor nodes. After sensor node receives a meta-data it advertises the neighboring node whether interested in receiving the meta-data.

A Tiny Aggregation Approach (TAG) [7] is a data centric protocol. It is a tree based data aggregation approach and designed especially for monitoring applications. This means that all nodes should produce relevant information periodically. Therefore, it is possible to classify TAG as a periodic per hop adjusted aggregation approach.

A Tree based Data Aggregation Mechanism in WSN (TDAM) [8] in which this mechanism describes hop count and energy as new parameters in order to construct aggregation tree. The main aim of this design is to reduce the power consumption of the nodes in the network. Also reduces the number of nodes to relay, thereby reducing the amount of transmitted packets and no too complex operation.

Adaptive clustering based data aggregation technique [10] is a method that implements both static and dynamic clustering methods. This method assumes that the static clustering based data aggregation technique has advantages when there are multiple targets, and when the velocity of those targets is high. On the other hand, the dynamic clustering based technique has great advantages when there are only a few targets with low velocity. Therefore this method will select the static cluster based aggregation when data traffic is high, and adaptively switch to dynamic cluster based aggregation when the network realizes that the data traffic is low. The threshold for deciding when to switch between the data aggregation methods will be configured and decided at the sink node. The initial clustering method of the network will also be configured at the sink.

Threshold sensitive Energy Efficient sensor Network (TEEN) [3, 9] is a Cluster based Hierarchical approach h which follows LEACH protocol. This is an important routing approach used in the Time Critical application. TEEN is a Cluster-based reactive protocol. TEEN uses the LEACH protocol to design a network topology. It follows the same approach of LEACH to identify the Cluster Head and sensor nodes.

In Directed Spanning Tree (DST) [11] routing protocol, a node considers one of its neighbor nodes, which is nearest to the sink as a parent node in the tree. It chronically transmits packets to the parent node. As the case may be, every node (except for the sink) can choose a neighbor node which is nearest to the sink as its parent node. So a tree shape communication path will be constructed, which sets the sink node as its root. By the Directed Spanning Tree, any node can find a shorter and a time-saving path to transmit data packets to the sink.

Low energy adaptive clustering hierarchy (LEACH) [3] is randomized; self-organizing cluster based routing protocol used in wireless sensor network. In this protocol the base station will be a fixed and located far away from the sensor region. In a cluster of sensors a node acts as cluster head or a group leader, which performs aggregation and routing of packets to the sink. In this protocol sensing and gathering of information are equally done with all sensor nodes and aggregated at the cluster head node. Rumor Routing (RR) [3, 12, and 13] is an adaptive algorithm which directs diffusion method. It follows Hybrid type flat protocol. The RR method combines query flooding and event flooding. Rumor Routing is applicable on a network which is composed of densely distributed nodes. RR uses query flooding and event flooding protocols in a randomized manner to fetch the interested information.

The clustered Aggregation algorithm is to compute approximate answers to queries by using spatial and temporal properties of the data [15]. CAG forms clusters of nodes sensing similar values. It ignores redundant data using the spatial and temporal correlations provide significant energy savings. In [16], EECDA combines energy efficient cluster based routing and data aggregation for improving the performance in terms of lifetime and stability [4]. It is for the heterogeneous WSN. EECDA balances the energy consumption and prolongs the network lifetime by a factor of 51%, when compared with LEACH. Chain Oriented Sensor Network for Efficient Data Collection (COSEN) [17]; it is a two-tier hierarchical chain-based routing scheme. COSEN compared to PEGASIS, it can alleviate the transmission delay and energy consumption. In [18] simulation results show that EECHDA has significant gain in network lifetime over direct transmission under the assumption that nodes are randomly and densely deployed.

## 3.0 ARCHITECTURAL MODELS IN HIERARCHICAL NETWORKS

Hierarchical networks are the special type of networks that comes under WSN. A characteristic of the hierarchical wireless sensor network is creation of cluster head where cluster heads perform several special functions such as maintaining the clusters and aggregation. Data aggregation is performed by cluster heads or a leader node. Overhead is involved in a cluster or chain formation throughout the network. As such the concept of hierarchical network is also utilized to perform energy-efficient task in WSNs. In a hierarchical network, creation of clusters and assigning of special tasks to cluster-heads can greatly contribute to overall system scalability, lifetime and energy efficiency. Several architectural models that exist in hierarchical networks and some of the data gathering techniques have been proposed under each model. The four hierarchical networks under study are clustered based data aggregation, chain based data aggregation, tree based data aggregation and grid based data aggregation.

### 3.1 Chain based Architecture

In which each sensor sends data to the closest neighbor. All sensors are structured into a linear chain for data aggregation. The nodes can form a chain by employing a greedy algorithm or the sink can determine the chain in a centralized manner. Figure 3 explains chain based architecture. Greedy chain formation assumes that all nodes have global knowledge of the network. The farthest node from the sink initiates the chain formation and at each step, the closest neighbor of a node is selected as its successor in the chain. In each data gathering round, a node receives data from one of its neighbors, fuses the data with its own and transmits the fused data to its other neighbor along the chain.

### 3.2 Tree Based Architecture

In tree based architecture, data aggregation is performed by constructing aggregation tree which could be a minimum spanning tree where sensor nodes act as the leaf nodes and the sink node or master node act as root node [7, 15]. Figure 4 shows the principle of tree based architecture.

The flow of the data takes place from the leaf node to the parent node. Tree based architecture is suitable for designing optimal aggregation techniques. The aggregation is done at the base station also acts as the parent node.

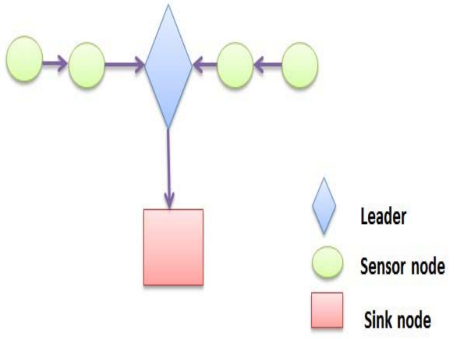


**Figure 3: Chain based architecture**

### 3.3 *Cluster Based Architecture*

Cluster based data aggregation approach is widely used in WSN. In cluster based approach the whole network is divided into several clusters. The sensor nodes themselves form a cluster and elect a node as cluster head. The data sensed by the sensor nodes are passed to the cluster head and in the cluster head data aggregation is performed. Cluster head performs data aggregation and forward the data to the sink. Fig. 5 shows the Cluster based approach, data aggregation is performed by cluster heads. Communication cost is reduced since only aggregated results reach the base station.

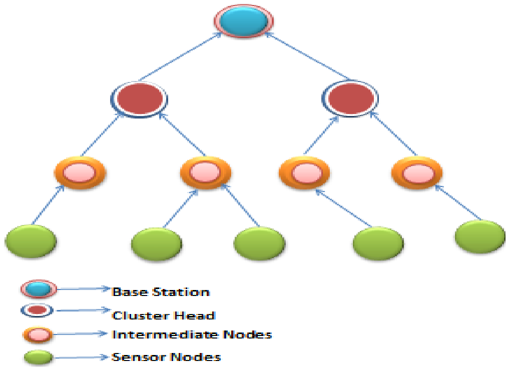


**Figure 4: Tree based architecture**

In cluster based networks, user can put some more powerful nodes, in terms of energy, in the network, which can act as a cluster-head and other simple node work as a cluster-member only. There is several clusters based network organization and data aggregation protocols have been proposed.

### 3.4 Grid Based Architecture

In grid based architecture set of sensors is assigned as data aggregators in fixed regions of the sensor network as shown in fig. 6. The sensors in a grid send the data packet directly to the aggregator of that grid. Hence, the sensors within a grid do not communicate with each other. Each sensor within a grid communicates with its neighboring node. Any node within a grid can assume the role of the aggregator node in terms of rounds until the last node dies.

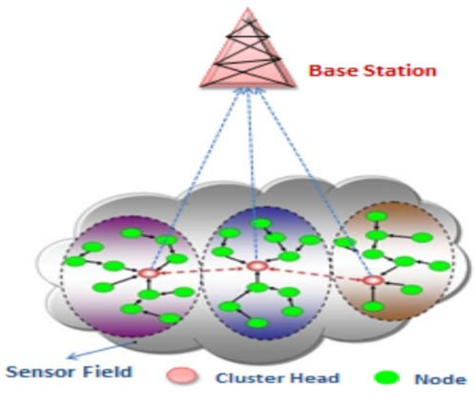


**Figure 5: Cluster Based architecture**

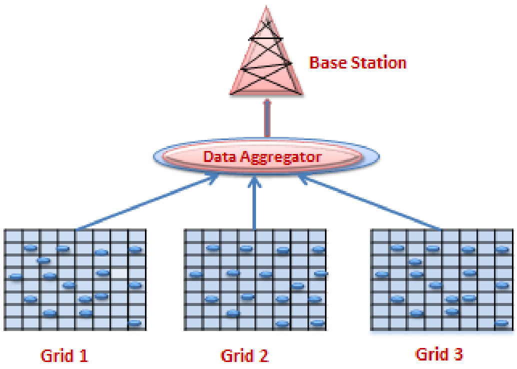


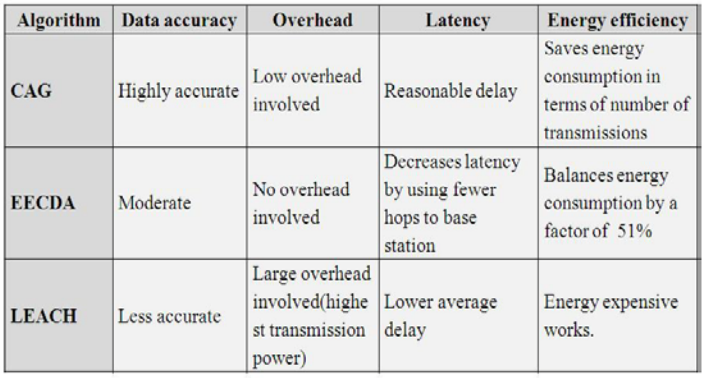
**Figure 6**: **Grid Based architecture**

## 4.0 COMPARATIVE ANALYSIS

Algorithms that are considered in each model are compared against the following performance metrics : (i) data accuracy (ii) overhead (iii) latency (iv) energy efficiency. According to the survey analysis the observed details are reported on Clustered based data aggregation algorithm, Chain based data aggregation algorithm, Tree based data aggregation algorithm and Grid based data aggregation algorithms in distinct scenarios and are depicted in table1, table 2, table 3 and table 4.

### 4.1 Clustered based Data Aggregation

Table 1 shows the performance characteristics of cluster based aggregation algorithms and algorithms under study are Clustered Aggregation Technique (CAG), Energy Efficient Clustering and Data Aggregation Technique (EECDA) and Low-Energy Adaptive Clustering Hierarchy (LEACH). As shown in table1 the first observation we made is that the CAG is much more efficient than EECDA and LEACH in terms of the total number of messages (control and data forwarding) incurred by the algorithms. As reported, CAG is highly accurate than EECDA and LEACH. CAG provides energy efficient and approximate aggregation results with small and often negligible and bounded error. The advantage of CAG is the high precision of the approximate results. The main difference between LEACH and CAG is that LEACH does not provide a mechanism to compute aggregate using cluster head values, while CAG does. LEACH has worse energy consumption, distribution.



| Algorithm | Data accuracy | Overhead | Latency | Energy efficiency |
|---|---|---|---|---|
| CAG | Highly accurate | Low overhead involved | Reasonable delay | Saves energy consumption in terms of number of transmissions |
| EECDA | Moderate | No overhead involved | Decreases latency by using fewer hops to base station | Balances energy consumption by a factor of 51% |
| LEACH | Less accurate | Large overhead involved(highest transmission power) | Lower average delay | Energy expensive works. |

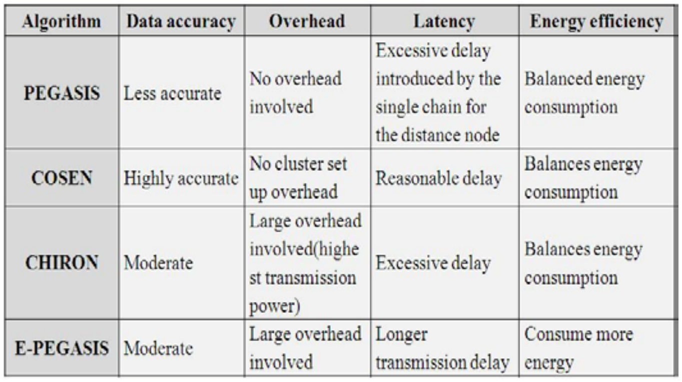
**Table 1: Comparison of Cluster based data aggregation algorithms**

### 4.2 Chain based Data Aggregation

Table 2 studies the Chain based data aggregation algorithms like Power Efficient Gathering in Sensor Information System (PEGASIS), Chain Oriented Sensor Network for Efficient Data Collection (COSEN), Enhanced PEGASIS (E-PEGASIS), Chain-Based Hierarchical Routing Protocol (CHIRON).COSEN is efficient in the ways that it ensures maximal utilization of network energy, it makes the lifetime of the network longer, as well as it takes much lower time to complete a round. Simulation results show that COSEN demonstrate around 20% better performance than that of PEGASIS in respect of the number of rounds before the first sensor dies. It also saves about 260% time on average in comparison to PEGASIS. Performance analysis and simulation show that COSEN noticeably give a good compromise between energy efficiency and latency. COSEN require much lower time and energy as compared to other algorithms of WSN for data collection. However, this achievement is faded by the excessive delay introduced by the single chain for the distant node in CHIRON AND E-PEGASIS. The ultimate Improvement of COSEN from PEGASIS is that, the delay is much lower in COSEN.

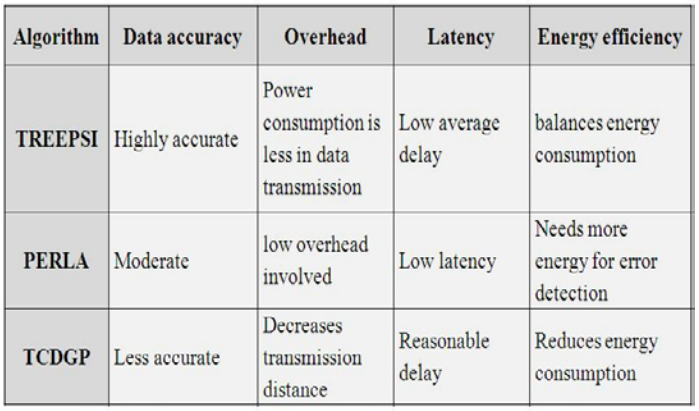
### 4.3 Tree Based Data Aggregation

Table 3 show tree based algorithms under study are Tree-based Efficient Protocol for Sensor Information (TREEPSI), Power Efficient Routing with Limited Latency (PERLA), Tree-Clustered Data Gathering Protocol (TCDGP). A tree-based data gathering protocol TREEPSI improves upon the PERLA and TCDGP. This protocol further reduces power consumption. We can shorten the transmission distance between nodes and prevent the root nodes from dying quickly.



| Algorithm | Data accuracy | Overhead | Latency | Energy efficiency |
|---|---|---|---|---|
| PEGASIS | Less accurate | No overhead involved | Excessive delay introduced by the single chain for the distance node | Balanced energy consumption |
| COSEN | Highly accurate | No cluster set up overhead | Reasonable delay | Balances energy consumption |
| CHIRON | Moderate | Large overhead involved(highest transmission power) | Excessive delay | Balances energy consumption |
| E-PEGASIS | Moderate | Large overhead involved | Longer transmission delay | Consume more energy |

Table 2: Comparison of Chain based data aggregation algorithms



| Algorithm | Data accuracy | Overhead | Latency | Energy efficiency |
|---|---|---|---|---|
| TREEPSI | Highly accurate | Power consumption is less in data transmission | Low average delay | balances energy consumption |
| PERLA | Moderate | low overhead involved | Low latency | Needs more energy for error detection |
| TCDGP | Less accurate | Decreases transmission distance | Reasonable delay | Reduces energy consumption |

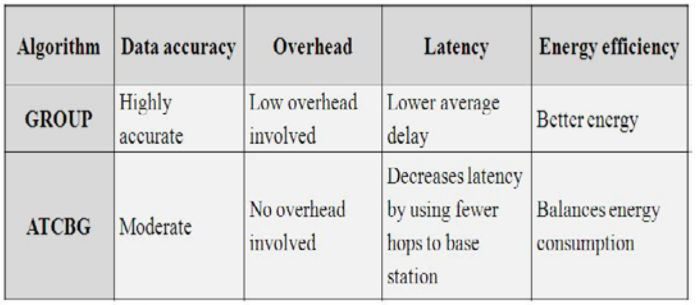
Table 3: Comparison of Tree based data aggregation algorithms

### 4.4 Grid based Data Aggregation

As described in table 4 algorithms under study are Grid-clustering Routing Protocol for Wireless Sensor Networks (GROUP), Aggregation Tree Construction Based on Grid (ATCBG). GROUP is an energy-efficient and scalable routing protocol for large-scale wireless sensor networks. In GROUP, cluster heads can perform data aggregation expediently in order to reduce the number of data packets and save energy. GROUP has lower maximum energy consumption than ATCBG. Our simulations have confirmed that GROUP is an effective, scalable and energy-efficient routing protocol for large-scale wireless sensor networks. It can also be observed that the average energy consumption of GROUP is evidently lower than ATCBG, and the lifetime of the network is much longer than ATCBG before the emergence of node death.

## 5.0 SIMULATION RESULTS

In this section we evaluate the performance of Clustered based data aggregation algorithm, Chain based data aggregation algorithm, Tree based data aggregation algorithm and Grid based data aggregation algorithms through simulations. The Network life time of WSN is determined by the time duration before the first node fails in the network. Therefore, it is very important to manage the sensor nodes in an energy efficient way to extend the lifetime of the sensor network.



| Algorithm | Data accuracy | Overhead | Latency | Energy efficiency |
|---|---|---|---|---|
| GROUP | Highly accurate | Low overhead involved | Lower average delay | Better energy |
| ATCBG | Moderate | No overhead involved | Decreases latency by using fewer hops to base station | Balances energy consumption |

Table 4: Comparison of Grid based data aggregation algorithms

To increase the network lifetime the number of packet transmission between the sensor node and the sink must be decreased. We set up a simulation environment using NS-2. The simulation was performed using this environment in a 100mx100m sensor field and 50 sensors were randomly deployed in this field which is constant with various parameters with respect the architectural models.The graphs in fig.7 depict the comparison of network lifetime with CAG, EECDA and LEACH in relation to the network lifetime and the data transfer rate. The graphs show an increase in network lifetime of the simulated network with CAG. It can be observed that CAG has less energy consumption of nodes in the process of cluster head selection than EECDA and LEACH algorithms. As compared to EECDA and LEACH, CAG gives better performance and extends the lifetime of the network. Based upon the simulation results, CAG can control the residual node energy and effectively extend the network lifetime without performance degradation. The reason is that extra transmissions have been eliminated and total energy consumption has been decreased.As depicted in figure 8 blue lines shows the performance of COSEN which has a better network lifetime, stability and energy efficiency when compared with CHIRON, PEGASIS and E-PEGASIS.It is shown that after several hundreds of rounds the amount of energy consumed is approximately same. But the good point for COSEN is that it spends energy in a totally distributed way such that the network can operate a higher number of rounds before the first sensor dies. COSEN, CHIRON, PEGASIS, E-PEGASIS lifetime pattern is shown in figure 8.Figure 9 shows network lifetime has been uploading for tree based algorithms TREEPSI, PERLA and TCDGP. By observing graphs plotted in figure 9 one can notice that TREEPSI is slightly better than

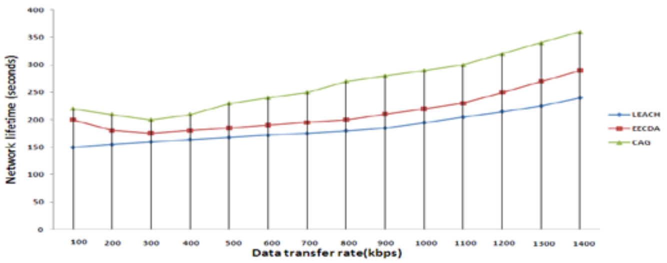


**Figure 7**: **Network Lifetime of Cluster Based Algorithms**

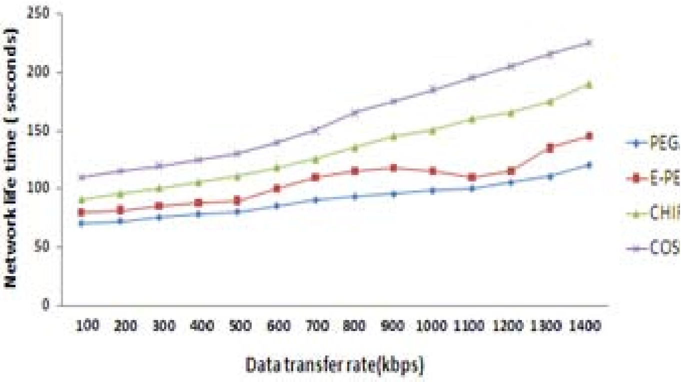


**Figure 8**: **Network Lifetime of Chain Based Algorithms**

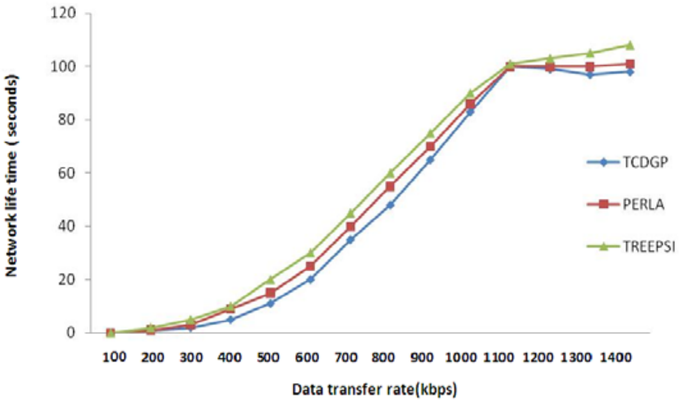


**Figure 9: Network lifetime vs. data transfer rate**

TCDGP and PERLA. This behavior is because the delay in TREEPSI is lesser compared to other two algorithms PERLA and TCDGP. The graphs in figure 9 depict TREEPSI shows good performance even in highly dynamic situations. Because PERLA needs more energy for error detection and recovery procedures in case of root failure to sink. But in TREEPSI, the path has made a detour in the topology.

Table 4 describes Grid Based data aggregation techniques and the algorithms under study are Grid Clustering Routing Protocol (GROUP) and Aggregation Tree Construction Based on Grid (ATCBG). As per the survey analysis, the following details have been given.

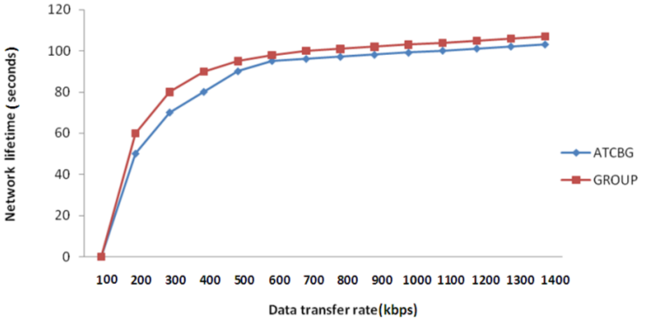


**Figure 10**: **Network Lifetime vs. Data transfer rate**

Table 4 shows the comparison of the GROUP and ATCBG Grid based algorithms. GROUP has a better data transmission rate than ATCBG especially in the scenarios with more nodes. GROUP has lower maximum energy consumption than ATCBG. GROUP has smaller gaps between maximum and average energy consumption. GROUP has a lower average delay with fewer nodes. GROUP is more scalable grid based algorithm and shows significantly better performance than ATCBG. Lifetime pattern of grid based algorithms is shown in figure 10.

## 6.0 CONCLUSION

In this paper, the authors studied various data aggregation algorithms based on various architecture such as Cluster based data aggregation, Chain based data aggregation, Tree based data aggregation and Grid based data aggregation in WSN. Through simulation, the performance of different data aggregation algorithms is evaluated and analyzed . Their advantages and disadvantages are discussed and compared. Results demonstrate data accuracy, overhead, latency, and energy efficiency of these algorithms.

## 7.0 REFERENCES

[1]. Sushrutha Mishra, HirenThakkar, "Features of WSN and Data Aggregation techniques in WSN: A Survey", International Journal of Engineering and Innovative Technology (IJEIT), Volume 1, Issue 4, April 2012.

[2]. P. N. Renjith, E. Baburaj, "An Analysis on Data Aggregation in Wireless Sensor Networks", International conference on Radar, communication and computing (ICRCC) SKP Engineering college, Tiruvanamalai, TamilNadu, India, December, 2012 pp. 62-71

[3]. AHeinzelman, W.; Chandrakasan, A.; Balakrishnan, H. "Energy–efficient communication protocol for wireless micro sensor networks". In Proceedings of the 33rd Annual Hawaii International Conference on SystemSciences (HICSS), Big Island, HI, USA, January 2000; pp. 3005-3014.

[4]. Lindsey, S.; Raghavendra, C.S. "PEGASIS: Power Efficient gathering in sensor information systems". In Proceedings of IEEE Aerospace Conference, Big Sky, MT, USA, March 2002.

[5]. Martorosyan, A.; Boukerche, A.; NelemPazzi, R.W. Taxonomy of cluster-based routing protocols for wireless sensor networks". In International Symposium on Parallel Architectures, Algorithms, and Networks, Sydney, NSW,Australia, May 7–9, 2008; pp. 247-253.

[6]. Jamal, N.; E. Kamal, A.-K.A. "Routing techniques in wireless sensor networks: A survey". IEEE Wireless. Communication. 2004, 11, 6- 28.

[7]. S. Madden et al., "TAG: a Tiny Aggregation Service for Ad- hoc Sensor Networks," OSDI 2002, Boston, MA, Dec. 2002.

[8]. Chih Hsiao Tulsa, Hao Yi Huang, Chih Wei Huang, Ying Hong Wang, "TDAM: The Tree Based Data

Aggregation Mechanism in Wireless Sensor Network", 2012 IEEE International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS 2012) November 4-7, 2012.

[9]. Manjeswar, A.; Agrawal, D.P. TEEN: "A protocol for enhanced efficiency in wireless sensor networks". In Proceedings of 1st International Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile Computing, San Francisco, CA, USA, 2001; p. 189.

[10]. Woo-Sung Jung, Keun-Woo Lim, Young-Baku, Sang-Joon Park, "A Hybrid Approach for Clustering-based Data Aggregation in Wireless Sensor Networks", 2009 Third International Conference on Digital Society.

[11]. PengJi, Chengdong Wu, Yunzhou Zhang and ZixiJia, "Research of Directed Spanning Tree Routing Protocol for Wireless Sensor Networks", Proceedings of the 2007 IEEE International Conference on Mechatronics and Automation August 5 - 8, 2007, Harbin, China

[12]. Braginsky, D.; Estrin, D." Rumor routing algorithm for sensor networks", In Proceedings of the First Workshop on Sensor Networks and Applications (WSNA), Atlanta, GA, USA, October 2002.

[13]. Akkaya, K.; Younis, M." A survey of routing protocols for wireless sensor networks". J. Ad Hoc Netw. 2005, 3, 325-349.

[14]. JyotirmoyKarjee, H.S Jamadagni, "Data Accuracy Estimation for Spatially Correlated Data in Wireless Sensor Networks under Distributed Clustering"

[15]. Olivier Dousse, PetteriMannersalo, Patrick Thiran "Latency of Wireless Sensor Networks with Uncoordinated Power Saving Mechanisms" MobiHoc'04, May 24–26, 2004, Roppongi, Japan.

[16]. HuseyinOzgur Tan and Ibrahim Korpeoglu, "Power Efficient Data Gathering and Aggregation in Wireless Sensor Networks".

[17]. SunHee Yoon and Cyrus Shahabi, March 2007, "The Clustered Aggregation (CAG) Technique Leveraging Spatial and Temporal Correlations in Wireless Sensor Networks", ACM Transactions on Sensor Networks (TOSN), Vol. 3, Issue 1, No.3.

[18]. D. Kumar, T.C. Aseri, R.B. Patel "EECDA: Energy Efficient Clustering and Data Aggregation Protocol for Heterogeneous Wireless Sensor Networks " Int. J. of Computers, Communications & Control, ISSN 1841-9836, E-ISSN 1841-9844 Vol. VI (2011), No. 1 (March), pp. 113-124.

[19]. N. Tabassum, Q. E. K. M. Mamun, and Q. Urano, "COSEN: A Chain Oriented Sensor Network for Efficient Data Collection," Proceedings of the Global Telecommunications conference,Vol. 6, pp. 3525-3530, 2003

[20]. Dilip Kumar, T. C. Aseri and R. B. Patel, ". EECHDA: Energy Efficient Clustering Hierarchy and Data Accumulation For Sensor Networks", BIJIT – 2010; Jan – June, 2010; Vol. 2 No. 1; ISSN 0973 – 5658.

# Comparative Study of Endpoint Detection Algorithms Suitable for Isolated Word Recognition

**A. Akila**[1] and **E. Chandra**[2]

*Submitted in December, 2013; Accepted in September, 2014*

***Abstract - Voice Activity Detections (VAD) are used all over the speech processing applications such as speech recognition, speech enhancement etc. In Isolated word speech recognition, the end point detection reduces the computational process. In this paper, a comparative study of three VAD algorithms and the algorithms were analyzed using performance evaluation criteria. The algorithm suitable for the dataset used in the proposed research work is found using the performance criteria like misdetection, speech quality and compression.***

***Index Terms – Frequency domain, Short Term Energy (STE), Voice Activity Detection (VAD), Zero Crossing Rate (ZCR).***

**NOMENCLATURE**

VAD – Voice Activity Detection
STE – Short Term Energy
ZCR – Zero Crossing Rate
ASR – Automatic Speech Recognition

## 1.0 INTRODUCTION

Automatic Speech Recognition (ASR) is a wide area in signal processing where the recognition of the utterance is done. The accuracy of recognition will improve if the input utterance contains only of speech after removing silence from the speech (i.e.) the accuracy will increase by the accurate end point detection. The speech can be classified as silence, voiced and unvoiced. The process of separating the speech segments of an utterance from the background noise is called the End point detection [1]. The end point detection is used for segmenting the input utterance into its subunits also. There are many end point detection algorithms developed. In Isolated word ASR, the detection of endpoints in a speech is done to separate the speech signal from unwanted background noise [2]. This process is called Voice Activity Detection. In isolated word automatic speech recognition, the detection of endpoints in a speech has been done to separate the speech signal from unwanted background noise. The main use of endpoint detection is in speech coding and speech recognition. It is an important enabling technology for a variety of speech based applications. The proposed research work is a Tamil speech recognition system which performs segmentation of the given speech data into syllables and recognize the syllable. The speech data has to undergo the preprocessing step of endpoint detection to improve the performance of the speech recognition system.

## 2.0 VOICE ACTIVITY DETECTION

VAD is a technique used in speech processing in which the presence or absence of human speech is detected. It is also known as speech activity detection or speech detection. The main uses of VAD are in speech coding and speech recognition [3]. It is usually language independent. VAD algorithm is used as first step in speech recognition system.

### 2.1 Characteristics of VAD

- Reliability – the endpoints computed should be correct taking into consideration the weak fricatives.
- Robustness – Suitable for any type of application
- Computation of end points should be accurate
- Adapting to non stationary background noise should be good.
- Simplicity- easy to compute
- Real Time Processing
- No prior knowledge of noise

The essential characteristics are simplicity and robustness [4].

### 2.2 Features used in VAD Algorithm

- Short term energy
- Zero Crossing Rate
- Autocorrelation function based Features
- Spectrum based Features
- Power in band limited region
- Mel Frequency Cepstral Coefficients
- Delta Line Spectral Frequencies
- Features based on higher order statistics

The use of multiple features leads to more robustness against different environment. Most of the VAD Algorithms use Short Term Energy and Zero Crossing Rate features because of their simplicity.

#### 2.2.1 Short Term Energy

The amplitude of the speech signal varies with time. As in Fig 1, the amplitude of unvoiced segments is generally much lower than that of voiced segments. The amount of energy carried by a wave is related to the amplitude of the wave.

[1] *Department of Computer Science, D.J Academy for Managerial Excellence, Coimbatore, India,*
[2] *Department of Computer Science, Dr. SNS Rajalakshmi College of Arts & Science, Coimbatore-32, India.*
*E-mail:* [1] *akila.ganesh.a@gmail.com and*
[2] *crcspeech@gmail.com*

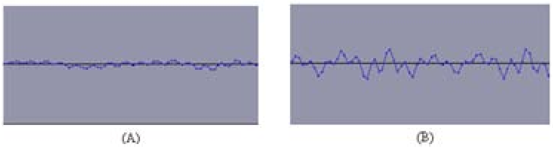


**Figure1: A sample wave signal representing (A) the amplitudes of unvoiced segment (B) the amplitude of voiced segment**

A high energy wave is characterized by high amplitude; a low energy wave is characterized by low amplitude. The energy of a segment indicates the presence of voice data. The energy (E) transported by wave is directly proportional to the square of the amplitude (A) of the wave which is specified in (1).

$$E \propto A^2 \qquad (1)$$

The short term energy can be calculated using the formula[1] as specified in the (2)

$$E = \sum_{n=1}^{N} |S(n)^2| \qquad (2)$$

where s(n) is the amplitude of each frame and n is the current frame of the N frames in the signal.

### 2.2.2 Zero Crossing Rate

Zero crossing is a commonly used term in electronics, mathematics, and signal processing which refers to a point where the sign of a signal changes by crossing of the axis. Fig 2 shows some of the zero crossing point of a sample wave signal.

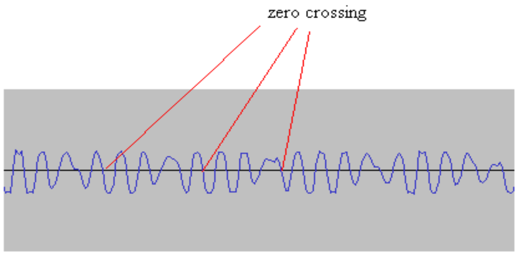


**Figure 2: Zero Crossing points of a sample wave signal**

The rate of sign changes along a signal is called Zero Crossing Rate. It is simple measure of the frequency content of a signal. It is a measure of the number of times in a given signal, the amplitude passes through a value zero. The zero crossing rate [1] is calculated using the (3)

$$\mathrm{ZCR} = \frac{1}{\mathrm{len}-1} \sum_{n=1}^{\mathrm{len}-1} \left| \frac{\mathrm{sgn}(s(n)-\mathrm{sgn}(s(n-1))}{2} \right| \qquad (3)$$

where sgn(s(n))=+1 if s(n) >0 and sgn(s(n))=-1 if s(n)<0,
s(n) is the amplitude of current frame, n is the current frame and len is the number of frames in the signal.

## 3.0 REVIEW OF END POINT DETECTION ALGORITHMS

There are many algorithms to find the end point of the input signal. The three algorithms discussed below are suitable for end point detection of isolated word.

### 3.1 Rabiner's Endpoint Detection Algorithm

The algorithm proposed in [5] uses the two basic features zero crossing rate and short term energy. The uttered speech signal is divided into n frames each of 80ms length. The first 10 frames are considered for calculating the threshold value. The threshold value is used to find the initial point and the endpoint of a speech signal. The algorithm can be used in any environment with a signal to noise ratio of at least 30dB.

### 3.2 Qiang He Algorithm

The VAD Algorithm was developed by Qiang He in the year 2001. It was implemented using MATLAB. The code of the algorithm is available as a free software. The signal is split into overlapping frames of length 80ms. The Short Term energy (STE) and the Zero Crossing Rate (ZCR) are calculated for each frame using equation 2 and 3 respectively. Then ZCR and STE are compared with the threshold values which are chosen between 2 and 10. The comparison result specifies whether the segment or frame is a silent, Voiced or noise signal. If the STE and ZCR are within the threshold values, the frame is a voiced signal or noise signal. If they are below the initial threshold values then the frame contains silent signal. To differentiate between noise and voiced frames, a count of frames which has voice or noise is manipulated. If the total frame length of the counted frames is below 150 ms, then the signal is noise else it is a voiced signal.

### 3.3 VAD with Frequency Domain Approach

This algorithm takes its decisions based on energy comparisons of the signal frame with a reference energy threshold in the frequency domain. The frequency domain (F) of the frame is obtained by (4) where FFT is Fast Fourier Transform function. The signal is divided into frames of length 80ms.

$$F\left(f_j\right) = FFT\left\{f_j\right\} \qquad (4)$$

The Frequency domain is used to find whether the frame is Active or Inactive by comparing with the threshold value. The initial point is the frame where the current frame is Active and the predecessor is Inactive. Similarly the endpoint is frame where current frame is Inactive and the predecessor is Active [6].

## 4.0 EXPERIMENTAL RESULTS

### 4.1 Dataset

The signals that are used for comparison of the three different End point detection Algorithms were 6 simple words of Tamil language which are equivalent to yes and no in English language. The words were recorded for 3 seconds with a frequency of 8 KHz.

The utterances tested were drawn from a single female speaker. For recording the speech, Audacity was used. The sounds were recorded in a normal room where environment may not be quiet. Reliability of endpoint detection Algorithms will be more when environment condition is quiet. But it is not always practical.

### 4.2 Computation

MATLAB environment was used to test the algorithms on the 6 signals. The Initial and the endpoint of voiced segments for each signal are computed. Table 1 list the initial and end points of each signal computed using the three algorithms.

### 4.3 Criteria for assessing the end point detection algorithms

Performance of the algorithms was analyzed based upon the following criteria[7].

Subjective speech quality: The quality of the samples was rated on a scale of 1(poor) to 5(best). The initial signal is assumed to have the best quality of rating 5. The speech samples after end point detection were played and rated.

- Compression Ratio : The ratio of total inactive frames detected to the total number of frames formed
- Misdetection: The number of frames which have speech content, but were classified as inactive and number of frames without speech content but classified as active are counted. The ratio of this count o the total number of frames in the signal is taken as misdetection ratio [8].

**Table 1: Initial and End point (in frame number) of each word without background noise**

| Word uttered | Rabiner's End point Detection | | Qiang He VAD Algorithm | | VAD with Frequency Domain Approach | |
|---|---|---|---|---|---|---|
| | Initial point | End point | Initial point | End point | Initial point | End point |
| AAM | 105 | 169 | 106 | 169 | 107 | 186 |
| AAMAA | 106 | 163 | 105 | 163 | 108 | 179 |
| AAMAAM | 103 | 178 | 102 | 178 | 105 | 194 |
| IILLAI | 103 | 201 | 103 | 201 | 105 | 217 |
| IILLA | 102 | 155 | 102 | 154 | 105 | 171 |
| IILLLA | 199 | 261 | 197 | 261 | 200 | 278 |

The effective algorithm should have high compression and with low misdetection and should maintain speech quality [7]. The misdetection was calculated with manual end point detection using the audacity tool.

### 4.4 Observations

The performance of the three algorithms was analyzed and the result is graphically represented in Fig 3. We observed the following from the result.

- The algorithm that is using Frequency domain has more misdetection when compared to Rabiner and Qiang He Algorithms.
- Compression was slightly better in Qiang He when compared with Rabiner
- Speech Quality was very less in frequency domain when compared with the other two algorithms.

### 5.0 CONCLUSION

The study of three end point detection algorithm was presented. From the experimental result, it is observed that Qiang He Algorithm has good compression ratio and speech quality with less misdetection which is shown in Fig 3. So Qiang He algorithm suits better for end point detection of the given dataset. After performing the end point detection the speech signal can be used as an input to the speech recognition system. The performance of the speech recognition system can be enhanced by using proper end point detection algorithm. Qiang He algorithm was used in our research work of syllable based Tamil speech recognition system at the preprocessing phase.

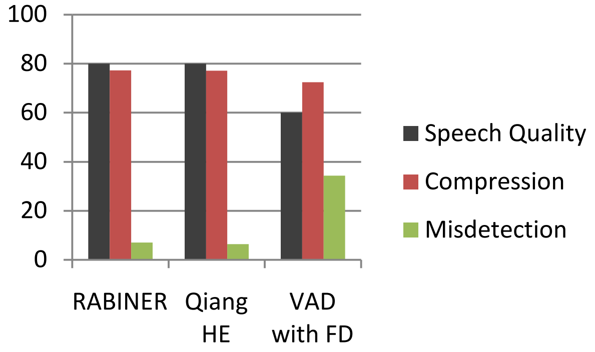


**Figure 3: Graphical representation of the performance criteria of the three algorithms**

### 6.0 REFERENCES

[1]. Miael Nilsson and Marcus EJnarsson, "Speech Recognition using Hidden Markov Model", Department of Telecommunications and speech Processing, Biekinge Institute of Technology, 2002.

[2]. Lawrence Rabiner and Biing-Hwang Juang, "Fundamentals of speech Recognition", Prentice Hall, Englewood Cliffs, N.J. 1993

[3]. Jonathan Kola, Carol Espy-Wilson and Tarun Pruthi, "Voice Activity Detection", MERIT BIEN 2011, pp 1-6

[4]. M.H.Moattar and M.M.Homayounpour, " A Simple but Efficient Real-Time Voice Activity Detection Algorithm", 17th European Signal Processing Conference, August 2009, pp 2549- 2553.

[5]. L.R.Rabiner and M.R.Sambur, "An Algorithm for Determining the endpoints of isolated utterances", The Bell System Technical Journal, Vol. 54, No. 2, February 1975, pp 297-315

[6]. Kirill Sakhnov, "Approach for Energy-Based Voice Detector with Adaptive Scaling Factor", IAENG International Journal of Computer Science, Vol. 36, No. 4, November 2009

[7]. T.Ravichandran and K.Durai Samy, "Performance Evaluation and Comparison of Voice Activity Detection Algorithms", Medwell Journals, International Journal of Soft Computing, 2007, pp: 257-261

[8]. R. Venkatesha Prasad, Abhijeet Sangwan, H.S. Jamadagni, Chiranth M.C, Rahul Sah, "Comparison of Voice Activity Detection for VoIP", Seventh International Symposium on Computers and Communications, IEEE, Tacrmina, 2002

# Fuzzy Logic Based Intruder Detection System in Mobile Adhoc Network

**Shadab Siddiqui[1], P. M. Khan[2]** and **Muhammad Usman Khan[3]**

*Submitted in March, 2014; Accepted in September, 2014*

***Abstract - The paper entitled "Fuzzy Logic based Intruder Detection System in Mobile Adhoc Network" is an approach to detect malicious nodes by applying fuzzy logic in Mobile ad-hoc networks. Security is a major concern in various scenarios of adhoc sensor network. Detection of malicious nodes forms an essential part of an approach to security. The proposed work uses fuzzy logic to identify the attack and malicious behavior of nodes. The proposed work will identify the attack over the network as well as provide the solution to reduce the execution time over the network. The objective of the work is to provide security in Mobile Adhoc Network. The proposed work uses AODV algorithm. This algorithm implies some fuzzy rules which is implemented on the nodes in the network. The if-then rules of fuzzy will identify the malicious node in the network. The proposed work will do comparison between the performance parameters obtained from AODV with priority based Intruder detection system with AODV implementing fuzzy logic to identify malicious nodes. The results will show great improvement of AODV with fuzzy logic over the previous algorithm. The proposed scheme is implemented using Matlab & its results show its effectiveness.***

***Index Terms – Fuzzy logic, AODV, Mobile Adhoc Network, fuzzy rules, attacks, RREQ- Route request, RREP- Route reply, RERR- Route error.***

## 1.0 INTRODUCTION

As the technology is increasing day by day the popularity of wireless technology is showing a tremendous rise & therefore opening various fields of applications in the area of networking. One of the most important fields in this is MANET in which the nodes do not depend on any preexisting infrastructure. MANET consists of collection of nodes that are connected by wireless links & therefore the interconnection between nodes can change on arbitrary basis. Nodes that are within the communication range of other nodes can communicate directly without the need of wireless links whereas nodes that are far away uses intermediate nodes as relays. The book Adhoc Sensor network [1] defines the network consist of number of nodes and mobile host MH connected by wireless links. Therefore MANET can operate as a standalone implementation with an infrastructure less network. Security in Mobile Adhoc Network is very difficult to achieve due to its dynamic & infrastructure less topology & due to limitations of wireless data transmission. The existing solution applied in wired network can obtain security to a certain level but not always suitable in wireless network. Therefore wireless network has its own vulnerability that cannot be handled by wired network. Due to the different characteristics of wireless & wired network the task of providing seamless environment for it is very difficult. In Mobile Adhoc Network nodes also have limited energy storage. Mostly, they are battery equipped, with very limited recharging or with no replacement possible. Another limited resource in Mobile Adhoc Network is bandwidth. All of the above features of MANETs do pose a serious challenge which is often easier to achieve or predict in wired or infrastructure based networks. Thus, guaranteeing data safety and reliability is a serious issue..

Therefore, the decentralized nature, scalable setup and the dynamic changing topology makes adhoc networks ideal for a variety of applications ranging from military, industrial and natural to data collection machinery analysis, bio-sensing as investigated in [2], [3]. But these same features also drive the key challenges in deploying and using them such as device compatibility, connectivity issues due to varying traffic, security and survivability of nodes in the network

## 2.0 FUZZY LOGIC

Boole [4] introduced the beautiful notion of binary sets, which is the foundation of modern digital computer but boolean logic is unable to model the human cognition and thinking process. Because of its rigid boundaries, the two valued logic is not so efficient in mapping real world situations. In order to handle real world problems Zadeh [5] introduced the concept of 'mathematics of fuzzy or cloudy quantities' followed by his seminal paper 'Fuzzy sets' [6].

Fuzzy logic is a superset of Boolean logic. Fuzzy logic uses fuzzy rules which are one of the important applications of fuzzy theory. Fuzzy logic is described as a mathematical system that uses analog input value between 0 and 1 in contrast to to digital logic. Steps for fuzzy logic are:-

**1) Fuzzification*:*** The aim of fuzzification is to define input variable & input membership function for each input variable.

**2) Knowledge base**: It classifies input according to membership values such as low, medium, high. The knowledge base consists of rules in the form of if-then rules.

**3) Defuzzification (mapping)**: In this two graphs are used.

- Template Graph- It contains all output membership function which are maximized when they have high fuzzy rules.
- User Action Graph- It includes audit log & user profiles.

*[1, 2, 3] Department of Computer Science and Engineering, Integral University, Lucknow*
*EMail: [1] shadabsiddiqui222222@gmail.com,*
*[2] pmkhan@hotmail.com and [3] usmanintegral@gmail.com*

INPUT
Processing Stage
OUTPUT

**Figure1: Fuzzy Controller**

Fuzzy logic deals with reasoning which is approximate instead of fixed. The value in truth table of fuzzy logic ranges between 0-1. It is a problem solving methodology from simple microcontroller to large control systems. Fuzzy logic gives a simple way to arrive at definite conclusion based upon noisy, ambiguous or missing input information.

### 2.1 Fuzzy logic toolbox

Fuzzy logic toolbox can create and edit fuzzy inference systems. These inference systems can be created using command line functions or by using graphical tools. By using simulink we can test our fuzzy system in simulation environment. The toolbox can run C programs without using simulink. This is possible because of fuzzy Inference engine which reads the fuzzy systems.

### 2.2 Fuzzy sets

Fuzzy sets are the sets without any fixed defined boundary. It contains elements with degrees of membership functions. Fuzzy sets is a pair (v, m), where v is a set & m : v→[0,1]. Fuzzy set theory assesses the membership function of elements in a set which is described by the help of membership function in the interval [0, 1].

### 2.3 Membership Function

It is the curve or square graph which defines the mapping of each input point to membership value between 0 and 1.

Ex: Consider a fuzzy sets is the set of tall people. We say from 3 ft to 9 ft word 'tall' will correspond to curve which defines the degree to which the person is tall. If the tall people in the set are within the boundary of classical set then we can say that all people taller than 6 ft are considered tall.

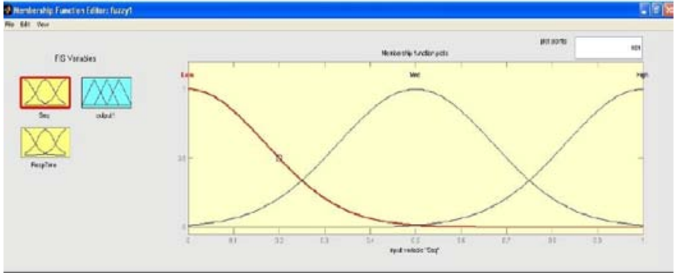


**Figure 2: Fuzzy Input Membership function**

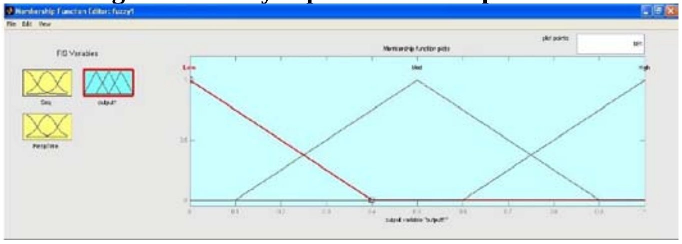

Membership Function Editor: fuzzy1
File Edit View
FIS Variables
Seq
output1
RespTime
Membership function plots
plot points
Low
Med
high
output variable "output1"

**Figure 3: Fuzzy Output Membership Function**

### 2.4 If-then rules

The if-then rules statements can formulate the conditional statements that consist of fuzzy logic. A single fuzzy rule comprises of:

If x is A then y is B, where A& B are values defined by fuzzy sets on the range x & y respectively. The if part of the rule states x is A and is called as antecedent, and then part of the rule is y is B and is called as consequent.

## 3.0 AODV ALGORITHM

AODV is used basically to address routing problems in Mobile Adhoc Network. & establish communication between nodes with minimum control overhead. AODV is a reactive protocol and it does not need the discovery & maintenance of routes which are not in communication instead it discovers the routes quickly to new destinations. AODV is loop free algorithm & operates in distributed manner. This freedom of loop is acquired by using sequence number. Every node has a sequence number which increases monotonically every time there is a change in topology of the network. This sequence number also ensures that recent route is selected when a route discovery process initiates. It basically has three phases:-

### 3.1 Route Discovery

If the node wants to communicate with destination node then it checks if the route to destination is free &valid in the routing table. If the route is available & valid then data is sent and if it does not have the valid route to destination then the source node sends RREQ packet & sets a timer to wait for RREP. Every node on receiving RREQ packet checks whether it has verified the IP address of source & broadcast ID of RREQ. After RREQ the next step is to set reverse routes in routing table. The reverse route contains the IP address of source, the sequence number & the hops required to reach source node.

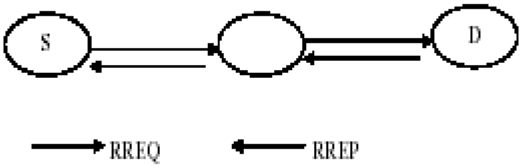


**Figure 4: Route discovery in AODV**

### 3.2 Route Maintenance

If a source node moves in between then it reinitiates the route discovery process. If a link to the node fails then that node should inform about breakage of link to source node by sending RERR message. The source again reinitiates route discovery process. In order to maintain connectivity between nodes AODV regularly send HELLO message to nodes. AODV uses sequence number to ensure freshness of the route. If there are multiple routes with same sequence number then route with smallest sequence number is chosen.

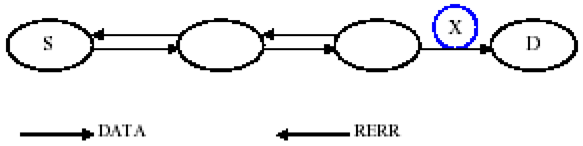


**Figure 5: Route Maintenance in AODV**

### 3.3 Data forwarding

In this process the nodes in between stores the address of neighbor from where first packet was received. If more than one copy of RREQ are received then they are discarded. When RREQ reaches the destination node it generates route reply RREP packet.

## 4.0 RELATED WORK

In this section we will study various national and international research papers and about the proposed techniques for malicious node detection in Mobile Adhoc Network. The research area related to this field is very large and complex. Here we will discuss some of them which are related to my proposed work.

Antonio M. Ortiz and Teresa Olivares proposed “Fuzzy Logic Applied to Decision Making in Wireless Sensor Networks” [7]. It states the application of fuzzy logic in decision making in wireless sensor network. The state that the developers should consider theoretical & practical issues when designing and implementing routing schemes. They proposed that with the use of fuzzy logic in decision making process of AODV protocol they can select best nodes to be the part of routes. Here they proposed fuzzy logic to improve the selection of routing metrics. It contains details of parameter selection, definition and fuzzy rule design. They have also showed the results in comparison to AODV by using AODV-ETX, an interesting metric used in wireless network. From results obtained they said that AODV-FL consumes less energy because it sends less messages thereby resulting in fewer collisions. Hence by using fuzzy logic as a metric in network routing improves the overall performance of the network.

B.Ben Sujitha1, R.Roja Ramani2, Parameswari3 proposed Intrusion Detection System using Fuzzy Genetic Approach [8]. It states that by using Fuzzy genetic algorithm FGA for intrusion detection we can detect new attacks and handle them. They state that IDS are effective against attacks. Various changes are done to IDS to detect new and malicious attacks. In this paper they introduced FGA. The FGA approach is based on if-then rules along with genetic algorithm. This method is tested on KDD’ 99 benchmark dataset and they are compared with already existing techniques. They state that implementation of FGA showed effective results in field of IDS. They also said that in future FGA algorithm can also be used to minimize computation time.

Sampada Chavan, Neha Dave and Sanghamitra Mukherjee proposed Adaptive Neuro-Fuzzy Intrusion Detection Systems [9]. It states that two paradign, Artificial Neural Network & Fuzzy Inference System are used for IDS. They proposed SNORT in order to perform traffic analysis & packet logging on IP network during the training phase of system. Then they constructed signature pattern database using Neuro Fuzzy learning method. They also state that 40% of original number of input variables, we can improve the performance & development time. In future IDS can also produce results by examining input from different sources.

Bharanidharan Shanmugam and Norbik Bashah Idris proposed Improved Intrusion Detection System using Fuzzy Logic for Detecting Anamoly and Misuse type of Attacks [10].They proposed a hybrid model based on fuzzy & data mining techniques which can detect any type of attack. They proposed to reduce the data retained for processing which includes selection process of attribute & to improve IDS using data mining technique. They have used KUoK fuzzy data mining algorithm which is the modified version of APRIORI for implementing fuzzy if-then rules. The proposed model is tested against DARPA 1999 data set for efficiency. The future work is to turn the system into light weight system by overcoming drawbacks such as bottleneck in packet processing & improve the performance of faster detection and alert correlation. The future work is to make this system as an open source project and get ready for real world challenges.

Devendra K. Tayal, Amita Jain and Vinita Gupta [19] proposed Fuzzy Expert System for Noise Induced Sleep Disturbance and Health Effects. They have developed a fuzzy based model in detecting noise effects on health and sleep disturbance. They have implemented their work in MATLAB 7.0.1. They have developed Fuzzy MIMO Expert system to predict the health conditions in noisy region.

This reference motivated me to apply fuzzy logic approach in my work.

## 5.0 PROPOSED WORK

The proposed work consists of four phases namely Path generation using AODV algorithm, applying Fuzzy logic, verification and detection of malicious nodes.

**Phase One: Path generation using AODV algorithm**

In this phase AODV algorithm is applied to generate path for route discovery. AODV uses all its features to generate the path from source to destination.

**Phase Two: Applying Fuzzy logic**

In this phase the generation of fuzzy rules takes place along with membership function. The fuzzy IF-THEN rules are applied in order to detect malicious node.

**Phase Three: Verification**

In this phase verification of IF-THEN rules takes place. The condition of IF statement verified by checking if the destination sequence number is much greater than source sequence number and if response time of node is greater than set threshold value then malicious node is detected

**Phase Four: Detection of malicious node**

In this phase we will be able to detect the malicious node by applying fuzzy rules.

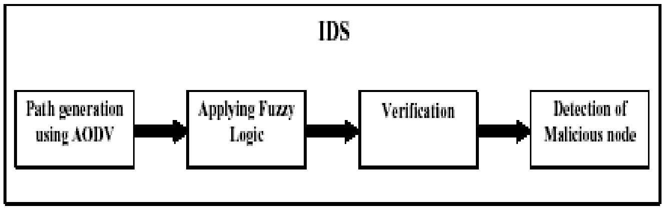


**Figure 6: Proposed Fuzzy based IDS**

### 5.1 Fuzzy Rules

Rule 1) If (source sequence is low) OR (response time is high) then output is medium

Rule 2) If (source sequence is low) AND (response time is medium) then output is high

Rule 3) If (response time is medium) then output is low.

Rule 4) If (source sequence is high) then output is low.

## 6.0 RESULTS AND DISCUSSION

Here we will evaluate our model Intruder Detection system in MANET using Fuzzy Logic. This system is developed to operate anywhere in any situation, therefore the experiment is carried out with same scenario with different experiments that shows the performance of system. The parameters used for simulation will be compared to the existing model.

It is very important to choose suitable parameters for system evaluation. The performance parameters will describe the result of simulation. These parameters are important as they will be used to notify what will actually happen during simulation

Our choice is using MatLab- 2010. Matlab uses the hierarichal architecture in order to define components like nodes & network. The components are defined by text based language. The components can be nested to form complex module inside each other.

Every module can be accomplished by C++ file which describe its behavior. MatLab provides many modules such as queues, tools etc. by using C++ computation. MatLab uses documentation & active discussion forums.

| | |
|---|---|
| The experiments were carried out by MatLab-2010. The scenarios developed to carry out the tests use as parameters the mobility of the nodes and the number of active connections in the network. Node are presented previously were utilized in the experiments. The choices of the simulator are presented in table 1 | Matlab-2010 |

| | |
|---|---|
| Simulation Area | 50*50m |
| No of nodes | 10 to 100 |
| Transmission range | 25m |
| Mobility Model | Random Waypoint |
| Max Speed | 5-20 m/sec |
| Traffic Type | CBR(UDP) |
| Data payload | 512 bytes |
| Packet rate | 2 packet/sec |
| No of malicious nodes | 3 |
| Simulation time | 30 sec |
| Routing Protocol | AODV |
| MAC | 802.11 |
| Pause Time | 10 sec |
| Mobility | 10.70 m/s |
| Terrain area | 100*100m |

**Table 1: Measurements in MATLAB**

### 6.1 Validation in terms of metrics used for comparison

The performance comparison is based on various metrics between existing AODV with proposed AODV using fuzzy logic.

### 6.2 Throughput

It is defined as total no of delivered packets divided by the total duration of simulation time.

Throughput = (Packets sent / Packet Total) *100

### 6.3 Hop Count

It is variation in time stuck between packets inward caused by network congestion & due to route changes.

### 6.4 Execution Time

It is the time used by algorithm for execution

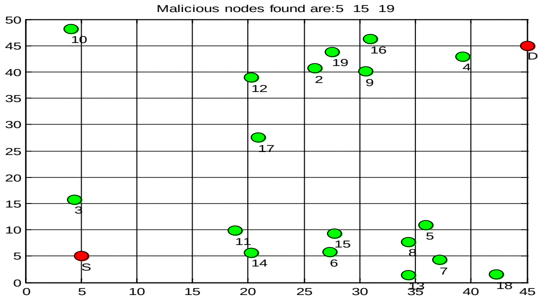


**Figure 7: Displaying no of nodes in 50*50 area**

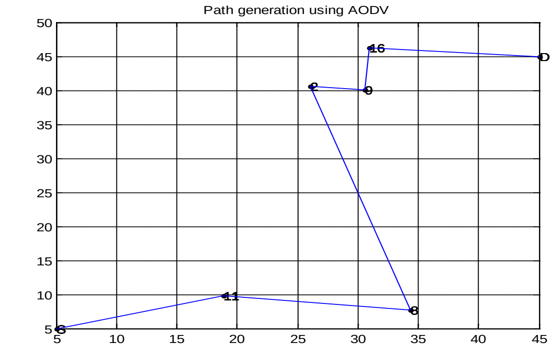


**Figure 8: Line showing path from source to destination**

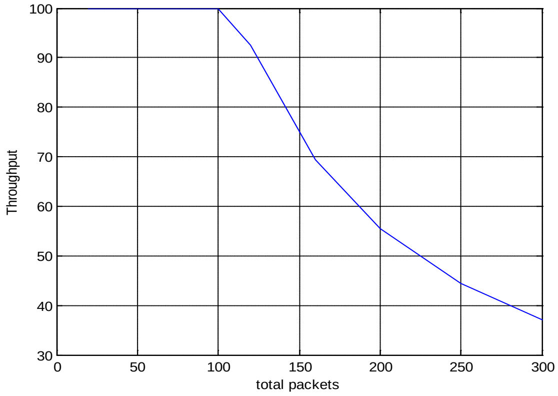


**Figure 9: Graph displaying throughput**

| RREQ SENDS to Nodes: | Acknowledgement Received from Nodes: |
|---|---|
| 11<br>14<br>6<br>15<br>17<br>8<br>5<br>12<br>2<br>9<br>19<br>16<br>4<br>20 | 11<br>8<br>2<br>9 |
| Selected Path is:<br>1 11 8 2 9 16 20 | Hop Count is: 6<br>Elapsed time is 1.365541 seconds.<br>Malicious count=3 |

**Table 2: Results showing RREQ and ACK packets along with execution time and malicious count**

**FIS Editor**

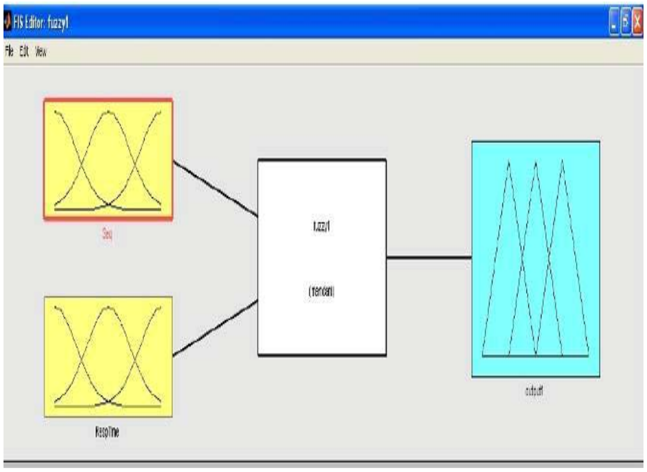
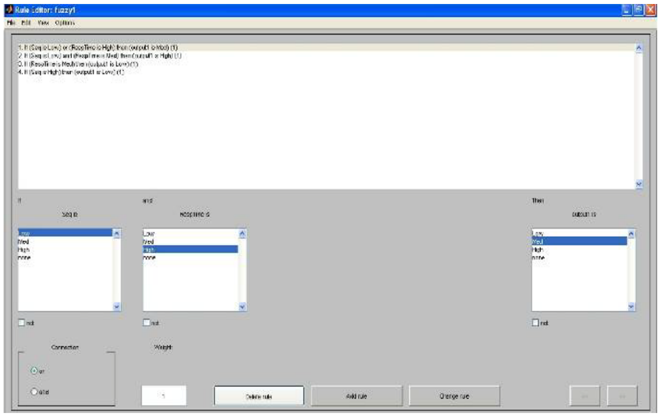


**Figure 10: FIS Editor**

**Rule Editor**



1. If (Seq is Low) or (RespTime is High) then (output1 is Med) (1)
2. If (Seq is Low) and (RespTime is Med) then (output1 is High) (1)
3. If (RespTime is Med) then (output1 is Low) (1)
4. If (Seq is High) then (output1 is Low) (1)

**Figure 11: Rule Editor displaying the if-then rules used**

**Rule Viewer**

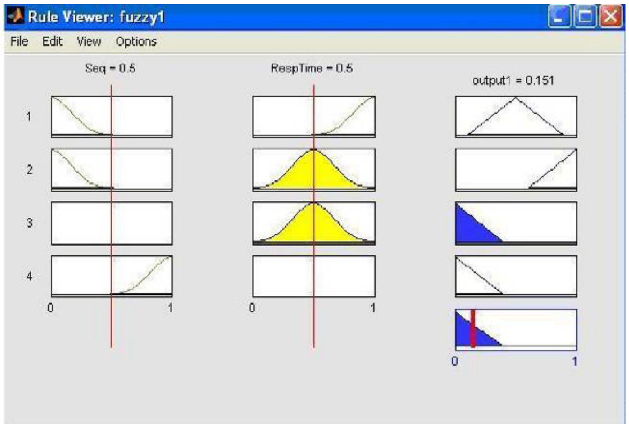


**Figure 12: Rule Viewer diagram in fuzzy logic**

**Surface Diagram**

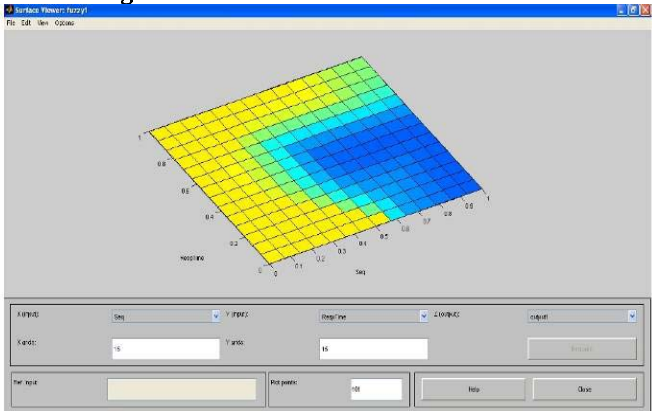


**Figure 13: Surface diagram displaying the no of malicious nodes**

## 7.0 CONCLUSION & FUTURE WORK

The security of MANET has gained popularity among research area. The security issues are discussed and we have analyzed the security system with our proposed model Intruder Detection System in MANET using Fuzzy Logic. This model is very efficient for protecting against attacks. Our proposed model can find the safe route and helps in preventing attack in MANET by identifying the node with sequence no & threshold value.

The proposed scheme is implemented using Matlab & its results show its effectiveness. The method will check for the difference between source sequence number & destination sequence number; if the source sequence no is greater & crosses the threshold value then that node is said to be malicious node. Mainly the malicious node will give fast route reply with high destination sequence number Moreover on identifying the malicious node the routing table and messages from malicious node are not forwarded in network. Our proposed algorithm has shown great improvement in Hop count, execution time and throughput. The proposed solution does not require any type of overhead on destination node or any intermediate node on AODV routing protocol. We have also used fuzzy IF-THEN rules to identify and delete attacks. The fuzzy rule is implemented by using response time from node. The algorithm will provide better solution for reduction of data loss over network.

Fuzzy logic is a rule based approach for solving the problem rather than automating the model. The proposed system will improve the performance of MANET under attack. The results have shown that proposed system has better performance than classic AODV in all its parameters like execution time, throughput, hop count etc. Our system not only detects the attack but also isolates it from network thereby improving the performance to great level.

The future scope of work is that the proposed security mechanism may be extended to defend against other attacks like grey hole attack, packet dropping attack, resource consumption attack. In order to detect attacks, various fuzzy rules can be generated by applying neuron fuzzy application.

**Surface Diagram (Lateral View)**

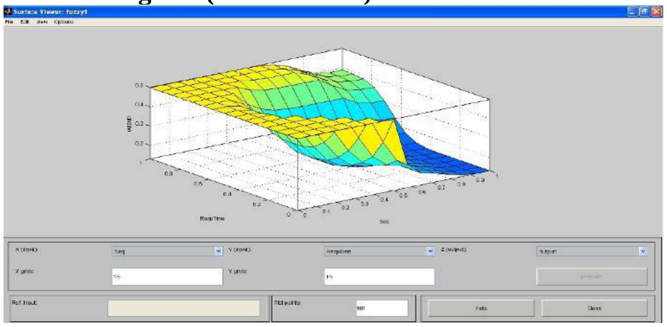


**Figure 14: Surface diagram displaying the no of malicious nodes (Lateral view)**

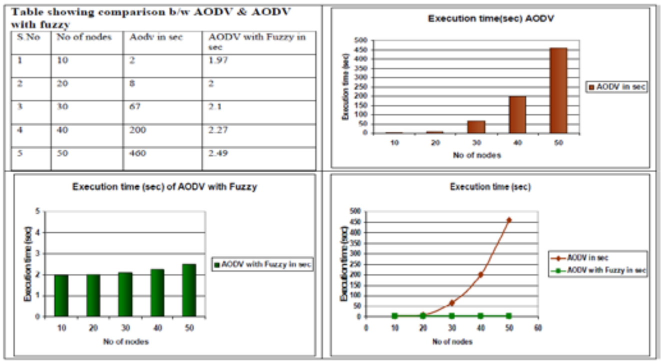


Table showing comparison b/w AODV & AODV with fuzzy

| S.No | No of nodes | Aodv in sec | AODV with Fuzzy in sec |
|---|---|---|---|
| 1 | 10 | 2 | 1.97 |
| 2 | 20 | 8 | 2 |
| 3 | 30 | 67 | 2.1 |
| 4 | 40 | 200 | 2.27 |
| 5 | 50 | 460 | 2.49 |

**Figure 15: Graph showing the execution time (sec) of AODV & AODV with fuzzy logic**

| No of nodes | Round | Malicious count | Rate | Rate*2500 |
|---|---|---|---|---|
| 100 | 10 | 3 | 3.00% | 7500 |
| 200 | 20 | 3 | 2.00% | 5000 |
| 300 | 30 | 6 | 2.00% | 5000 |
| 400 | 40 | 7 | 1.75% | 4375 |
| 500 | 50 | 9 | 1.80% | 4500 |

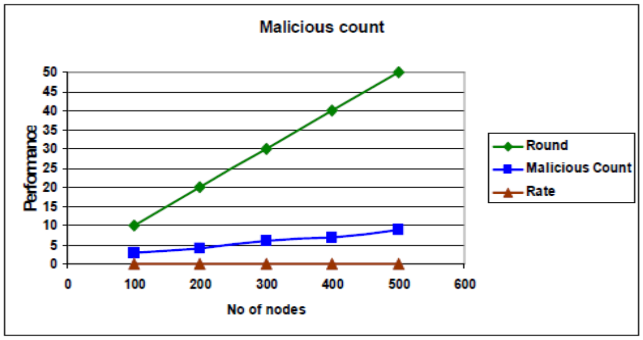


**Figure 16: Graph showing malicious count & rate% of AODV & AODV with fuzzy**

## 8.0 ACKNOWLEDGEMENT

I would like to thank the institution who helped a lot in this study. Also, special thanks to my guide and co-guide in sparing their time providing their valuable inputs as and when required during the course. Many thanks to all the peers, colleagues and organizations who supported this research by sharing their

valuable time and feedback on their experiences in application of Fuzzy logic in Intruder Detection System.

## 9.0 REFERENCES

[1]. Adhoc & Sensor Network by Carloss De Morais, ISBN-981-256-681-3 Pg [1] [2].

[2]. I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor net-works: a survey. Computer Networks, 38(4):393–427, 2002.

[3]. Dong Seong Kim, Khaja Mohammad Shazzad, and Jong Sou Park. A framework of survivability model for wireless sensor network. IEEE Proceedings of the First Interna-tional Conference on Availability, Reliability and Security, February 2006.

[4]. Boole G., "The Laws of Thought", New York: Dover Books (Reprinted), 1958.

[5]. Zadeh, L. A., "From Circuit Theory To Systems Theory", Proceedings of the Institute of Radio Engineering, Vol.50, 1962.

[6]. Zadeh, L. A., "Fuzzy Sets", Information And Control, Vol. 8, 1965

[7]. Antonio M. Ortiz and Teresa Olivares: Fuzzy Logic Applied to Decision Making in Wireless Sensor Networks, Fuzzy Logic – Emerging Technologies and Applications ISBN 978-953-51-0337-0

[8]. B.Ben Sujitha1, R.Roja Ramani2, Parameswari: Intrusion Detection System using Fuzzy Genetic Approach; International Journal of Advanced Research in Computer and Communication Engineering, Vol. 1, Issue 10, December 2012

[9]. Sampada Chavan, Neha Dave and Sanghamitra Mukherjee"Adaptive Neuro-Fuzzy Intrusion Detection Systems" Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'04) 0-7695-2108-8/04 $ 20.00 © 2004 IEEE

[10]. Bharanidharan Shanmugam and Norbik Bashah Idris."Improved Intrusion Detection System using Fuzzy Logic for Detecting Anamoly and Misuse type of Attacks" 2009 International Conference of Soft Computing and Pattern Recognition

[11]. Mohammad Al-Shurman and Seong-Moo Yoo, Seungjin Park "Black Hole Attack in Mobile Ad Hoc Networks" ACM SouthEast Regional Conference 2004.

[12]. Satoshi Kurosawa, Hidehisa Nakayama, Nei Kato Abbas Jamalipour, and Yoshiaki Nemoto; "Detecting Blackhole Attack on AODV-based Mobile Ad Hoc Networks by Dynamic Learning Method", International Journal of Network Security, Vol.5, No.3, PP.338–346, Nov. 2007, PP:338-346.

[13]. X. Wang, T. Lin and J. Wong, "Feature selection in intrusion detection system over mobile ad-hoc network," Technical Report, Computer Science, Iowa State University, 2005.

[14]. N.H.Mistry, D. C. Jinwala, M. A. Zaveri. "Prevention of Blackhole Attack in MANETs". In: Proceedings of EPWIE-2009, Gujarat, India, pp.89-94, July 2009.

[15]. Charles E. Perkins and Elizabeth M. Royer. "Ad-Hoc On-Demand Distance Vector Routing." In: Proceedings of the Second IEEE Workshop on Mobile Computing Systems and Applications (WMCSA'99), pages 90–100, February 1999.

[16]. Latha Tamilselvan, V Sankaranarayanan. "Prevention of Blackhole Attacks in MANET." In: Proceedings of the 2nd International Conference on Wireless Broadband and Ultra Wideband Communications (AusWireless 2007), pp. 21-21, Aug. 2007.

[17]. M. A. Shurman, S. M. Yoo, and S. Park, "Black hole attack in wireless ad hoc networks." In: Proceedings of the ACM 42nd Southeast Conference (ACMSE'04), pp 96-97, Apr. 2004.

[18]. Elmar Gerhards-Padilla," Detecting Black Hole Attacks in Tactical MANETs using Topology Graphs", 32nd IEEE Conference on Local Computer Networks 0742-1303/07© 2007 IEEE.

[19]. Devendra K. Tayal, Amita Jain and Vinita Gupta "Fuzzy Expert System for Noise Induced Sleep Disturbance and Health Effects" in BIJIT Issue3: (Jan-June 2010 Vol2 No1)

[20]. Zaheeruddin, Vinod K. Jain, and Guru V. Singh , "A Fuzzy Model For Noise-Induced Annoyance", IEEE transactions on systems, man, and cybernetics –Part A: Systems and Humans, Vol. 36(No. 4), July 2006.

# Framework for Multi-Agent Systems Performance Prediction Process Model: MASP$^3$

**S. Ajitha**[1], **T. V. Suresh Kumar**[2] and **K. Rajanikanth**[3]

*Submitted in August, 2013; Accepted in September, 2014*

***Abstract - Multi Agent System is one of the upcoming areas in the research/industry for building complex distributed application. MAS encompass multiple features for distributed application. For complex applications the non-functional requirements has the same importance as functional requirements. The non-functional requirements are performance, reliability, maintainability etc. We are focusing on the prediction of performance by considering the characteristics of agents in the early stages of MAS development. Evaluation of the performance at the end of software development leads to increase in the cost of design change. To compare design alternatives or to identify system bottlenecks, the quantitative system analysis must be carried out from the early stages of the software development life cycle. In this paper we propose a framework for predicting performance of a Multi-Agent System by considering the characteristics of agents in the early stages of software development.***

***Index Terms – Multi-Agent System, Performance Prediction Process Model, UML, Agent Characteristics, Software Performance Engineering.***

## 1.0 INTRODUCTION

Designing and building high quality industrial-strength software is difficult. Indeed, it has been claimed that such development projects are among the most complex construction tasks undertaken by humans. Each successive development either claims to make the engineering process easier or to extend the complexity of applications that can feasibly be built. Although there is some evidence to support these claims, researchers continually strive for more efficient and powerful software engineering techniques, especially as solutions for ever more demanding applications are required. There are compelling arguments for believing that an agent oriented approach will be of benefit for engineering certain complex software systems. These arguments have evolved from a decade of experience in using agent technology to construct large-scale, real world applications in a wide variety of industrial and commercial domains [1]. When adopting an agent oriented view of the world, it soon becomes apparent that a single agent is insufficient. Most problems require or involve multiple agents: to represent the decentralized nature of the problem, the multiple loci of control, the multiple perspectives, or the competing interests. Some of the important characteristics of the agents which distinguish an agent from objects are Autonomous, Cooperation, Goal oriented, Adaptability, Mobility, Negotiation etc. Analyzing, designing and implementing software as a collection of interacting, autonomous agents i.e., as a multi-agent system [2,3] represent a promising point of departure for software engineering. Whatever the complexity of the system the quality of the system cannot be neglected. The important non functional characteristics of the systems are performance, reliability, availability, maintainability etc.

Performance is an important but often neglected aspect of software development methodologies. Performance refers to system responsiveness, either the time required to respond to specific events, or number of events processed in a given time interval. Performance problems may be so severe that they require extensive changes to the system architecture. If these changes are made late in the development process, they can increase development costs, delay deployment, or adversely affects other desirable qualities of a design, such as understandability, maintainability, or reusability. Finally, designing for performance from the beginning produces better systems than using a 'fix-it-later' approach. Software Performance Engineering (SPE) has evolved over the past years and has been demonstrated to be effective during the development of many large systems [4]. Although the need for SPE is generally recognized by the industry, there is still a gap between the software development and the performance analysis domains. In this paper we propose a framework for predicting the performance of MAS using SPE techniques by considering the characteristics of agents in the MAS.

## 2.0 RELATED WORK

Agent-Oriented Software Engineering (AOSE) is being described as a new paradigm for the research field of Software Engineering. Some of the very widely used AOSE methodologies are MESAGE, MasE, Gaia, Tropos [5]. In [6] the author discusses the importance of performance engineering in Agent systems and suggested to define benchmarks and metrics that help to compare and contrast different Agent systems to support software engineering themes within Agent systems. In [7] estimating costs for agent oriented software is discussed. The scalability issue in Multi Agent System (MAS) is addressed in [8]. The main objective of this author was to combine performance engineering with agent oriented design methodologies to design and build large agent based applications. The author presents a solution for performance engineering of mobile agent systems during the development of agent code. In [9] a novel approach for performance improvement of Multi-Agent based system architecture is discussed. The authors also proposed a metrics suit for evaluating agent-oriented architectures. Most of the metrics are

[1, 2, 3] *M. S. Ramaiah Institute of Technology, Bangalore*
*E-mail:* [1]*ajithasankar@gmail.com*

inspired by the object-oriented metrics but they are adapted to agent oriented concepts. Williams and Smith in [10] applied the SPE methodology to evaluate the performance characteristics of a software architecture specified by using the Unified Modeling Language (UML) diagrams, ITU Message Sequence Chart (MSC) features. The extension to the SPE process and its associated models for assessing object oriented distributed systems are described in [11]. The use of SPE for web applications during software architectural design phase is discussed in [12]. An extension of the SPE approach was developed by Cortellessa and Mirandola in [13]. The proposed methodology, called PRIMA-UML, makes use of information from different UML diagrams to incrementally generate a performance model representing the specified system. The technique considered OMT based Class diagrams, Interaction diagrams and State Transition diagrams to specify the software systems and defined an intermediate model, called Actor-Event Graph, between the specification and the performance model. SAP-ONE [14] is a methodology for the performance modeling of a software system. It consists essentially in the annotated generation of a queuing network from an UML architectural model. Even though several of these approaches have been successfully applied, still there is a gap in integrating the software performance prediction process into the SLDC. We propose a framework based on the paper [15].

## 3.0 SOME CRITICAL ISSUES

- In general the software performance prediction starts from the analysis phase and continues throughout the SDLC. Most of the researches in MAS are concentrated on the AOSE methodologies, and implementation methodologies, agent theories, architectures and tools. The SPE approach for MAS is not addressed in the literature.
- Accessing the performance of MAS during the feasibility study of MAS development.
- Performance of the agents by considering the different characteristics (cooperation, negotiation, mobility, etc )
- Performance issues such as load balancing, scheduling and resource allocation for MAS in the context of SPE can be addressed.

## 4.0 PROSPOSE METHODOLOGY

The proposed frame work for predicting the performance of MAS in the early stages of software development is expressed in the form of flowchart in Figure1. The activities involved in the elements of the process model are:

1. **Consider the Agent Characteristics:** MAS are formed by the coalition of more than one agent. The characteristics of agents are autonomous, cooperation, negotiation, mobility etc which distinguishes the agents from objects. These characteristics of agents have a high influence on the performance of the system. So consider the characteristics of agents and define the SPE assessments for the application considered. In our work we considered the characteristics cooperation and negotiation of agents.
2. **Define the SPE assessments for a given software application:** SPE is applied to predict the performance of software systems early in the development life cycle. In our framework the possibility of assessing the performance during the feasibility study of agents by considering the characteristics of the agent is addressed. We have considered the SPE assessments such as acceptable response time or constraints on resource requirements based on the approach on [10].
3. **Develop UML models augmented with performance parameters:** Capture the performance requirement data by modeling the application using a modeling technique. Since UML is a universally accepted modeling language, we have extended the UML to model the application developed using agents
4. **If the transformation technique is necessary to get the performance models, apply the transformation technique and derive the performance models from UML models:** Use traditional transformation algorithms, e.g graph grammar techniques, XSLT, to generate the performance model from UML models. In our work we have devised an algorithm to transform a UML sequence diagram into agent execution graph along with the demand vector by considering the agent character "Cooperation".
5. **Alternatively, generate Performance Models from UML models:** Generate performance models from UML models by mapping the elements. We have proposed to map the elements of the UML models to the ANN model.
6. **Solve the model:** The models can be solved analytically or by simulation. Simple systems can be solved analytically, whereas simulation is preferred for larger and complex systems. So we propose to devise algorithms for solving the models analytically as well as by simulation.
7. **Report the performance metrics,** If the performance metrics obtained by considering one of the agent characteristic is acceptable, then proceed the same cycle for the next character of the agent
8. **Alternatively,** if the performance metric obtained for the particular characteristic is acceptable then the same methodology can be extended to the other characteristics.

## 5.0 CASE STUDY

Agent technology is adopted by different areas of applications [16,17]. The case study we have considered to validate our methodology is a sub module of supply chain management system [18]. The module we have considered consists of three agents namely Production Agent, Supply agent and Delivery agent. The use case diagram is used to represent a high level abstraction of the system. The use case diagram consists of the agents and the use cases. From this diagram we can identify the number of agents in the system and the interaction of the agents with the use cases. The use case diagram is presented in Figure2. The model is simulated using the tool SMTQA(Simulation of Multi Tier Queing Applications) for 1000 requests[19]. We have considered MAS with 3 agents. The inputs required for simulation are: software resource requirements, execution environment, software execution structure, and resource usage. The probability of occurrence of the use cases is considered from the cooperative index of the agents. The size of the use cases is estimated using the use case point approach. We have assumed the processing speed of the server as 2000 KB/sec and the speed of the internet considered is 96.8 KB/sec. The mean arrival rate we considered is 0.05. The performance metrics for the agents in the MAS are obtained and tabulated in the Table1.

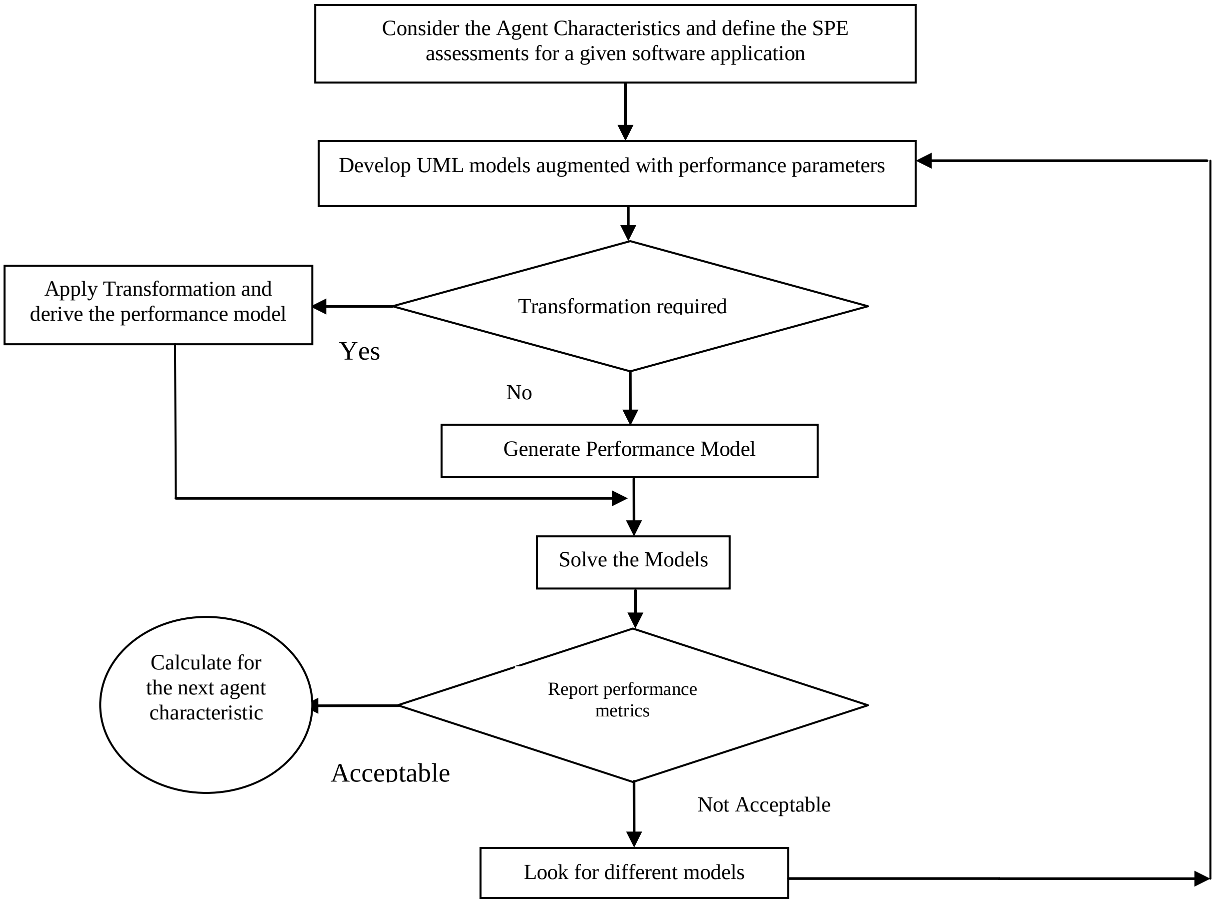

Consider the Agent Characteristics and define the SPE assessments for a given software application
Develop UML models augmented with performance parameters
Transformation required
Apply Transformation and derive the performance model
Yes
No
Generate Performance Model
Solve the Models
Report performance metrics
Calculate for the next agent characteristic
Acceptable
Not Acceptable
Look for different models

**Figure1: Framework for Performance Prediction Process Model**

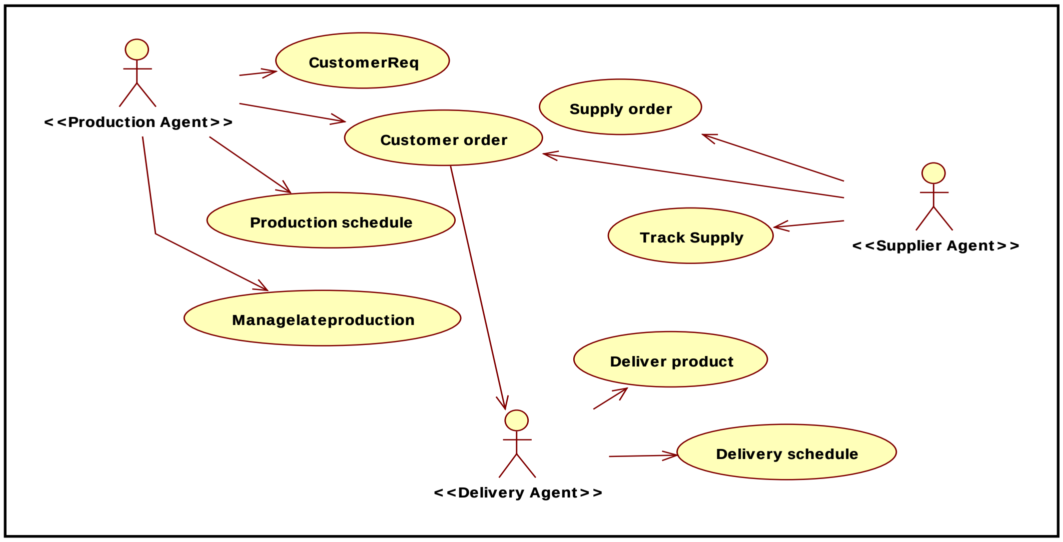


**Figure 2: Use Case diagram for case study**

| | Average Response time (Sec) | Average Service Time(Sec) | Average Waiting Time(Sec) | Probability of Idle Server(Sec) | Probability of Dropping of Sessions(Sec) |
|---|---|---|---|---|---|
| Production Agent | 0.417 | 0.040 | 0.377 | 0.0700 | 0.627 |
| Supply Agent | 0.023 | 0.019 | 0.004 | 0.724 | 0.000 |
| Delivery Agent | 0.020 | 0.017 | 0.003 | 0.771 | 0.000 |
| Internet | 0.014 | 0.011 | 0.003 | 0.621 | 0.017 |

**Table 1: Performance Metrics obtained for MAS using SMTQA**

## 6.0 CONCLUSION

This paper discusses a framework for MASP3 model that allows modeling MAS with the goal of assessing performance of the system during the feasibility study and early in the Software Development Life Cycle (SDLC). This framework describes the elements of MASP3 model and provides flexibility to integrate the software performance prediction process with SE process. The proposed MASP3 Model provides the possibility of deriving performance models from Unified Modeling Language (UML) models by mapping the UML model elements into the elements of multitier architecture simulation model and solving the simulation model. The description and significance of each element in the process are described in this paper. We considered a case study in supply chain management system to validate the framework.

## 7.0 REFERENCES

[1]. N. R. Jennings and M. Wooldridge (eds,) (1998) "Agent technology: foundations, applications and markets" Springer Verlag.

[2]. G.M.P. O'Hare and N. R. Jennings (editors) (1996) "Foundations of distributed artificial intelligence" John Wiley & Sons.

[3]. Gomaa H, Menasce D.A: "Design and Performance Modeling of Component Interconnection Patterns for Distributed Software Architectures, Proceedings of the WOSP 2000, ACM Second International Workshop on Software and Performance, Ottawa, Canada, September 2000.

[4]. Connie U. Smith, Performance Engineering of Software Systems, Reading, MA Addison-Wesley, 1990.

[5]. Lind J. Issues in Agent-oriented Software Engineering. The first International workshop on Agent oriented Software Engineering (AOSE-2000).

[6]. Omer Rana, Chris Preist, Michael Luck: "Progress in Multi-Agent Systems Research, March 2000

[7]. Jorge J. Gomez-Sanz, Juan Pavon, Francisco Garito: "Estimating Costs for Agent Oriented Software", TIC2002-04516-C03-03

[8]. Elema Gomez-Martinez Sergio Ilarrai Jose Merseguer "Performance Analysis if Mobile Agent tracking, WOSP '07, Feb G. Schneider and J. P. Winters: "Applying Use Cases, SecondEdition", Addison Wesley (2001).

[9]. Ivan Garcia-Magarino, Maximo Cossentino, Valeriascidita "A metric suit for evaluating agent oriented architecture", SAC'10-Mar 22/26.2010, Sierre, Switzerland ACM978-1-60558-638-0/10/03

[10]. C.U.Smith and Lioyd G.Williams., Performance Engineering Models of CORBA-based distributed-object systems. s.l. : Performance Engineering Services and Software Engineering Research, 1998.

[11]. Peter Utton, et al., Performance Analysis of Object-oriented Designs for Distributed Systems. University of Kent at Canterburry : Technical Report, 1999

[12]. Connie U.Smith and Lioyd G. Williams., "Building Responsive and Scalable Web Applications." December 2000. Proceedings CMGC

[13]. V. Cortellessa and R.Mirandola, "Deriving a Queueing Network Based Performance Model from UML Diagrams." s.l. : ACM Proc. intl, 2000. Workshop Software and Performance. pp. 58-70.

[14]. A. Di Marco, P. Inverardi "Compositional generation of Software Architecture Performance QN Models". Proc. 4th Working IEEE/IFIP Conference on Software Architecture (WICSA'04), 2004.

[15]. D.Evangelin Geetha, Dr.T.V.Suresh Kumar, Dr.K.Rajanikanth,"Framework for Hybrid Performance Predition Process Model: Use Case Engineering Approach"ACM Sigsoft Software Engineeering Notes, May 2011,Volume 36 Number 3.

[16]. Tomasek.M, Trelova. J, "An e-commerce applications based on the multi-agent system," Emerging eLearning Technologies & Applications (ICETA), 2012 IEEE 10th

International Conference on vol., no., pp.391,394, 8-9 Nov. 2012, Stara Lens

[17]. Islam, S. R.; Sutanto, D.; Muttaqi, K.M., "Application of multi-agent system for preventing power interruption in a large power system," Harmonics and Quality of Power (ICHQP), 2012 IEEE 15th International Conference.

[18]. T.Moyaux, B.Chaib-draa and S.D'Amours. Multiagent based Supply Chain Management, chapter Supply Chain Management and multiagent Systems, Springer-Verlag Berlin Heidelberg, 2006.

[19]. D.Evangelin Geetha, T.V. Suresh Kumar, Performance Modeling and evaluation of Distributed Systems, Ph.D thesis, Visvesvaraiah Technological University, Karnataka, 2012.

# Stable Adhoc on Demand Multipath Distance Vector – SAOMDV

**Bhavna Arora** and **Nipur**

*Submitted in October, 2013; Accepted in September, 2014*

***Abstract - SAOMDV is a multipath routing protocol for mobile ad hoc network that find multiple paths for the data packet without flooding the entire network with Route request – RREQ packets, but selectively forwarding to only those neighbors that are stable in terms of distance and link. SAOMDV is based on the protocol AOMDV. SAOMDV indentifies the stable neighbors and instead of blindly broadcasting the RREQ packet it receives, it only forwards the RREQ to these stable nodes.***

***Index Terms – Multipath, On-demand, Selective flooding, Stable routes.***

## 1.0 INTRODUCTION

A Mobile Ad hoc Network (MANET) is a collection of wireless mobile nodes dynamically forming a temporary network without the use of any existing network infrastructure or centralized administration. MANET nodes are typically distinguished by their limited power, processing, and memory resources as well as high degree of mobility. Routing in one of the main area of research, many routing protocols have been proposed both unipath as well as multipath. The basic categories of routing protocol is either table driven or on demand. All protocols proposed so far fall in one of the category, the basic being Destination sequenced distance vector [1], Wireless routing protocol [2], Dynamic source routing [3], Ad-hoc on demand distance vector [4] to name a few. Various routing protocols are being proposed by incorporating swarm intelligence as in [5], where the routes are selected based on ant agents. Application based routing protocol has been proposed in [6] where route is selected based on the application, it uses a home agent for the selection of the optimal network depending upon the type of session of the mobile nodes.In order to make the best use of the scarce resources that are available in a MANET multipath routing protocols are proposed. Multipath routing protocols provide many benefits in the overall performance of the network in terms of fault tolerance, load balancing, usage of bandwidth and lower delay. Load balancing can be achieved by spreading the traffic along multiple routes. This can alleviate congestion and bottlenecks. Multiple paths are used simultaneously to route data, the aggregate bandwidth of the paths may satisfy the bandwidth requirement of the application [7]. Also, since there is more bandwidth available, a smaller end-to-end delay may be achieved. Results of [8] show that using multipath routing in adhoc networks of high density results in better throughput than using unipath routing.

We propose an on demand multipath routing protocol Stable Ad hoc On demand Multipath Distance Vector - SAOMDV, based on AOMDV. SAOMDV is capable of finding link disjoint multiple paths which are more stable and robust. In SAOMDV, the mobile ad hoc network is not flooded by RREQ storm for route discovery as flooding route requests often results in broadcast storm (especially when nodes or connections increase) [9]. The remainder of the paper is organized as follows: in thenext section we give a brief overview of the AOMDV protocol. Section III illustrates our protocol SAOMDV in more detail. Section IV presents evaluation of the proposed protocol by simulation in NS2. Section V puts light on future work.

## 2.0 ADHOC ON DEMAD MULTIPATH DISTANCE VECTOR ROUTING

Ad-hoc On-demand Multipath Distance Vector Routing (AOMDV) [10] protocol is an extension to the AODV protocol for computing multiple loop-free and link disjoint paths. In AOMDV, *RREQ* propagation from the source towards the destination establishes multiple reverse paths both at intermediate nodes as well as the destination. Multiple *RREP*s traverse the reverse paths back to form multiple forward paths to the destination at the source and intermediate nodes. Intermediate nodes also keep track of alternate paths to the destination node. Nodes cannot broadcast duplicate *RREQ*s, so any two *RREQ*s arriving at an intermediate node via a different neighbor of the source could not have traversed the same node. In an attempt to get multiple link-disjoint routes, the destination replies to duplicate RREQs, the destination only replies to RREQs arriving via unique neighbors. The routing entries for each destination contain a list of the next-hops along with the corresponding hop counts. All the next hops have the same sequence number. This helps in keeping track of a route. For each destination, a node maintains the advertised hop count, which is defined as the maximum hop count for all the paths, which is used for sending route advertisements of the destination. Each duplicate route advertisement received by a node defines an alternate path to the destination. Loop freedom is assured for a node by accepting alternate paths to destination if it has a less hop count than the advertised hop count for that destination. Because the maximum hop count is used, the advertised hop count therefore does not change for the same sequence number. When a route advertisement is received for a destination with a greater sequence number, the nexthop list and the advertised hop count are reinitialized.

[1] *Research Scholar, Dept. of Computer Science, Gurukul Kangri Vishvidyalya, Haridwar. India.*
[2] *Professor, Dept. of Computer Science, Kanya Gurukul Campus, Dehradun, India.*
*Email: [1]shubhavna@yahoo.com and [2]nipursingh@hotmail.com*

AOMDV can be used to find node-disjoint or link-disjoint routes. To find node-disjoint routes, each node does not immediately reject duplicate RREQs. Each RREQs arriving via a different neighbor of the source defines a node-disjoint path. This is because nodes cannot be broadcast duplicate *RREQ*s, so any two *RREQ*s arriving at an intermediate node via a different neighbor of the source could not have traversed the same node. In an attempt to get multiple link disjoint routes, the destination replies to duplicate *RREQ*s, the destination only replies to *RREQ*s arriving via unique neighbors. After the first hop, the *RREP*s follow the reverse paths, which are node disjoint and thus link-disjoint.

## 3.0 STABLE ADHOC ON-DEMAND MULTIPATH DISTANCE VECTOR

### 3.1 Stable-Neighbor Discovery

The neighbor discovery in SAOMDV is done by broadcasting the Hello packets periodically at the time interval of 1 second [11]. When a node receives a hello packet it examines the received signal strength indicator (rssi), if this value is above the predefined threshold value then the receiving node adds the sender of the Hello packet as a Stable Neighbor. If the rssi value is below this threshold value, then the sender of the hello packet as a Neighbor. The RSSI value for two ray ground model is calculated as given in (1).

$$P\tau(d) = \frac{Pt * Gt * Gr * ht^2 *hr^2}{d^4 * L} \quad (1)$$

Where Pr: Power received at distance d, Pt: Transmitted signal power, Gt: Transmitter gain (1.0 for all antennas), Gr: Receiver gain (1.0 for all antennas), d: Distance from the transmitter, L: Path loss (1.0 for all antennas), ht: Transmitter antenna height (1.5 m for all antennas), hr: Receiver antenna height (1.5 m for all antennas). In SAOMDV routing protocol each node of the network maintains two neighbor list one is the normal neighbor list and another as Stable neighbor list. Also, the neighbor list has an extra field which is a Boolean variable that is set initially as 0 for all nodes and it is set as 1 for stable neighbor. The Stable neighbor cache contains the stable neighbor address, stable neighbor status which is 1 and stable neighbor link. The normal neighbor cache contains the same fields but its neighbor status is set to 0. The algorithm for stable neighbor discovery is given in fig1.

```
Algorithm 1: Stable Neighbor Discovery
Step1. When a node receives Hello packet, it calculates it rssi value
Step 2.If
 rssi value > threshold value
Add the sender node as Stable neighbor, set its status as 1
Else
Add the sender node as normal neighbor, set its status as 0
Step 3. Maintain separate neighbor cache for both types of neighbors
        a)   Stable Neighbor Cache
        b)   Neighbor Cache
```

**Figure 1: Stable Neighbor Discovery**

### 3.2 Route Discovery & Maintenance Phase

When a node wants to send a data packet to some other node and it does not have a valid path for that destination node then SAOMDV being an on-demand routing protocol initiates route the discovery phase. In the route discovery phase the source node broadcasts a route request packet (RREQ) in the network. When this RREQ packet reaches a node then this nodes first checks whether or not it has received this rreq before. If it has already received this rreq packet before then it does not forward it because it is a loop. The nodes check whether the rreq packet is destined for itself it generates a route reply packet (RREP) and uni casts it towards the source node via the reverse route formed by the RREQ packet.If the receiver of the RREQ packet is just a intermediate node then it forwards the RREQ packet to the set of stable neighbors instead of broadcasting it in the entire network. By doing so the network is stormed by RREQ packets saving the networks resources like energy of the nodes, congestion of the network and reduction in control overhead. Also from the list of stable neighbors only those stable nodes are chosen whose link queue occupancy is below a particular threshold value. By doing so only those nodes are allowed to take part in route discovery whose link quality is good as compared to other nodes.Also before forwarding the RREQ packet we adopt a cross layer approach by forwarding the RREQ only if the MAC layer is idle and not otherwise. By doing so again the number of retransmissions of RREQ packet is reduced. As retransmission of control packets only add up to the network congestion level. The algorithm for forwarding the RREQ packet at the intermediate node is as given in figure 2.

The reverse path links are also set up as the RREQ packet traverses the network. If the intermediate node has a path to the desired destination then it can generate a route reply packet and unicast it to the source node.

```
Algorithm 2: Forwarding of RREQ packet
When a node recives rreq packet
Step 1. Checks whether it has received rreq packet before.
    A)   If yes, the rreq packet is dropped.
    B)   If no, it checks if it is the destination node, if it is then it
         generates a rrep packet for the source node.
Step 2. If it is not the destination node then it should be one of the
intermediate node
Step 3. It looks up in its stable neighbor cache and selects only those
stable neighbors for forwarding the rreq packet whose link buffer
occupancy is below a threshold value.
Step4. Before actually forwarding the rSreq packet the node senses the
medium and forwards only when the mac layer is idle.
```

**Figure 2: Forwarding of rreq packet**

Once the stable routes are established between a source and a destination pair of nodes via stable neighbors as intermediate nodes, they are used for data transfer in the mobile ad hoc network. SAOMDV finds link disjoint paths which are more stable, thus lowering the probability of failure. If there happens a link breakage as it is common in mobile ad hoc networks because of changing topology and unpredictable nature of the wireless medium. In the case of route breakage, a route error (RERR) message is generated by the intermediate node and is sent upstream to inform other nodes about the route failure.

The route maintenance in SAOMDV is similar to that of AOMDV.

## 4.0 PERFORMANCE EVALUATION

### 4.1 Simulation Environment

Comparative simulation for both the protocols AOMDV and SAOMDV are carried out on Network Simulator-2 [12]. The simulation scenario is summarised in table 1. The traffic is constant bit rate (cbr) at 4.0 packets/s, size 512 bytes. The traffic density is dense comprising of 50 sources. The node's maximum speed of 20m/s, the variation in mobility is achieved by changing the pause time of the nodes from 0 (highly mobile) to 500 (quasi static). The dimension of the topography is 1000 x 1000 and the simulation is carried out for 500 s. The radio propagation model used is Two Ray Ground and the Mac is specified as IEEE802.11.

| | |
|---|---|
| Simulation time | 500 seconds |
| No. of nodes | 50 |
| Topology | 1000 x 1000 |
| Traffic type | Cbr |
| Rate | 4.0 packet/s |
| Pause time | 0, 100, 200, 300, 400,500 |
| No. of connections | 50 |
| Radio Propagation Model | Two Ray Ground |
| Mac type | IEEE 802.11 |

**Table 1: Simulation Parameters**

### 4.1 Simulation Results

In order to analyze the simulation results of SAOMDV in MANET, we compare its performance with AOMDV in terms of packet delivery ratio, end-to-end delay and number of dropped packets.

*A.* Packet delivery fraction: The fraction of the data packets delivered to the destinations to those generated by the sources.

*B.* Average end-to-end delay of data packets: It is defined as the mean time in seconds taken by the data packets to reach their respective destinations.

*C.* Number of Dropped Data Packets: Packet loss occurs when one or more packets of data travelling across a network fail to reach their destination. Reasons for packet loss can be many ranging from unpredictable nature of the wireless medium or route breakage.

Figure 4 shows the packet delivery ratio of SAOMDV being much higher than that of AOMDV as the routes used for data transfer are much more stable comprising of only stable nodes as defined in the protocol.

## 5.0 CONCLUSION AND FUTURE SCOPE

In this paper we have proposed a multipath routing prtocol which is capable of finding stable paths in the mobile ad hoc network. Also, by simulation we have comapred the performance of SAOMDV and AOMDV. It can be seen that performance of the proposed protocol is much in terms of packet delivery ratio, end-to-end delay and packet loss.In the future work, we will incorparate Energy conservation in SAOMDV, so as to make the proposed protocol power efficient. Also, performance of SAOMDV will be tested under varoius conditions such as by varying traffic pattern and scalibility of the protocol.

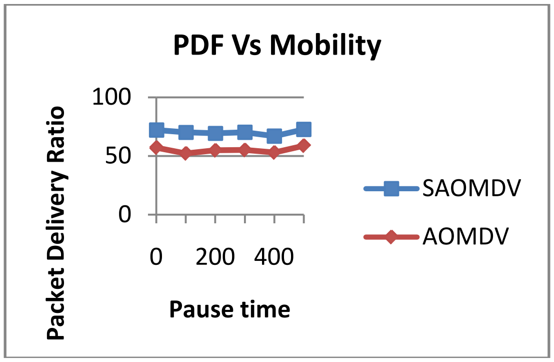


**Figure 4: PDF Vs Mobility**

Figure 5 demonstrates the end to end delay for the scenario that shows that SAOMDV has much lesser end to end latency as compared to the AOMDV routing protocol.

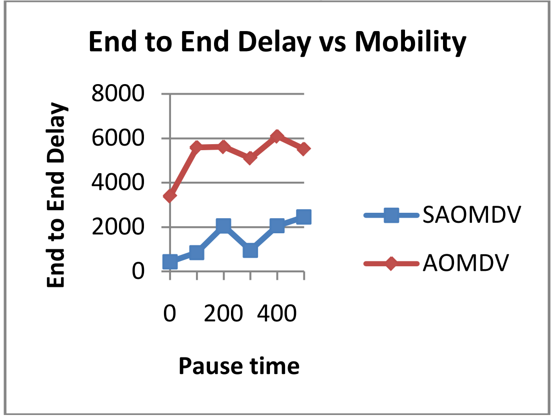


**Figure 5: End to End Delay Vs Mobility**

In figure 6 the number of dropped data packets are shown, again in SAOMDV the loss of data packets is much much lower than that in AOMDV.

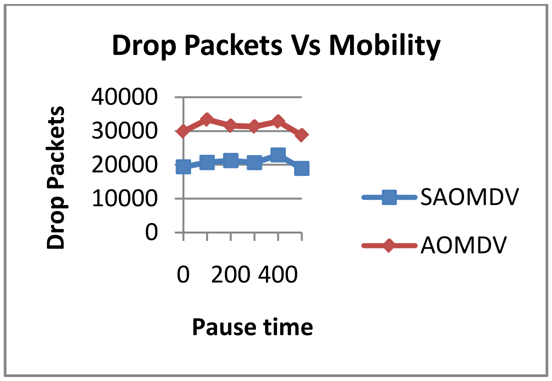


**Figure 6: Drop Packets Vs Mobility**

## 6.0 REFERENCES

[1]. Perkins CE, Bhagwat P. "Highly dynamic destination sequence distance vector (DSDV) routing for mobile computers". In: Proceedings of the ACM SIGCOMM, London, UK, 1994.

[2]. Murthy, Shree; Garcia-Luna-Aceves, J. J, "An efficient routing protocol for wireless networks", Mobile Networks and Applications pp: 183–197, 1996.

[3]. Johnson, D. and D. Maltz, "Dynamic Source Routing in Ad Hoc wireless Networks" T. Imielinski and H. Korth (Eds). Mobile computing Ch. 5, Kluwer, 1996.

[4]. Perkins, C.E. and E.M. Royer, 1999 "Ad-hoc on-demand distance vector routing." Proceedings of 2$^{nd}$ IEEE Workshop on Mobile Computing Systems and Applications.

[5]. Ramkumar K. R., Sakthivel K. and Ravichandran C. S. " ACBRAAM: A Content Based Routing AlgorithmUsing Ant Agents for Manets", BIJIT - BVICAM's International Journal of Information Technology Bharati Vidyapeeth's Institute of Computer Applications and Management (BVICAM), New Delhi, Volume 3, Number 1, January - June, 2011. Pg 276

[6]. Sulata Mitra and Arkadeep Goswami, "Load Balancing in Integrated MANET, WLAN and Cellular Network", ", BIJIT - BVICAM's International Journal of Information Technology Bharati Vidyapeeth's Institute of Computer Applications and Management (BVICAM), New Delhi, Volume 3, Number 1, January - June, 2011. Pg 304

[7]. R. P. T. a. D. G. S. Mueller, "Multipath Routing in Mobile Ad Hoc Networks:Issue & Challenges," Performance Tools and Applications to Networked Systems, Lecture Notes in Computer Science, vol. 2965, pp. 209-234, 2004.

[8]. P. Phamm and S. Perrau, "Performance analysis of reactive shortest path and multipath routing mechanism with load balance," in Proceedings of the IEEE INFOCOM, San Francisco, California, USA, 2003.

[9]. P.-J. Chuang, P.-H. Yen and T.-Y. Chu, "Efficient Route Discovery and Repair in Mobile Ad-hoc Networks," in Advanced Information Networking and Applications (AINA), Fukuoka, Japan, 2012.

[10]. S. R. D. Mahesh K. Marina, "Ad hoc on-demand multipath distance vector routing," WIRELESS COMMUNICATIONS AND MOBILE COMPUTING, p. 969–988, 2006.

[11]. I. D. Chakeres and E. M. Belding-Royer, "The Utility of Hello Messages for determining Link Connectivity," in proceedings of the 5th International Symposium on Wireless Personal Multiledia Communications (WPMC), Honolulu, Hawaii, 2002.

[12]. NS-2: Network Simulator ; http:// www.isi.edu/ nsnam/ns/.

# Certificate Based Security Services in Adhoc Sensor Network

**Shahin Fatima**[1], **Shish Ahmad**[2] and **P. M. Khan**[3]

*Submitted in March, 2014; Accepted in September, 2014*

***Abstract - The paper entitled "CERTIFICATE BASED SECURITY SERVICES IN ADHOC SENSOR NETWORK" proposed an approach in which the aim is to find the method for authentication which is more energy efficient and reduces the transmission time of the network. MANETs are of dynamic topology and have no predefined infrastructure. Due to its dynamic topology this network is prone to various kinds of vulnerable attacks. Sensor networks are battery operated and is a major concern. Methods on ID based Authentication consumes more network bandwidth and increases the computation and transmission time of the network. So for better operation, authentication must be the major factor of concern. In this paper a method for authentication in adhoc sensor network is proposed which is based on certificate based security services. Here we will make use of X.509 certificate format. In this some modification is made to the certificate format such that the transmission time and energy consumption of the network is reduced. Our proposed model will provide authentication among nodes and security in MANET. The proposed work is implemented in MATLAB and the result will show the effectiveness of proposed certificate in MANET. The objective of certificate based authentication is to ensure that messages can be read by authorized person only. It also overcomes the non repudiation attacks thereby minimizing the computation and shows how energy varies by making changes in certificate of node.***

***Index Terms – X.509 certificate, certificate authority (CA), authentication, confidentiality, securityHashing algorithm SHA-1, PrCA - Private Key of Certification Authority (CA), PuCA - Public Key of Certification Authority (CA).***

## 1.0 INTRODUCTION

With the advancement in technology the need for wireless communication has also increased. As we know wireless communication can reach eventually on every surface of the earth and to millions of people. One of the kind of network is MANET (Mobile Adhoc Network) in which nodes does not have any predefined infrastructure [1]. This contrast to cellular network in which BS (base station) act as access point. MANET consists of a group of nodes that communicate with each other without having any predefined infrastructure. For example it may be used in natural disasters such as earthquakes where fixed infrastructures have got damaged & in such cases. A MANET is an autonomous system of mobile nodes [7] and can be used as a communication network for a rescue team in case of emergency. In MANET network topology may dynamically change and nodes are free to move. Security means physical protection of system by using appropriate policies and cryptographic techniques.

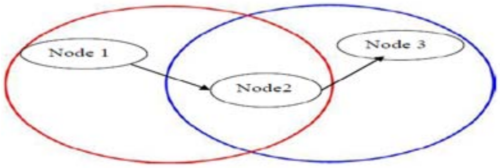


**Figure 1: Example of Mobile Adhoc Network**

As we can see in the figure it has three nodes, node 1, node 2 and node 3. Node 1 and node 3 are not within each other's range; hence node 2 can act as router to forward the packets from node 1 to node 2. Because we know that adhoc network are deployed randomly over a particular area so security here is a bit less important than security in various web services. Cryptography, then, not only protects data from theft or alteration, but can also be used for user authentication [14]. In Adhoc network physical protection of device is very important and is a great challenge. Therefore, we depend and rely on cryptographic techniques for prevention of attacks. While designing security methods for mobile ad hoc networks, consideration about the attacks variations and the characteristics of the attacks should be kept in mind that could be launched against the ad hoc networks [8].

### 1.1 Need for security

MANET's are practical and cost effective way for deployment of sensor networks. MANET's are used in large range of applications from civilian to military purposes. It throws different challenges as compared to traditional networks. Therefore different mechanisms can be brought about enormous research potential.

## 2.0 PUBLIC KEY ENCRYPTION

Asymmetric algorithm uses one key for encryption and same key for decryption. Symmetric encryption is vulnerable to brute force attack. To overcome from brute force attack the key size must be large enough. But because of large sizes the encryption & decryption speeds become too slow. Another way to attack is to find way to compute private key from the given public key. The history of cryptanalysis states that problems which seems to be insolvable can find a solution if looked from different way. Suppose for instance the message is to be sent & consist of 56 bit key. The attacker can encrypt all 56 bit key using public key & discover the encrypted key by matching the transmitted cipher text. Modern cryptography is basically designed for use on computers and no longer concerns about

[1, 2, 3]*Department of Computer Science and Engineering, Integral University, Lucknow*
*EMail:*[1]*shahinfatima@hotmail.com,*[2]*shish@iul.ac.in and* [3]*pmkhan@hotmail.com*

the written alphabet. Its focus is on the use of binary bits [10]. One of the main parts of the modern cryptosystem is quantum cryptography.

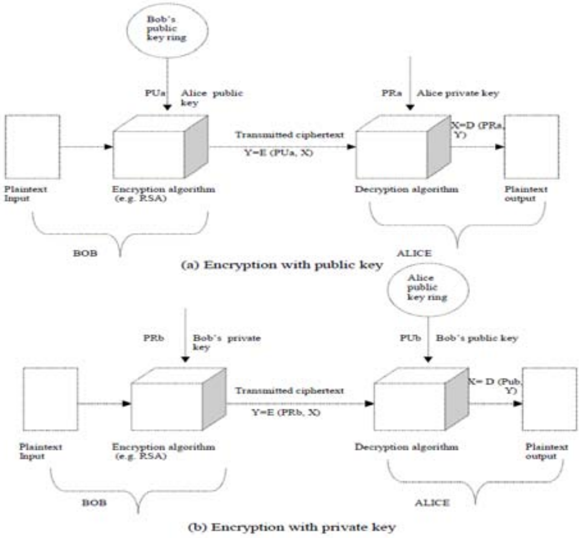


**Figure 2: Public Key Cryptography**

### 2.1 X.509 Authentication Service

X. 509 is a framework for authentication services .It uses Public-key Cryptography & defines authentication protocols. Certification authority signs the public key of user & stores the certificate in directory.

### 2.2 X. 509 Certificate Format

It is issued by certification authority CA & contains following fields:-
version which can be (1, 2, or 3)
serial number which is used for identification of certificate
signature algorithm identifier
issuer X.500 name i.e. name of CA
period of validity (from - to dates)
subject X.500 name (name of owner)
subject public-key info (algorithm, parameters, key)
issuer unique identifier
subject unique identifier

### 2.3 Need for X 509 Certificates

The X. 509 Certificate is needed because of following reasons:-

- X. 509 is more secure than using normal user ID or password
- It has trusted third party certifying authority which authenticates and distributes the digital certificates, thereby establishing a chain of trust.
- X. 509 also takes care of non repudiation attacks (i.e., the act of denying an action after the fact). For example, once someone uses a digital certificate and private key, the user cannot deny his action, because the private key resides with the user only.
- In X. 509 non authorised users face difficulty to extract the private key when stored on a smart card.

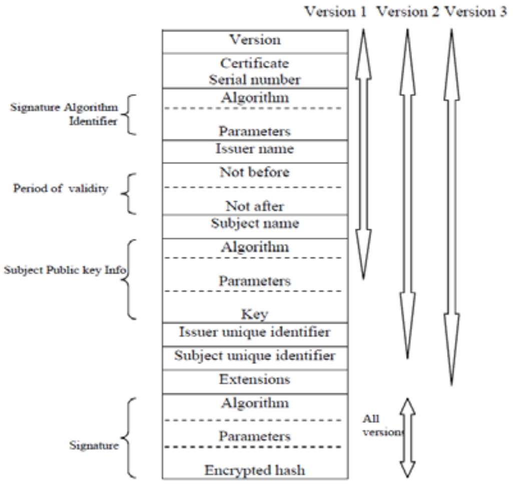


**Figure 3: Format of X. 509 Certificates**

### 2.4 Hashing Algorithm (SHA-1)

It is a cryptographic algorithm which is used to provide authentication & integrity of data. It is also used to avoid the need for storage of plaintext password in password based system. It is a function which takes input as block of data & returns the hash value in the form of fixed size string. It has the properties of primary resistance, second primary resistance and collision resistance. Hash Key management has been proposed as one of the best options for security, [9] although other options are also available depending upon need of security.

### 2.5 Encryption Algorithm (RSA)

This algorithm is based on factorising large prime numbers. This works on public & private key system. The public key is available to everyone [5]. This public key is used to encrypt the data. The decryption is done with the help of private key. This private key is associated with the user who will decrypt the message. The generation of private key from public key is very difficult therefore RSA algorithm is very popular for data encryption.

## 3.0 RELATED WORK

A mobile ad hoc network (MANET) consists of number of mobile stations connected by wireless links. It is known as infrastructure less network as it does not trust on predefined infrastructure. In MANET nodes can easily exchange information with nodes in its range and nodes which are beyond its range uses the concept of multihop communication.

Here we will study various national and international journals about the X 509 certificate and the proposed work for it in mobile ad-hoc network. There are various research papers related to this work & about the quality of X. 509. The research area related to this field is very broad. Following are few important research papers that describe the quality of X. 509. Mr. Vinod Saroha, Annu Malik, Madhu Pahal presents a survey on Digital Signature Certificate [2] which states about the digital signature, creation of digital signature, revocation of digital signature & authentication procedures. Digital signature ensures the identity of sender. A digital signature adds data electronically to any message in order to make it more authentic & more secured. Digital signature guarantees that once document is digitally signed then data cannot be tampered. Digital signature ensures security of message. The use of digital signature is to ensure that a user who is sending a message is the one who he/she claims to be.

Christian Bauer proposed an article "X.509 Identity Certificates with Local Verification" [3] which states that X. 509 identity certificate ensures authentication in communication system. A global trust anchor verifies this certificate which is accepted by communicating parties in order to authenticate each other. Because of non availability of services like certificate revocation services prevents proper authentication. In this paper X. 509 identity certificate is extended that allows authenticating parties to verify each other certificate in absence of global trust anchor. They have used this for describing their problem & giving the proposed solution. This paper states that revocation of certificate can be performed by using revocation service provided by trust anchor. M. Rameshkumar proposed "Design an Enhanced Certificate Based Authentication Protocol for Wireless Sensor Networks" [4] which states that WSN can use µTESLA and MULTILEVEL µTESLA symmetric method for encryption. µTESLA methods have drawback that they suffer from DoS attacks. Therefore to overcome the weakness of µTESLA this paper presents key based method to achieve authentication. To overcome from the computation cost of these schemes, techniques such as hash tree & identity based schemes have been adopted. They have used one way hash function h () and uses the hash pre images as keys in a message authentication code (MAC). Dilbag Singh anf Ajit Singh proposed [14] A Secure Private Key Encryption Technique for Data Security in Modern Cryptosystem. They have proposed private key encryption technique which can be used for security of data in encryption. This technique employs the concept of arithmetic coding and can be used in any encryption system. This reference motivated me to apply a form of public key cryptography in my work for security and authentication.

## 4.0 PROPOSED WORK

The characteristics of MANET such as dynamic topology & energy constrained operation are a challenging issue. Security is also an important issue in the field of MANET. Security leads to authentication among nodes. Many methods have been proposed to provide security & authentication among nodes in MANET. In our work we will make use of X.509 authentication certificate format. We will modify the certificate format such that the size of certificate is reduced thereby leading to reduction in transmission time & energy consumption. This system will make use of RSA algorithm for encryption and SHA-1 logic for hashing

### 4.1 Proposed Certificate (X. 509 M)

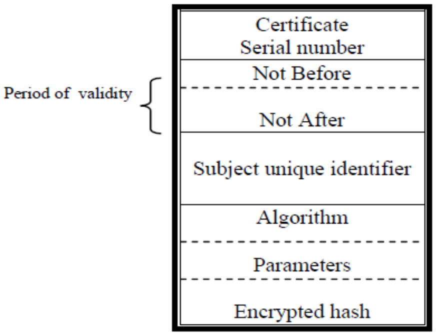


**Figure 4: The (X.509 M) Certificate**

In our proposed certificate the fields which are taken are as follows:-

- **Serial number:** It is an integer value unique within the issuing CA that is associated with the certificate.

- **Period of validity:** It consists of two dates: the first and last date on which the certificate said to be valid.

- **Subject unique identifier:** It is a bit string field which optional and is used to identify uniquely the subject.

- **Signature:** It covers all the fields of the proposed certificate. It also contains the hash value of all the other fields and encrypted with the CA (Certification Authority) private key. This field includes the signature algorithm identifier.

### 4.2 Security Model for MANET

The security model for MANET consists of encryption and decryption of signature among nodes in MANET.

#### 4.2.1 Encryption and sending signed message to Y

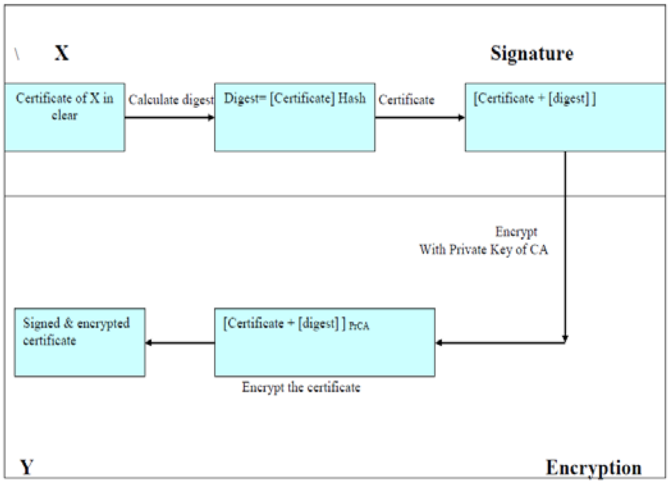

\ X
Signature
Certificate of X in clear
Calculate digest
Digest= [Certificate] Hash
Certificate
[Certificate + [digest] ]
Encrypt
With Private Key of CA
Signed & encrypted certificate
[Certificate + [digest] ] PrCA
Encrypt the certificate
Y
Encryption

**Figure 5: Signature and Encryption details with signature & key**

Figure shows the operation required when X wants to send a signed & encrypted certificate to Y.
It consists of following steps:-

#### 4.2.1.1 Certificate Signature

The signature includes two steps:-

- **Message Digest Evaluation**
  This is called as hashing. The main purpose for calculating digest is to ensure that message is unaltered & ensuring message integrity
- **Digest Signature**
  The signature is calculated by encrypting it with CA's private key. The hashing algorithm is also included in the signature. By using public key encryption & hashing algorithm the recipient has the proof that:
  - The CA's private key has also encrypted the digest
  - The message is not modified against any alteration.

#### 4.2.1.2 Message encryption

Encryption includes 3 steps:-

- **Creating encryption/decryption key**

Here we will create key for one time for encryption/decryption algorithm which is public and private key of CA.

- **Message encryption**

The whole message is encrypted with private key of CA.

- **Key used for Encryption**

Public key of CA is the key used by the receiver side to decrypt the message. Therefore public key of CA must be available to recipient only.

### 4.2.2 Decryption & Verification of signature of message

Figure shows the steps required when Y wants to decrypt & verify the message send by X.

#### 4.2.2.1 Message decryption

This involves the following steps:-

- **Message decryption**
  The certificate is now decrypted using public key of CA. The one time public key of CA is used to decrypt the message.

#### 4.2.2.2 Signature Verification

It includes the following steps as follows:-

- **Message digest decryption**
  The digest was encrypted using CA's private key. This can now be decrypted using CA's public key.
- **Digest Evaluation**
  Because hashing is only a one way process therefore the original message cannot be derived from certificate, the recipient has to calculate the hash again using the similar hashing algorithm as used by the sender.

- **Comparison of digest**
  The digest which got decrypted above & the digest evaluated above will be now compared. If both get match then the signature is said to be verified & recipient can accept the messages coming from the issuer.

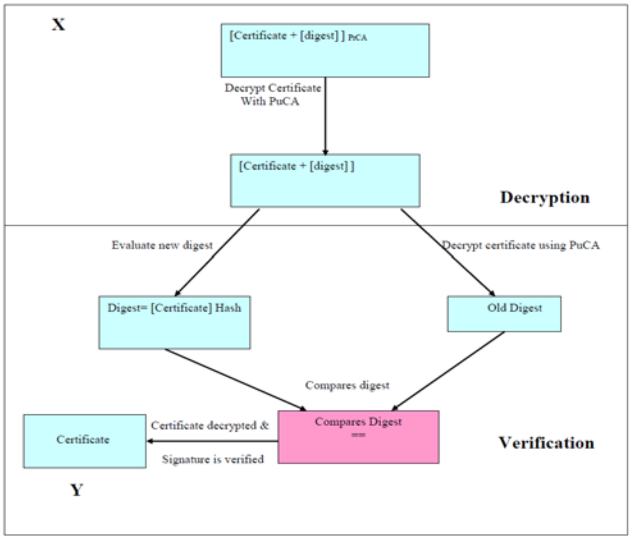


**Figure 6: Signature & Decryption details with certificate and keys**

If a mismatch occurs then it means that:-

- The message is not signed by CA's private key
- The message is altered
- In both the above cases the message should get rejected.

## 5.0 RESULTS AND DISCUSSION

Here we will evaluate our model Certificate based security services in Adhoc Sensor Network. The parameters used for simulation will be compared to the existing certificate. It is very important to choose suitable parameters for system evaluation. The performance parameters will describe the result of simulation. These parameters are important as they will be used to notify what will actually happen during simulation. MatLab- 2010 will be used as simulation tool because Matlab uses the hierarchal architecture in order to define components like nodes & network.

| | |
|---|---|
| The experiments were carried out by MatLab-2010. The scenarios developed to carry out the tests use as parameters the mobility of the nodes and the number of active connections in the network. Node are presented previously | Matlab-2010 |

| | |
|---|---|
| were utilized in the experiments. The choices of the simulator are presented in table 1 | |
| Simulation Area | 100*100m |
| No of nodes | 10 to 100 |
| Transmission range | 25m |
| Mobility Model | Random Waypoint |
| Max Speed | 5-20 m/sec |
| Traffic Type | CBR(UDP) |
| Data payload | 1500 bytes |
| Packet rate | 2 packet/sec |
| Sensor type | Crossbow MICA2DOT mote. |
| Simulation time | 30 sec |
| MAC | 802.11 |
| Pause Time | 20 sec |
| Mobility | 10.70 m/s |
| Terrain area | 100*100m |

**Table 1: Measurement of MATLAB**

### 5.1 Validation in terms of metrics used for comparison

In this section we will validate our thesis by comparing modified & original certificate based on various parameters. The sensor used for validation is Chipcon CC1000 radio in Crossbow MICA2DOT mote.

### 5.1.1 Energy Consumption in transmission & reception

$E_s= (E_{tx})$

$E_r= (E_{rx)}$

Where,

$E_{tx}$–It is the energy required to transmit a byte.

$E_{rx}$–It is the energy required to receive a byte.

Here we will calculate the energy consumption due to transmission/reception of varying certificate sizes.

The energy consumption in transmission of 1 byte is 28.6 µJ [4] respectively. The energy consumption in reception of 1 byte is 59.2 µJ [4] respectively. For proposed certificate the size is of 31 bytes, therefore total energy associated with proposed certificate in transmission is 886.6 µJ and 1835.2 µJ in reception respectively.

The original size of X.509 certificate is of 82 bytes; therefore total energy associated with proposed certificate in transmission is 2345.2µJ and 4854.4µJ in reception respectively.

### 5.1.2 Energy consumption on computation

Here we will calculate the computation overhead of the proposed schemes in terms of energy consumption. The energy consumption in 1 byte of computation is 7.6mJ respectively [4]. For proposed certificate the size of 31 bytes requires energy consumption as 235.6 mJ.

The original size of X.509 certificate is of 82 bytes; therefore total energy associated with proposed certificate is 623.2 mJ.

### 5.3.4 Transmission time

It is the amount of time from beginning till the end of message transmission. The cost of 1 byte in transmission is $8.8*10^{-4}$ msec [6] respectively. Therefore the transmission cost associated with total size of proposed certificate is $2.728*10^{-2}$ msec.

The cost associated with 82 bytes of original certificate is $7.216*10^{-2}$ msec.

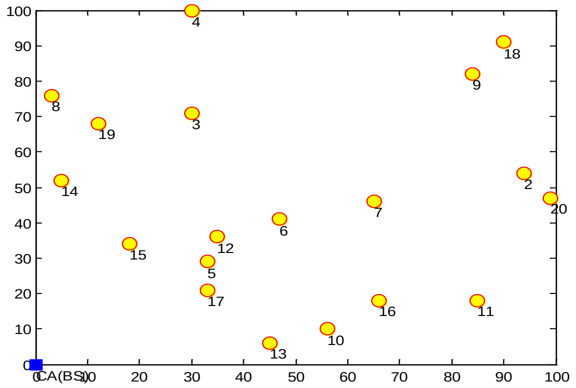
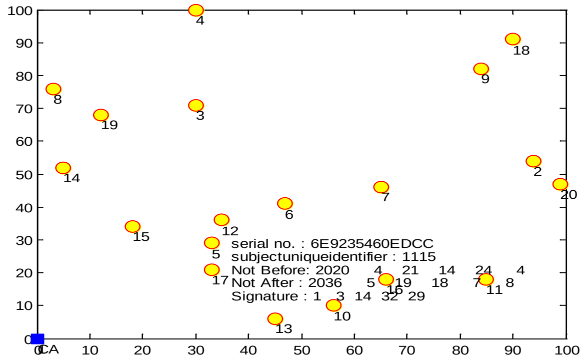


**Figure 7: Displaying no of nodes in 100*100 area with CA**



serial no. : 6E9235460EDCC
subjectuniqueidentifier : 1115
Not Before: 2020 4 21 14 24 4
Not After : 2036 5 19 18 7 8
Signature : 1 3 14 32 29

**Figure 8: Displaying certificate of a specified nodes**

### 7.0 CONCLUSION & FUTURE WORK

Security is an important issue for communication among nodes in Mobile Adhoc Network because of its important characteristics like infrastructure less and dynamic topology. By using X. 509 certificate we can provide better security & authentication among nodes in network. With the help of X. 509 certificates the network can be protected against unauthorized access. The advanced technology in adhoc network is facing issues related to proper key management. The security of MANET is coping up from these issues to provide better security. MANET consists of mobile nodes and because of its dynamic nature it faces many challenges. Our proposed model will provide authentication among nodes and security in MANET. The proposed work is implemented in MATLAB and the result will show the effectiveness of proposed certificate in MANET. The proposed work will initially reduce the size of original certificate which is then deployed among nodes in MANET by CA (Certification Authority). The proposed solution shows a great improvement over X.509 certificate in terms of computational energy, transmission/reception energy and transmission time.

The future scope of work is that the proposed authentication scheme of modified X.509 certificate can be used in MANET by forming clusters among nodes in MANET. Each cluster will have cluster head CH which will act as certification authority CA. This CA will a lot the certificate to its child nodes and keep the record of malicious nodes & original nodes in its respective cluster. This will improve the security among nodes in MANET. This technique can also be used on mobile Certification Authority (BS). The tedious task will be then how to select CA.

## 8.0 ACKNOWLEDGEMENT

I would like to thank the institution who helped a lot in this study. Also, special thanks to my guide and co-guide in sparing their time providing their valuable inputs as and when required during the course . Many thanks to all the peers, colleagues and organizations who supported this research by sharing their valuable time and feedback on their experiences in application of X.509 Authentication Certificate in Adhoc Sensor Network.

## 9.0 REFERENCES

[1]. Carloss De Morais “Adhoc & Sensor Network “, ISBN- 981-256-681-3 Pg [1] [2].

[2]. Mr. Vinod Saroha, Annu Malik, Madhu Pahal : The Enormous Certificate: Digital Signature Certificate International Journal of Advanced Research in Computer Science and Software Engineering Volume 3, Issue 6, June 2013 ISSN: 2277 128X

[3]. Christian Bauer: X.509 Identity Certificates With Local Verification First IEEE International Workshop on Security and Forensics in Communication Systems Institute of Communications and Navigation, German Aerospace Center (DLR), Wessling, Germany.

[4]. M.Rameshkumar ,“Design an Enhanced Certificate Based Authentication Protocol for Wireless Sensor Networks” Volume 2, Issue 10, October 2012 ISSN: 2277 128X International Journal of Advanced Research in Computer Science and Software Engineering

[5]. William Stallings “Cryptography and network security principles and practice fifth edition”, ISBN- 10: 0-13-609704-9 Pg [428].

[6]. Thomas Kunz, S.S.Ravi: Ad-hoc Mobile and Wireless Network: 5th International Conference on Adhoc Sensor Network, ISSN-0302-9743 Pg [174].

[7]. Ashema Hasti, “Study of Impact of Mobile Ad – Hoc Networking and its Future Applications” in BIJIT January - June, 2012; Vol. 4 No. 1; ISSN 0973 – 5658 439

[8]. B. B. Jayasingh1 and B. Swathi, “A Novel Metric for Detection of Jellyfish Reorder Attack on Ad Hoc Network” BIJIT – 2010; Jan – June, 2010; Vol. 2 No. 1; ISSN 0973 – 5658

[9]. Ashwani Kush1 and C. Hwang, “Hash Security for Ad hoc Routing”, BIJIT – 2011; January – June, 2011; Vol. 3 No. 1; ISSN 0973 – 5658

[10]. Dilbag Singh1 and Alit Singh, “An Effective Technique for Data Security in Modern Cryptosystem”, BIJIT – 2010; Jan – June, 2010; Vol. 2 No. 1; ISSN 0973 – 5658

[11]. National Institute of Standrads and Technology. Recomended elliptic curves for federal government use, 1997.

[12]. Albert Levi and Erkay Savas. Performance evaluation of public-key cryptosystem operations in WTLS protocol. In (ISCC’03), pages 1245–1250. IEEE Computer Society, 2003.

[13]. Richard Kuhn, Vincent Hu, Timothy Polk, and Shu-Jen Chang. Introduction to public key technology and the federal PKI infrastructure. NIST, February 2001.

[14]. Dilbag Singh and Ajit Singh “A Secure Private Key Encryption Technique for Data Security in Modern Cryptosystem” in BIJIT Issue 4: (July-December, 2010 Vol 2 No 2).

| | |
|---|---|
| **certificate for node 1**<br>serial no. : 6E9235460EDC8<br>subjectuniqueidentifier : 1111<br>Not Before: 2019 6 6 12 9 4<br>Not After : 2037 7 28 17 35 49<br>Signature : 1 3 14 32 29 | **certificate for node 2**<br>serial no. : 6E9235460EDCE<br>subjectuniqueidentifier : 1112<br>Not Before: 2021 12 1 21 37 60<br>Not After : 2032 6 25 6 30 55<br>Signature : 1 3 14 32 29 |
| **certificate for node 3**<br>serial no. : 6E9235460EDC7<br>subjectuniqueidentifier : 1113<br>Not Before: 2018 11 23 15 15 40<br>Not After : 2020 8 20 18 54 59<br>Signature : 1 3 14 32 29 | **certificate for node 4**<br>serial no. : 6E9235460EDCC<br>subjectuniqueidentifier : 1114<br>Not Before: 2020 7 28 14 2 8<br>Not After : 2038 6 26 6 34 38<br>Signature : 1 3 14 32 29 |
| **certificate for node 5**<br>serial no. : 6E9235460EDBC<br>subjectuniqueidentifier : 1115<br>Not Before: 2013 8 11 2 30 12<br>Not After : 2016 3 5 5 3 39 | **certificate for node 6**<br>serial no. : 6E9235460EDC2<br>subjectuniqueidentifier : 1116<br>Not Before: 2015 7 21 12 33 27<br>Not After : 2018 6 26 21 17 13 |

| | |
|---|---|
| Signature : 1 3 14 32 29 | Signature : 1 3 14 32 29 |
| **certificate for node 7**<br>serial no. : 6E9235460EDC6<br>subjectuniqueidentifier : 1117<br>Not Before: 2018 8 13 5 57 5<br>Not After : 2021 2 5 15 35 4<br>Signature : 1 3 14 32 29 | **certificate for node 8**<br>serial no. : 6E9235460EDCF<br>subjectuniqueidentifier : 1118<br>Not Before: 2022 9 23 2 52 57<br>Not After : 2042 11 24 13 11 24<br>Signature : 1 3 14 32 29 |
| **certificate for node 9**<br>serial no. : 6E9235460EDBE<br>subjectuniqueidentifier : 1119<br>Not Before: 2014 1 29 8 18 20<br>Not After : 2024 8 1 21 34 52<br>Signature : 1 3 14 32 29 | **certificate for node 10**<br>serial no. : 6E9235460EDC4<br>subjectuniqueidentifier : 1120<br>Not Before: 2016 6 2 5 40 20<br>Not After : 2034 2 30 13 43 60<br>Signature : 1 3 14 32 2 |

**Table2: showing certificates of 10 nodes**

| | |
|---|---|
| hash code of node(1):<br>hash coding time<br>0.7500<br>02AEF106<br>A9697608<br>1AFF4891<br>804B1BD3<br>906F0597 | Intaializing:<br>RSA encrpted certiicate :<br>75 84 109 137 42 26 75 10<br>109 63 10 63 132 10 75 23<br>26 109 42 42 171 23 63 26<br>23 75 171 77 26 77 85 68<br>63 75 10 42 75 25 63 132 |
| hash code of node(2):<br>hash coding time<br>0.3290<br>FA8E33BF<br>86310A61<br>64684DA1<br>BE52379C<br>3A5D4D41 | Intaializing:<br>RSA encrpted certiicate :<br>42 109 23 137 68 68 77 42<br>23 10 68 26 75 109 10 26<br>10 171 10 23 171 85 109 26<br>77 137 25 84 68 132 63 67<br>68 109 25 85 171 85 171 26 |
| hash code of node(3):<br>hash coding time<br>0.3280<br>EC059560<br>D62BC9AA<br>DF9F5D53<br>0245639C<br>F2C2E118 | Intaializing:<br>RSA encrpted certiicate :<br>137 67 75 25 63 25 10 75<br>85 10 84 77 67 63 109 109<br>85 42 63 42 25 85 25 68<br>75 84 171 25 10 68 63 67<br>42 84 67 84 137 26 26 23 |
| hash code of node(4):<br>hash coding time<br>0.3130<br>669977FB<br>08751621<br>93A2E205<br>7736C27F<br>B8D6489 | Intaializing:<br>RSA encrpted certiicate :<br>10 10 63 63 132 132 42 77<br>75 23 132 25 26 10 84 26<br>63 68 109 84 137 84 75 25<br>132 132 68 10 67 84 132 42<br>77 23 85 10 171 23 63 25 |

**Table3: Calculated hash codes along with encryption at all nodes**

| | |
|---|---|
| Enter the no. of nodes that are willing to communicate any no. from 1 to20:<br>**Enter no of first node:**<br>**Enter no of second node:**<br>**nodes 2 and 3 are selected to communicate** | Decrypted ASCII of Message:<br>137 67 75 25 63 25 10 75<br>85 10 84 77 67 63 109 109<br>85 42 63 42 25 85 25 68<br>75 84 171 25 10 68 63 67<br>42 84 67 84 137 26 26 23 |
| Decrypted Hash Message is:<br>EC059560<br>D62BC9AA<br>DF9F5D53<br>0245639C<br>F2C2E118 | Hash code of node 3 after decryption :-<br>EC059560<br>D62BC9AA<br>DF9F5D53<br>0245639C<br>F2C2E118 |
| **The hash code of selected node is similar before and after decryption, Hence Selected node is authentic to communicate** | |

**Table 4: Decryption at receiver side & comparison of calculated hash**

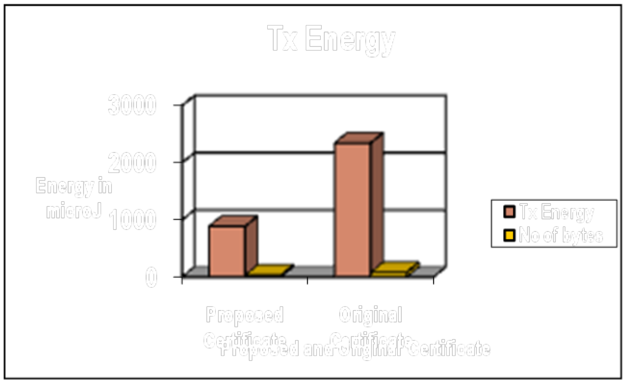


Figure 9: Transmission energy of proposed & original certificate

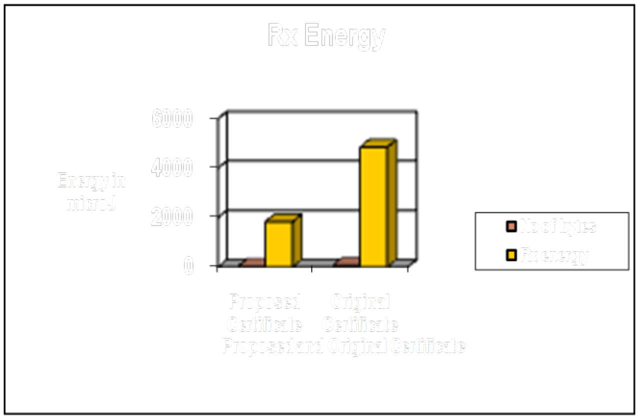


Figure 10: Reception energy of proposed & original certificate

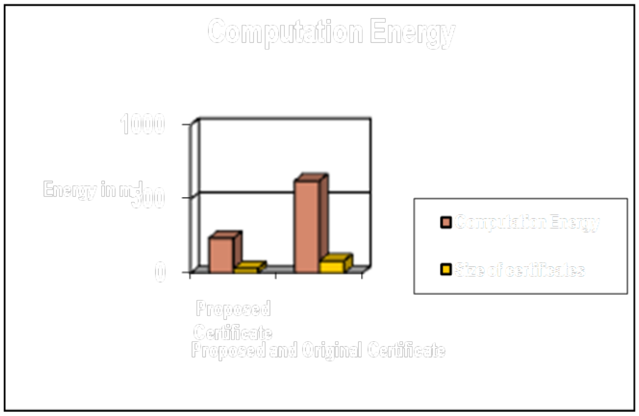


Figure 11: Computational energy of proposed & original certificate

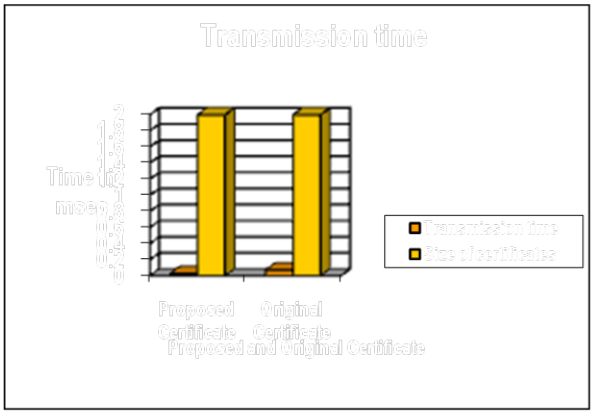


Figure 12: Transmission of proposed & original certificate

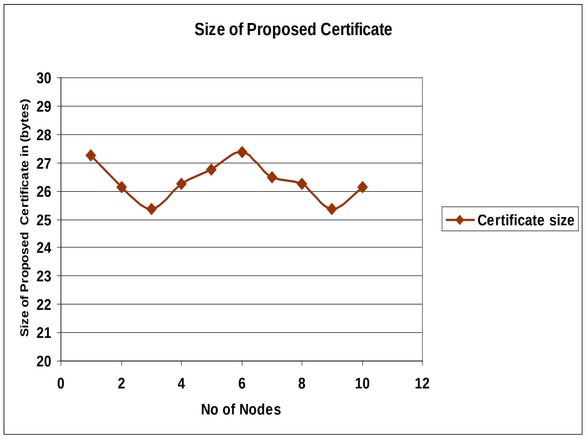


Figure 13: Size of proposed & original certificate with varying nodes.

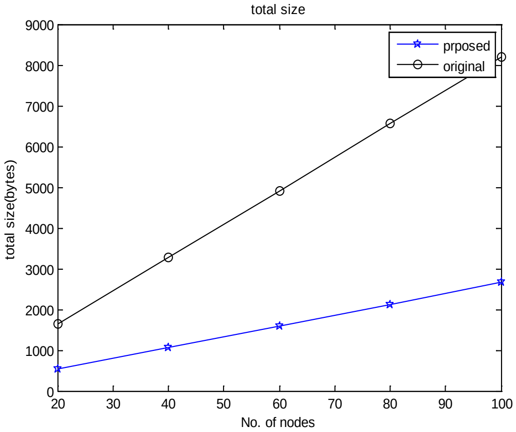


Figure 14: Size of proposed & original certificate with varying nodes.

# BIJIT - BVICAM's International Journal of Information Technology

(A Half Yearly Publication; ISSN 0973 - 5658)

## Subscription Rates (Revised w.e.f. January, 2012)

| | **1 Year** | | **3 Years** | |
|---|---|---|---|---|
| **Category** | **India** | **Abroad** | **India** | **Abroad** |
| Companies | Rs. 1000 | US $ 45 | Rs. 2500 | US $ 120 |
| Institution | Rs. 800 | US $ 40 | Rs. 1600 | US $ 100 |
| Individuals | Rs. 600 | US $ 30 | Rs. 1200 | US $ 075 |
| Students | Rs. 250 | US $ 25 | Rs. 750 | US $ 050 |
| Single Copy | Rs. 500 | US $ 25 | - | - |

---

## Subscription Order Form

Please find attached herewith Demand Draft No.________________ dated ___________________

For Rs._______________ drawn on ______________________________________________Bank in favor of **Director, "Bharati Vidyapeeth's Institute of Computer Applications and Management (BVICAM), New Delhi"** for a period of 01 Year / 03 Years

## Subscription Details

Name and Designation ___________________________________________________________

Organization ____________________________________________________________________

Mailing Address _______________________________________________________________

________________________________________ PIN/ZIP ____________________________

Phone (with STD/ISD Code)______________________________FAX____________________

E-Mail (in Capital Letters)________________________________________________________

Date: **Signature**

Place: (with official seal)

*Filled in Subscription Order Form along with the required Demand Draft should be sent to the following address:-*

**Prof. M. N. Hoda**
Editor-in- Chief, BIJIT
Director, Bharati Vidyapeeth's
Institute of Computer Applications & Management (BVICAM)
A-4, Paschim Vihar, Rohtak Road, New Delhi-110063 (INDIA).
Tel.: +91 – 11 – 25275055 Fax: +91 – 11 – 25255056 E-Mail: bijit@bvicam.ac.in
Visit us at: www.bvicam.ac.in/bijit

BHARATI VIDYAPEETH PUNE

**Bharati Vidyapeeth's**
**Institute of Computer Applications**
**& Management (BVICAM)**
**A-4, Paschim Vihar, Rohtak Road, New Delhi-63 (INDIA)**

**Technically Sponsored by**

**IEEE**
**Delhi Section**

**Supported by**

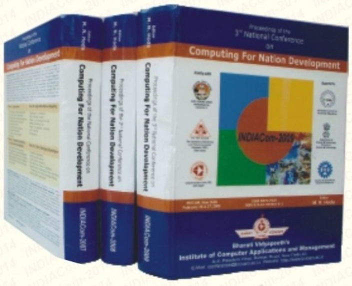
**GURU GOBIND SINGH**
**INDRAPRASTHA UNIVERSITY**

**The Institution of**
**Electronics and Telecommunication**
**Engineers (IETE), Delhi Centre**

**The Institution of**
**Engineering and Technology**
**Delhi Local Networks**

**CSI Region-I &**
**CSI Divisions- I, II, III, IV & V**

**ISTE, Delhi Section**



(Copies of the proceedings of past ***INDIACom***s)

**Correspondence**
All correspondences related to the conference
must be sent to the address:-

**Prof. M. N. Hoda**
General Chair, *INDIACom - 2015*
Director, BVICAM, A-4, Paschim Vihar, New Delhi -63 (INDIA)
Tel.: 91-11-25275055, TeleFax:91-11-25255056, 09212022066 (Mobile)
**E-Mails: conference@bvicam.ac.in, indiacom2015@gmail.com**
**visit us at: http://www.bvicam.ac.in/indiacom**

# INDIACom-2015

## 9th INDIACom; 2015 2nd International Conference on
## Computing for Sustainable Global Development
## (11th-13th March, 2015)
## IEEE Conference Record Number # 35071

**Paper Submission Link : http://www.bvicam.ac.in/indiacom/ SubmitPaper.asp.**

*INDIACom-2015* is aimed to invite original research papers in the field of, primarily, Computer Science and Information Technology and, generally, all interdisciplinary streams of Engineering Sciences, having central focus on sustainable computing applications, which may be of use in enhancing the quality of life and contribute effectively to realize the nations' vision of sustainable inclusive development using Computing. it is an amalgamation of four different international conferences, organized as parallel tracks. These are listed below:-

Track #1: International Conference on Sustainable Computing (ICSC-2015)
Track #2: International Conference on High Performance Computing (ICHPC-2015)
Track #3: International Conference on High Speed Networking & Information Security (ICHNIS-2015)
Track #4: International Conference on Software Engineering & Emerging Technologies (ICSEET-2015)

*INDIACom-2015* will be held at **Bharati Vidyapeeth, New Delhi (INDIA)**. The conference will provide a platform for technical exchanges within the research community and will encompass regular paper presentation sessions, special sessions, invited talks, key note addresses, panel discussions and poster exhibitions. In addition, the participants will be treated to a series of cultural activities, receptions and networking to establish new connections and foster everlasting friendship among fellow counterparts.

Full length original and unpublished research papers basedon theoretical or experimental contributionsre to the following topics, but not limited to, are solicited for presentation and publication in the conference:-

- Algorithms and Computational Mathematics
- Green Technologies and Energy Efficient Systems
- IT for Education, Health & Development
- IT for Environmental Sustainability
- IT for Sustainable Agriculture Development
- IT for Water Resources Management
- IT for Consumers' Right
- IT for Crisis Prevention & Recovery
- IT for Disaster Management and Remote Sensing
- IT for other day to day problems
- E-Governance
- Knowledge Management
- E-Commerce, ERP, CRM & Knowledge Mining
- Technology for Convergence
- Distributed and Cloud Computing
- Parallel, Multi-core and Grid Computing
- Reconfigurable Architectures
- Changing Software Architectural Paradigms
- Programming Practices & Coding Standards
- Software Inspection, Verification & Validation
- Software Sizing and Estimation Techniques
- Agile Technologies
- Artificial Intelligence and Neural Networks
- Computer Vision, Graphics, and Image Processing
- Modelling and Simulation
- Embedded Systems and Robotics
- Human Computer Interaction
- Databases
- Data Mining and Business Intelligence
- Big Data Analytics
- Operating Systems
- Data Communication, Computer Networks and Information Security
- Wireless Networking
- Network Monitoring Tools
- Next Generation Internet
- Mobile Computing
- Entertainment Technologies
- Multimedia Computing
- Information and Collaboration Systems
- Fuzzy and Soft Computing
- Bioinformatics
- Medical Informatics
- Education Informatics
- Computational Finance
- Research Methods for Computing
- Case Studies & Applications

## Paper Submission

Authors from across different parts of the world are invited to submit their original papers online at http://www.bvicam.ac.in/indiacom/ SubmitPaper.asp. Only electronic submissions will be considered. Papers submitted through E-mail, as attachment, will not be considered.

## Review Process, Publication and Indexing

All the submitted papers shall be doubled blind reviewed, by 03 experts, on the basis of their technical suitability, scope of work, plagiarism, originality, novelty, clarity, completeness, relevance, significance and research contribution. The shortlisted papers will be accepted for presentation and publication in the conference proceedings, having **ISSN 0973–7529** and **ISBN 978-93-80544-14-4** serials. Conference proceedings will also be available in soft copy. All accepted papers, which will be presented in the conference, will be submitted for publication and indexing to **IEEE Xplore.**

## Important Dates

| | | | |
|---|---|---|---|
| **Submission of Full Length Paper** | **10th November, 2014** | **Paper Acceptance Notification** | **12th January, 2015** |
| **Submission of Camera Ready Copy (CRC) of the Paper** | **22nd January, 2015** | **Registration Deadline** (for inclusion of Paper in Proceedings) | **22nd January, 2015** |