Advances in Intelligent Systems and Computing 870

Raj Kamal
Michael Henshaw
Pramod S. Nair *Editors*

International Conference on Advanced Computing Networking and Informatics

ICANI-2018

Springer

Advances in Intelligent Systems and Computing

Volume 870

Series editor

Janusz Kacprzyk, Systems Research Institute, Polish Academy of Sciences, Warsaw, Poland
e-mail: kacprzyk@ibspan.waw.pl

The series "Advances in Intelligent Systems and Computing" contains publications on theory, applications, and design methods of Intelligent Systems and Intelligent Computing. Virtually all disciplines such as engineering, natural sciences, computer and information science, ICT, economics, business, e-commerce, environment, healthcare, life science are covered. The list of topics spans all the areas of modern intelligent systems and computing such as: computational intelligence, soft computing including neural networks, fuzzy systems, evolutionary computing and the fusion of these paradigms, social intelligence, ambient intelligence, computational neuroscience, artificial life, virtual worlds and society, cognitive science and systems, Perception and Vision, DNA and immune based systems, self-organizing and adaptive systems, e-Learning and teaching, human-centered and human-centric computing, recommender systems, intelligent control, robotics and mechatronics including human-machine teaming, knowledge-based paradigms, learning paradigms, machine ethics, intelligent data analysis, knowledge management, intelligent agents, intelligent decision making and support, intelligent network security, trust management, interactive entertainment, Web intelligence and multimedia.

The publications within "Advances in Intelligent Systems and Computing" are primarily proceedings of important conferences, symposia and congresses. They cover significant recent developments in the field, both of a foundational and applicable character. An important characteristic feature of the series is the short publication time and world-wide distribution. This permits a rapid and broad dissemination of research results.

Raj Kamal · Michael Henshaw
Pramod S. Nair
Editors

International Conference on Advanced Computing Networking and Informatics

ICANI-2018

Springer

Editors
Raj Kamal
Medi-Caps University
Indore, Madhya Pradesh, India

Pramod S. Nair
Medi-Caps Group of Institutions
Indore, Madhya Pradesh, India

Michael Henshaw
Loughborough University
Loughborough, UK

ISSN 2194-5357 ISSN 2194-5365 (electronic)
Advances in Intelligent Systems and Computing
ISBN 978-981-13-2672-1 ISBN 978-981-13-2673-8 (eBook)
https://doi.org/10.1007/978-981-13-2673-8

Library of Congress Control Number: 2018955170

This Springer imprint is published by the registered company Springer Nature Singapore Pte Ltd.
The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721, Singapore

Preface

This AISC volume contains documented versions of the papers presented at the *International Conference on Advanced Computing, Networking and Informatics* (*ICANI-2018*). The conference was held during February 22–24, 2018, at Medi-Caps University, Indore (Madhya Pradesh), India. The conference was a platform for academicians, researchers, and industry delegates to present their research and contributions. The conference highlights the state-of-the-art as well as emerging research on computing, networking, and informatics. The objective of this international conference was to provide opportunities for the participants to interact and exchange ideas, experience, and expertise in the recent technological trends. Along with sharing, we had an array of lectures from eminent personalities in this field to bring value to our conference. The inauguration was held in the kind presence of Dr. A. K. Nayak (Secretary, CSI), Dr. Aditya Abhyankar (Dean, Faculty of Technology, University of Pune), and Dr. Raj Kamal, Dr. D. K. Mishra on the first day, the 22nd February, along with their enlightening talks. The conference had been a good opportunity for the participants from across the country. The sessions were a perfect learning with speakers from diverse expertise. The sessions were mentored by academic leaders like Dr. P. K. Chande (Director, NMIMS, Indore), Dr. N. Dagdee (Director, SDBCT, Indore), Mr. Piyush Bhadoriya (TCS), Dr. Sanjay K. Tanwani (Professor and Head, SCS, DAVV, Indore), Dr. C. N. S. Murthy (Director, Chameli Devi Group of Institutions, Indore), Dr. Ravi Sindal (Professor, DAVV, Indore), Dr. Pragya Shukla (Professor, DAVV, Indore), Dr. Shraddha Masih (Professor, DAVV, Indore), and Dr. Namrata Tapaswi (Professor, IPS Academy, Indore). The areas covered in the sessions included advanced computing and algorithms, informatics, networking, Web technology, big data and analytics, and recent technical topics that align with the theme of the conference. There were about 66 papers in ten sessions in the next 2 days of the conference that filled the gaps in the recent researches, which suggested new measures and tools for improvising the existing state of research, applications of the new techniques and innovations. The future of the new world that lies ahead in IoT and cloud had motivated several minds to provide ideas and insights on the programming and networking domains.

A committee of reviewers, external and internal, was formed for a rigorous peer review of the submitted papers which were around 225 in number. The shortlisting of papers was maintained keeping the reputation of the conference, and we achieved the acceptance ratio of 0.29. Due acknowledgment is to Prof. Aninda Bose who was a constant support for communications with the Springer publication.

Finally, we take the privilege to thank all the sponsors, press, print, and electronic media for promoting the conference.

Indore, India — Raj Kamal
Loughborough, UK — Michael Henshaw
Indore, India — Pramod S. Nair

Organizing Committee

Chief Patron

Shri R. C. Mittal, Chancellor, Medi-Caps University, Indore
Shri Gopal Agrawal, Pro-Chancellor, Medi-Caps University, Indore

Patron

Prof. (Dr.) Sunil K. Somani, Vice-Chancellor, Medi-Caps University, Indore

International Advisory Committee

Dr. M. A. Abido, Professor, King Fahd University, Saudi Arabia
Dr. Oscar Castillo, Professor, Tijuana Institute of Technology, Mexico
Dr. Mu-Song Chen, Professor, Dayeh University, Taiwan
Dr. Almoataz Youssef Abdelaziz, Professor, Ain Shams University, Egypt
Dr. T. K. Sarkar, Professor, Syracuse University, USA
Dr. Mo Jamshid, Professor, University of Texas, USA
Dr. C. J. Reddy, Vice President, Altair, USA
Dr. Georgi N. Georgiev, Professor, University of Veliko Tarnovo, Bulgaria
Dr. Hong Shen, Professor, University of Adelaide, Australia
Dr. Yoram Haddad, Professor, Jerusalem College of Technology, Israel
Dr. S. G. Ponnambalam, Professor, Monash University, Malaysia
Dr. Hong Shen, Professor, University of Adelaide, Australia
Dr. Ying Tan, Professor, Peking University, Beijing, China
Dr. Emad Eldin Mohammed, Associate Professor, Canadian University, Egypt
Dr. Felix J. Garcia Clemente, Associate Professor, University of Murcia, Spain
Dr. Maode Ma, Associate Professor, Nanyang Technical University, Singapore
Dr. X. Z. Gao, Docent, Department of Electrical, Aalto University, Finland
Dr. Lingfeng Wang, Director, University of Wisconsin–Milwaukee, China

Editors

Dr. Raj Kamal, Professor and Ex-Vice-Chancellor, DAVV, Indore, India
Dr. Machel Henshaw, Professor, Loughborough University, UK
Dr. Pramod S. Nair, Professor, Aksum University, Ethiopia

Technical Program Committee

Chair

Shri Sunil Kumar Kopparapu, TCS Research Lab, India

Organizing Chairs

Prof. (Dr.) Suresh Jain, HOD, CSE, Medi-Caps University, Indore
Prof. Dheeraj Rane, Medi-Caps University, Indore

Conference Chair

Dr. Nitika Vats Doohan, Medi-Caps University, Indore
Prof. D. Srinivasa Rao, Medi-Caps University, Indore

Publication Chair

Prof. Prashant Panse, Medi-Caps University, Indore

Chair: Tracks Management

Prof. Vaishali Chourey, Medi-Caps University, Indore

Organizing Committee

General Chair

Prof. Anil Patidar, Medi-Caps University, Indore

Organizing Chair

Prof. Ritesh Joshi, Medi-Caps University, Indore
Prof. Latika Mehrotra, Medi-Caps University, Indore
Dr. Jitendra Choudhary, Medi-Caps University, Indore

Local Arrangements Chair

Prof. Pratiksha Asati, Medi-Caps University, Indore
Prof. Amit Sharma, Medi-Caps University, Indore

Members

Prof. Saurabh Dave, Medi-Caps University, Indore
Prof. Sachin Solanki, Medi-Caps University, Indore
Prof. Dharmendra Mangal, Medi-Caps University, Indore
Prof. Gaurav Sharma, Medi-Caps University, Indore
Prof. Kuber Datt Gautam, Medi-Caps University, Indore
Prof. Pragya Pandey, Medi-Caps University, Indore
Prof. Priyanka Dhasal, Medi-Caps University, Indore
Prof. Ganesh Patidar, Medi-Caps University, Indore
Prof. Sunil Kushwaha, Medi-Caps University, Indore
Prof. Mahendra Patel, Medi-Caps University, Indore
Prof. Khushbu Goyal, Medi-Caps University, Indore
Prof. Pallavi Asrodia, Medi-Caps University, Indore
Prof. Indrajeet Chhabra, Medi-Caps University, Indore
Prof. Swati Tahiliani, Medi-Caps University, Indore
Prof. Rudresh Shah, Medi-Caps University, Indore
Prof. Chandrashekhar Kothari, Medi-Caps University, Indore

Contents

About the Editors

Raj Kamal completed his M.Sc. at the age of 17 and published his first research paper in a UK journal, and his first program, written in FORTRAN was ran at ICT 1904, at the age of 18, and he completed his Ph.D. at the Indian Institute of Technology Delhi at 22. He is currently associated with Medi-Caps University, Indore, India. He pursued his postdoctoral studies (1978–1979) at Uppsala University, Uppsala, Sweden. He has 51 years of research and teaching experience. He has published 142 research papers. He is known all over for his books which include Embedded Systems and Internet of Things from McGraw-Hill. His research interest areas include Internet of things, data analytics, data visualization, app visualization, vehicular technology embedded systems, machine learning and fuzzy logic-based expert systems, optical communication, spectroscopy, and material science.

Michael Henshaw is Professor of Systems Engineering and Head of the Systems Division at Loughborough University, UK, and leads the Engineering Systems of Systems (EsoS) Research Group. His research focuses on integration and management of complex socio-technical systems, with particular emphasis on the challenges of through-life management of systems and capabilities. He graduated in applied physics, and his early research focused on laser–plasma interactions, using computational fluid dynamics to investigate various phenomena in applications such as X-ray lasers. He joined British Aerospace (later BAE Systems) as an aerodynamicist and worked for seventeen years in aeronautical engineering, tackling problems associated with unsteady aerodynamics (computational and experimental) and, later, multidisciplinary integration. He was appointed to a chair in Systems Engineering at Loughborough in 2006 to direct the large (£4M) multi-university, multidisciplinary program sponsored by EPSRC and BAE Systems, NECTISE, which ran from November 2005 to April 2009. He has an international reputation for his work in systems of systems. His research covers the

themes of systems of systems, interoperability, through-life management, network-enabled capability (NEC), autonomous systems, cyber security, and system lifecycles.

Pramod S. Nair is Professor and Head of the Computer Science Department at Medi-Caps University, Indore. He has more than ten years of academic and research experience, together with seven years of industrial experience. He received his B.Tech. from Cochin University of Science and Technology, Kerala, and his M.Tech. in computer science and information technology from MS University, Tamil Nadu. He subsequently completed his Ph.D. in online data mining at IIIT Allahabad. His current research interests are online data mining, business intelligence, utility mining, Web mining, and sentiment analysis. He has published many research articles on these topics in journals and conference proceedings.

Design and Implementation of Fuzzy Expert System for Dengue Diagnosis

Tanmay Kasbe and Ravi Singh Pippal

Abstract Medical diagnosis expert system is one among the best elements of expert system. This paper utilizes fuzzy expert system to raise the diagnosis level of dengue fever and early detection of dengue in patient. Fuzzy expert system is one of the most traditional artificial intelligence techniques to diagnose any disease. This paper uses MATLAB fuzzy logic toolbox to create an expert system, which is based on crisp values, rules, and defuzzification. The design of fuzzy expert system relies on patient symptoms with diagnosing report as input variable. To analyze the proposed system, accuracy, sensitivity, and specificity are measured.

Keywords DF · DHF · DSS · Fuzzy expert system · Medical diagnosis MATLAB · Dengue

1 Introduction

The dengue fever, also known as life-threatening disease, is caused by dengue virus. It is also referred to breakbone fever which is one among the major deadly diseases around the world transmitted by blood-feeding-mosquito, i.e., Aedes aegypti. According to data provided by National Vector Borne Disease Control Programme, Delhi and Maharashtra have the highest mortality rate in India whereas as per WHO, 40% population of world is affected by this disease. A lot of viral infections exist in the world, but dengue fever virus infection causes more illness and death. It comes severe for the people who have weak immune system. An early diagnosis of this disease can help for quick recovery in patient. It can be broadly classified into three categories which are dengue fever (DF), dengue hemorrhagic fever (DHF), and dengue shock syndrome (DSS) [1]. In all the three types, DSS is the most dangerous

T. Kasbe (✉) · R. S. Pippal
RKDF University, Bhopal, India
e-mail: tanmay.kasbe@gmail.com

R. S. Pippal
e-mail: ravesingh@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_1

Table 1 Sign and symptoms table with short investigation

Type	Sign and symptoms	Investigation
Dengue fever (DF)	• High fever • Throat infection • Chills and headache • Breakbone aching • Prostration	• WBC count decrease • AST/ALT increase • Platelets may or may not be decreased
Dengue hemorrhagic fever (DHF)	• High fever • Low bleeding • Headache • Abdominal pain • Restlessness	• WBC count decrease • AST/ALT increase • Platelets decreased • CT abnormal
Dengue shock syndrome (DSS)	• Very high fever • Higher abdominal pain • Bleeding and rashes • Hypothermia • Very high weakness • Consciousness level decreased	• WBC count decrease • AST/ALT increase quickly • Platelets decreased • BP falls • Electrolyte imbalancing

type of dengue fever, and the recovery is even more difficult as compared to DF and DHF. Numerous researchers have done good work in the diagnosis of dengue fever disease using artificial intelligence techniques [1–13] (Table 1).

2 Related Work

In 2015, Pabbi [1] implemented a fuzzy expert system using MATLAB. In this paper, the author gives complete details like symptoms, input variable with their ranges and rules. It uses the local dataset provided by hospitals. Kaur et al. [2] propose a viral infection diagnosis system for chicken pox, swine flu, and dengue. They implement a common fuzzy expert system for all three infections disease and used all the common in nature. The proposed system demonstrates fuzzy expert system for determination of the risk level of the patient, possibly affected by infection. Faisal et al. [3] design an adaptive neuro-fuzzy interface system for the diagnosis of dengue disease. The proposed expert system achieves 86.13% accuracy. In this paper, authors give complete details about ANFIS with its architecture and subtractive clustering algorithm. They use nine input variables and evaluate their system through performance table.

In 2016, Princy and Muruganandam [4] implement a dengue fever disease by using Informatica tool and Oracle database. The patient data is gathered from local hospitals. In this paper, complete detail about implementation and analysis of dataset is given by the author. Sharma et al. [5] develop a decision support system for the

diagnosis of dengue fever symptom by using MATLAB as implementation tool. They work on 34 patients and achieve 91.3% accuracy. Rachmt and Nurhayati [6] present a survey on prediction of dengue hemorrhagic fever (DHF) which is based on artificial neural network (ANN). In this paper, three methods, grid partition, subtractive clustering, and fuzzy C means, are introduced with proper analysis of each method. Saqib et al. [7] introduce a method for diagnosis of dengue fever with death analysis. The authors give complete dengue patient analysis with day wise and values of all symptoms. Dom et al. [8] present method for diagnosis of dengue fever which is the most deadly disease in Malaysia. It consists of the data from 1990 to 2005 and complete analysis of DHF, DSS with death figures. Saikia and Dutta [9] implement a fuzzy expert system to predict a dengue fever at early stage. In this paper, complete implementation method is given by providing details of membership functions with their ranges. It uses centroid techniques for defuzzification.

Shaukat et al. [10] introduce a survey paper of dengue fever using data mining techniques. In this paper, all the data mining techniques like WEKA tool are given and complete comparison of all the techniques is also provided. Pardeshi et al. [11] present a survey paper on dengue of all patients admitted in KEM hospital, Mumbai. In this paper, the authors present complete details of patients with medical reports and provide complete statically data. Dagar et al. [12] introduce a fuzzy expert system for medical diagnosis. In this paper, authors use the MATLAB tool for GUI and they have used common interface for all the medical disease which is diagnosed by FES. A complete development approach of MATLAB is given along with screenshot. Singh et al. [13] develop a fuzzy expert system for diagnosis of dengue fever using MATLAB tool on the basis of content-based filtering. The authors use eight input attributes and one output attribute. Kasbe and Pippal [14] present a useful survey based on dengue fever. This paper consists of analysis of dengue fever affected in globe. The authors provide complete symptoms with diagnosis criteria of dengue fever. Prihatini and Putra [15] develop a fuzzy expert system for tropical infection disease like dengue with accuracy of 91.07%. In this paper, complete detail of system architecture and proposed methodology is given with consultant example. Razak et al. [16] introduce a dengue notification system by using fuzzy logic. They have used the dataset from the local hospital and also taken interview session with the patient. This paper uses the Center of Area (CoA) technique for defuzzification and description of linguistic variables.

3 Proposed Algorithm

This section demonstrates the proposed algorithm for the diagnosis of dengue disease.

INPUT
Input the fuzzy set for **Age**, **WBC Count**, **Platelet Count**, **AST/ALT**, **BP** and **Fever**
OUTPUT
Output the fuzzy set for **Result**
METHOD

Begin
Step 1: Input the crisp values for **Age**, **WBC Count**, **Platelet Count**, **AST/ALT**, **BP** and **Fever**
Step 2: Set the triangular membership function for the fuzzy number with equation.

$$f(x;a,b,c) = \begin{cases} 0, & x \le a \\ \frac{x-a}{b-a}, & a \le x \le b \\ \frac{c-x}{c-b}, & b \le x \le c \\ 0, & c \le x \end{cases}$$

OR

$$f(x;a,b,c) = \max\left(min\left(\frac{x-a}{b-a}, \frac{c-x}{c-b}, 0\right)\right)$$

The parameters a and c locate the "feet" of the triangle and the parameter b locates the peak.
Step 3: Built the fuzzy numbers for **Age**, **WBC Count**, **Platelet Count**, **AST/ALT**, **BP** and **Fever**
Step 3.1: Built the fuzzy number for **Result** for the output set.
Step 4: Fuzzy inference are executed by Mamdani's method.
Step 4.1: Input the rule as {Rule 1, 2…k}
Step 4.2: Matching degree of rule with OR fuzzy disjunction are calculated for fuzzy input set (A11, A12, A13, A21, A22, A23, A31, A32, A33, A41, A42, A43, A51, A52, A53, A61, A62, A63, R1, R2, R3, R4).
Step 5: Defuzzify into the crisp values by **R**

$$R \leftarrow \frac{\sum_{i=1}^{n} Z_i \cdot \mu(Z_i)}{\sum_{i=1}^{i=n} \mu(Z_i)}$$

where Z_i means the weight for $\mu(Z_i)$ and $\mu(Z_i)$ means the number of fuzzy numbers of the output fuzzy variable R.
Step 6: Present the knowledge in the form of human nature language.
End.

4 Implementation

For implementation purpose, this paper uses MATLAB (Matrix Laboratory). It is one the best tools for implementing fuzzy interface system. It uses six input variables and one output variable for identification of disease level. Following are the steps of our proposed fuzzy expert system.

Step: 1

In this step, membership function is created for every input variable as well as output variable. The membership functions used in this system are triangular member function and trapezoidal member function. This member function is used to describe the ranges of variable.

(A) **Age**: Following are the ranges of age

Input filed	Ranges	Fuzzy variables	Fuzzy set value
Age	0–14	A11	Child
	14–44	A12	Young
	44–100	A13	Old

(B) **WBC Count**: Following are the ranges of WBC

Input filed	Ranges	Fuzzy variables	Fuzzy set value
WBC_Count	3–4	A21	Low
	4–11	A22	Normal
	11–16	A23	High

(C) **AST/ALT**: Following are the ranges of AST/ALT

Input filed	Ranges	Fuzzy variables	Fuzzy set value
AST/ALT	1–7	A31	Low
	7–55	A32	Normal
	55–65	A33	High

(D) **Platelet Counts**: Following are the ranges of platelet counts

Input filed	Ranges	Fuzzy variables	Fuzzy set value
Platelet count	5–15	A41	Low
	15–40	A42	Normal
	40–50	A43	High

(E) **Blood Pressure**: Following are the ranges of blood pressure

Input filed	Ranges	Fuzzy variables	Fuzzy set value
BP	60–90	A51	Low
	90–120	A52	Normal
	120–180	A53	High

(F) **Fever**: Following are the ranges of fever

Input filed	Ranges	Fuzzy variables	Fuzzy set value
Fever	94–99	A61	Normal
	99–102	A62	High
	102–108	A63	Very high

(G) **Result**: After creating membership function of all input variable, we need to use membership function for output which is called result. Following are the ranges of result.

Output filed	Ranges	Fuzzy variables	Fuzzy set value
Result	0–3	R1	No dengue
	3–5	R2	DF
	5–8	R3	DHF
	8–10	R4	DSS

Step: 2

Now in this step, the fuzzy rules for all possible outputs are prepared. Result accuracy completely depends upon the strength of rule. If the rules are created with proper combination of input variables, then surely a good accuracy can be achieved. All the rules created in this system are with the help of AND/OR connection. It is related with real-life situation, so before creating the rules we must take care of all the possible situations.

In this implementation part, before creating a rule, we have one special session with doctor, where doctor clearly gives idea about when and what attribute is essential for identifying level of dengue in patient. In the case of dengue, not only high fever is important but with high fever, the level of WBC count, platelet count, and AST is also an important part for DSS.

Step: 3: Rule Viewer

By the help of rule viewer, one can check the disease condition by inputting the patient's information given as input parameter. Following is the screenshot of the implemented rule viewer (Fig. 1).

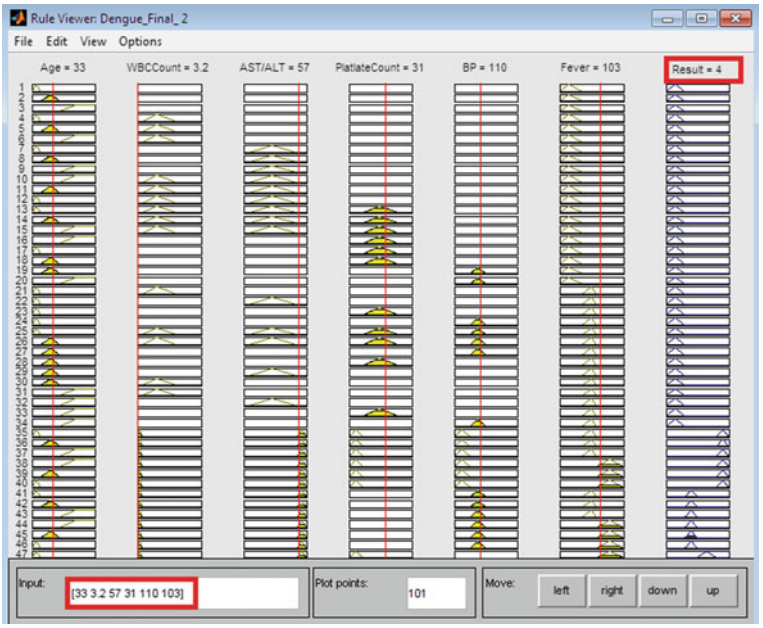


Fig. 1 Rule viewer

5 Result and Discussion

In this study, different levels of dengue fever are categorized using fuzzy logic toolbox and by the help of hospital dataset. This proposed method used the database provided by SAIMS hospital, Indore. All the patient data belongs to year 2017. The proposed fuzzy-based expert system achieves 92.8% accuracy and 94.25% sensitivity. The accuracy and sensitivity are measured by the help of confusion matrix as follows (Table 2).

True Positive (TP) = In this case, we have predicated dengue and they suffer from dengue.
True Negative (TN) = In this case, we have predicated a healthy person and they do not suffer from dengue.
False Negative (FN) = In this case, we have predicated a healthy person and they suffer from dengue.

Table 2 Confusion matrix

Total = 125	Predicated value: No	Predicated value: Yes
Actual value: No	TN = 52	FP = 0
Actual value: Yes	FN = 4	TP = 69

Age	WBC count	AST/ALT	Platelet Counts	BP	Fever	Our Result	Actual Result
13	7	29	20	105	97.5	No Dengue	No Dengue
37	8.75	39	50.5	101	96	No Dengue	No Dengue
52	9	47	31.41	110	101	No Dengue	No Dengue
23	7.74	53	37.33	108	98	No Dengue	No Dengue
57	8	41	22	104	102	No Dengue	DF
34	5.74	11	17.24	100	98.6	No Dengue	No Dengue
18	4.95	16	26	108	97	No Dengue	No Dengue
39	4	41	29.35	110	98.7	No Dengue	No Dengue
39	3.3	59	14.45	105	98.7	No Dengue	No Dengue
33	3.2	57	31	110	103	DF	DF
28	3.9	61	19	108	106	DF	DF
19	3	63	27	102	102.5	DF	DF
24	3	59	35	116	103.5	DF	DF
25	3.6	58	38	115	104	DF	DF
71	3.1	61.16	13.33	107	106.3	DHF	DHF
36	3.3	63.25	11	119	105	DHF	DHF
43	4	60	7.14	117	104	DHF	DHF
29	3.8	62.5	6.66	101	105.5	DHF	DHF
53	3.5	64	7	61	104.7	DSS	DSS
41	3	61	11	84	106.5	DSS	DSS
27	3.2	58	12.35	83	106	DSS	DSS
71	3.6	58.74	11	79	105.5	DSS	DSS
31	3.7	64	9.75	75	105	DSS	DSS
21	3.6	58	13	69	104	DSS	DSS
65	3.9	56.65	6.5	65	105.5	DSS	DSS

Fig. 2 Partial sample result sheet

False Positive (FP) = In this case, we have predicated dengue and they do not suffer from dengue (Fig. 2).

6 Conclusion

Today, dengue is one of the most common life-threatening diseases not only in India but all over the world. The aim of this paper is to provide an expert system which is used to diagnose the level of dengue fever early. The proposed expert system can be utilized by any individual as it is very user-friendly having simple GUI. It is based on fuzzy logic which achieves 92.8% accuracy so far. The proposed system has better performance in comparison with other existing research work.

In future research, the proposed work can be extended in the same field so that it can be utilized for other severe diseases.

References

1. V. Pabbi, Fuzzy expert system for medical diagnosis. Int. J. Sci. Res. Publ. **5**(1), 1–7 (2015)
2. R. Kaur, S. Kaur, V. Rehani, Fuzzy based automated system for predicting viral infections. Int. J. Innovat. Res. Multidiscip. Field **2**(11), 426–434 (2016)

3. T. Faisal, M.N. Taib, F. Ibrahim, Adaptive Neuro fuzzy-interface system for daignosis risk in dengue patients. Expert Syst. Appl. **39**(4), 4483–4493 (2012)
4. S.S.L. Princy, A. Muruganandam, An implementation of dengue fever disease spread using informatica tool with special reference to Dharampuri district. Int. J. Innovat. Res. Comput. Commun. Eng. **4**(9), 16215–16222 (2016)
5. P. Sharma, D. Singh, M.K. Bandil, N. Mishra, Decision support system for Malaria and Dengue disease diagnosis. Int. J. Informat. Comput. Technol. **3**(7), 633–640 (2013)
6. B. Rachmt, O.D. Nurhayati, Prediction the number of patients at Dengue H fever cases using adaptive neural Fuzzy interface system. Int. J. Innovat. Res. Advanc. Eng. **3**(4), 23–28 (2016)
7. M.A.N. Saqib, I. Rafique, S. Bashir, A.A. Salam, *A Retrospective Analysis of Dengue Fever Case Management and Frequency of Co-Morbidities Associated with Deaths*, BMC Research Notes, pp. 1–5 (2014)
8. N.C. Dom, A.H. Ahmed, R. Adawiyah, R. Ismail, Spatial mapping of temporal risk characteristics of Dengue cases in Subang Jaya, in *Proceedings of the 2010 IEEE International Conference on Science and Social Research (CSSR 2010)* (Kuala Lampur, Malaysia, 2010), pp. 361–366
9. D. Saikia, J.C. Dutta, Early diagnosis of Dengue disease using Fuzzy interface system, in *Proceedings of the 2016 IEEE International Conference on Microelectronics, Computing and Communication (MicroCom 20016)* (Durgapur, India, 2016)
10. K. Shaukat, N. Masood, S. Mahreen, U. Azmeen, Dengue fever prediction–a data mining problem. Data Mining Genom. Proteom. **6**(3), 1–5 (2015)
11. A. Pardeshi, R. Shinde, A. Jagtap, R. Kembhavi, M. Giri, S. Kavathekar, Retrospective cross-sectional study of Dengue cases in IPD with reference to treatment- monitoring & outcome in KEM hospital. Mumbai Am. J. Epidemiol Infect. disease **2**(4), 97–100 (2014)
12. P. Dagar, A. Jatain, D. Gaur, Medical diagnosis system using Fuzzy logic, in *Proceedings of the 2015 IEEE International Conference on Computing, Communication and Automation (ICCCA 2015)*, Noida, India, pp. 193–197 (2015)
13. S. Singh, A. Singh, M.Singh Samson, Recommender system for Dengue using Fuzzy logic. Int. J. Comput. Eng. Technol. **7**(2), 44–52 (2016)
14. T. Kasbe, R.S. Pippal, Dengue fever: state-of-the-art symptoms and diagnosis. Int. J. Comput. Sci. Eng. **4**(6), 1–5 (2016)
15. P.M. Prihatini, I.K.G.D. Putra, Fuzzy knowledge based system with uncertainty for tropical infectious disease daignosis. Int. J. Comput. Sci. **9**(4), 157–163 (2012)
16. T.R.B. Razak, M.H. Ramli, R.A. Wahab, Dengue notification system using Fuzzy logic, in *Proceedings of the 2010 IEEE International Conference on Computer, Control, Informatics and its Application (IC3INA 2013)* (Jakarta, Indonesia, 2013), pp. 231–235

Reduced Order Modeling of Linear Time-Invariant Systems Using Soft Computing Technique

Shilpi Lavania and Deepak Nagaria

Abstract Nature has the key to solve every single problem that exists. This paper strives to propose a method which is based on flower pollination algorithm to acquire a stable reduced order model (ROM) of a higher order linear time-invariant (LTI) system. The suggested method utilizes Padé-based moment matching technique to deduce denominator approximants for the reduced order system, whereas the reduced order approximants of numerator polynomial are obtained using flower pollination (FP) algorithm. To prove the viability of the suggested method, numerical examples are solved considering single-input single-output (SISO) systems. A comparison has been drawn between suggested method and existing methods which are available in the current literature and is presented in this paper. This comparison is founded on a performance index which is known as integral square error (ISE).

Keywords Moment matching · Single-input single-output system (SISO) Integral square error (ISE) · Reduced order model · Padé approximation Flower pollination (FP) algorithm

1 Introduction

Complexity has always been a major concern for the researchers while analyzing complex and larger scale systems. A large and complex system takes ample time to be modeled mathematically. At later stages, it requires large simulation time and implementation time. This has escalated the requirement for model order reduction (MOR). A grand literature exists in the area of model order reduction (MOR). Davison [1] instigated first MOR method to diminish the order of a linear higher order system

S. Lavania (✉)
Dr. A.P.J. Abdul Kalam Technical University, Lucknow, India
e-mail: shilpilavania31989@gmail.com

D. Nagaria
Bundelkhand Institute of Engineering & Technology, Jhansi, India
e-mail: deepaknagariaaktu@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_2

(HOS). In 1967, Chidambra [2, 3] suggested a refinement to the Davison technique. Since its development, myriad of techniques have been proposed in the field of MOR. All the proposed methods are classified in time domain and frequency domain. In each category, umpteen techniques have been suggested. Further, diverse methods, i.e., modal method, optimal solution, and singular method, etc. have been proposed in time domain, whereas Padé approximation, Routh approximation, Continued fraction, etc., in frequency domain [4].

The area of model order reduction has advanced with time. With the development of metaheuristic nature-based algorithms, the domain of model order reduction has constantly taken advantage of these metaheuristic algorithms. This evolution started when Buttleman and Lohmann [5] proposed model order reduction technique based on genetic algorithm. Afterward, many variants of this evolutionary method for the reduction of linear time systems (LTI) and discrete time systems suggested [6–8]. This evolution gave an extensive range of research in MOR and optimization of error using particle swarm optimization [9, 10]. After this, Big bang big crunch optimization [11] and recently Cuckoo search algorithm [4] have been used for model order reduction. But the area of MOR is not restricted here, and it holds various modern researches to be investigated. Padé approximation is one of the foremost robust tools for developing reduced order approximants of higher order system. It was first suggested by Padé [12–16]. Xin She-yang in the year of 2012 proposed flower pollination algorithm. It mimics flower pollination process that exists in nature. Posterior, a multi-objective FP algorithm was proposed, which used a method founded on weighted sum with random weights, was also capable of finding the Pareto fronts for a set of test functions successfully [17, 18].

This paper strives to propose a model order reduction technique which is a composition of a modified Padé approximation and metaheuristic flower pollination (FP) algorithm. Padé-based moment matching technique, i.e., matching of both time moments and Markov parameters, is utilized to acquire reduced order approximants for denominator polynomial. Whereas, the reduced order approximants for numerator polynomial are derived using metaheuristic flower pollination (FP) algorithm. A performance comparison with existing methods to prove the worthiness of the suggested method is also presented in this paper. The estimation of performance is done on the basis of a performance index which is known as integral square error (ISE). Numerical example is solved in support of the literature suggested in this correspondence.

2 Problem Statement

Consider an nth order SISO system:

$$G_n(s) = \frac{a_1 s^{n-1} + a_2 s^{n-2} + \cdots + a_n}{s^n + b_1 s^{n-1} + \cdots + b_n} \tag{1}$$

The technique is intended to acquire an approximated rth order model

$$G_r(s) = \frac{a_1^{'}s^{r-1} + a_2^{'}s^{r-2} + \cdots + a_r^{'}}{s^r + b_1^{'}s^{r-1} + \cdots + b_r^{'}} \tag{2}$$

where $r < n$.

3 Model Order Reduction Procedure

A. ***Modified Padé Approximation***

This section explains the entire procedure to obtain reduced order model for a HOS using proposed method. Consider a nth order system as described in Eq. (1) [17, 18]:

On expanding (1) about $s = 0$ and $s = \infty$:

$$= t_0 + t_1 s + t_2 s^2 + \ldots \tag{3}$$

(Expansion about $s = 0$)

$$= M_1 s^{-1} + M_2 s^{-2} + \ldots \tag{4}$$

(Expansion about $s = \infty$)

The aspiration is already discussed in the previous section. On performing the power series, expansion of (2) yields:

$$= t_0^{'} + t_1^{'} s + t_2^{'} s^2 + \ldots \tag{5}$$

(Expansion about $s = 0$)

$$= M_1^{'} s^{-1} + M_2^{'} s^{-2} + \ldots \tag{6}$$

(Expansion about $s = \infty$)

In order to validate that the following conditions hold true [17]:

$$t_1^{'} = \frac{a_r^{'}}{b_r^{'}} \quad i = 1 \tag{7}$$

$$t_i^{'} = a_{r+1-i}^{'} + \sum_{j}^{i-1} (t_i^{'} b_{r+j-i}^{'}) b_r^{'-1} \quad i = 2, 3 \ldots \tag{8}$$

and $a_i^{'} = 0$ for $i \leq 0; b_0^{'} = 1; b_i^{'} = 0$ for $i \leq -1$

$$\begin{aligned} M_i^{'} &= a_i^{'} \qquad\qquad i = 1 \\ M_i^{'} &= a_i^{'} - \sum_{j=1}^{i-1} M_i^{'} b_{i-j}^{'} \; i = 2, 3, 4, \ldots \end{aligned} \tag{9}$$

The acquired reduced order model must satisfy following equations:

$$\begin{aligned} t_0^{'} &= t_0 \\ a_{r+1}^{'} &= \sum_{j=1}^{i} t_j b_{i-j}^{'}, \; i \in \{1, 2, \ldots \delta\} \\ M_i^{'} &= M_i \\ a_i^{'} &= \sum_{j=1}^{i} M_j b_{i-j}^{'}, \; i \in \{1, 2, \ldots \gamma\} \end{aligned} \tag{10}$$

where $\delta + \gamma = 2r$.

By using above Eqs. (3)–(10), approximants of reduced order models can be achieved.

B. ***Flower Pollination algorithm***

FP is rooted on the endurance of the fittest along with the optimal proliferation of the plants concerning the numbers as well as fitness, is actually an optimization procedure for plant species, in which all of the listed factors and processes of FP interact to gain effective reproduction of plants. This was the inspiration behind the development of flower pollination optimization algorithm [17, 18].

Pollination can be attained by either self-pollination or cross-pollination. In allogamy which is also known as cross-pollination, conception occurs by pollen of one flower of a different plant. Whereas in self-pollination, conception of one flower from the pollen of same flower is the cause of reproduction. Also if conception occurs by different flower of same plant occurs, it falls under the category of self-pollination [17, 18]. Followings are the characteristics/rules for pollination process, behavior of pollinators, and flower constancy:

1. During allogmy and biotic process pollinators which are carrying pollens performs Lévy flights. This is also termed as global pollination.
2. Flower constancy is regarded as the probability of proliferation which is proportional to the common characteristics of two flowers involved.
3. Self-pollination and abiotic pollination fall under local pollination category.
4. In order to control global and local pollination process, a switch probability is defined as $p \in [0, 1]$.

Composition of rule 1 and flower persistence is regarded as [17, 18]:

$$x_i^{k+1} = x_i^k + L\left(m^* - x_i^k\right)$$

where m^* represents the most fit and recent best solution. x_i^k is the pollen,x_i is the solution vector at k iterations, and L is the pollination strength.

Lévy distribution is represented as [17, 18]:

$$L \sim \frac{\lambda\Gamma(\lambda)\sin(\lambda/2)}{\pi}.\frac{1}{s^{1+\lambda}}$$

Here, $\Gamma(\lambda)$ is standard gamma function. Its dispersal is valid for step size

Rule 2, i.e., local pollination and flower constancy may be regarded as:

$$x_i^{k+1} = x_i^k + \varepsilon\left(x_t^k - x_u^k\right)$$

x_t^k and x_u^k are the pollens of different flowers those belong to the same plant.

4 Numerical Examples

Example 1 Let the higher order system to be reduced is represented as [19, 20] (Fig. 1 and Table 1):

$$G(s) = \frac{2s^5 + 3s^4 + 16s^3 + 20s^2 + 8s + 1}{2s^6 + 33.6s^5 + 155.94s^4 + 209.46s^3 + 102.42s^2 + 18.3s + 1}$$

Let the reduced order model form as [14]:

$$G_r(s) = \frac{p_1^{'}s + p_2^{'}}{s^2 + q_1^{'}s + q_2^{'}} \tag{11}$$

Required reduced order denominator using Padé approximation is obtained as:

$$D^r(s) = s^2 + 16.83s + 1.53 \tag{12}$$

The reduced order approximants of numerator are obtained using flower pollination (FP) algorithm as given in the prior section. The reduced order approximants of numerator polynomial are obtained as:

$$N^r = 0.672s + 1.5 \tag{13}$$

Thus, the reduced order model is obtained as:

$$G_2(s) = \frac{0.672s + 1.5}{s^2 + 16.83s + 1.53} \tag{14}$$

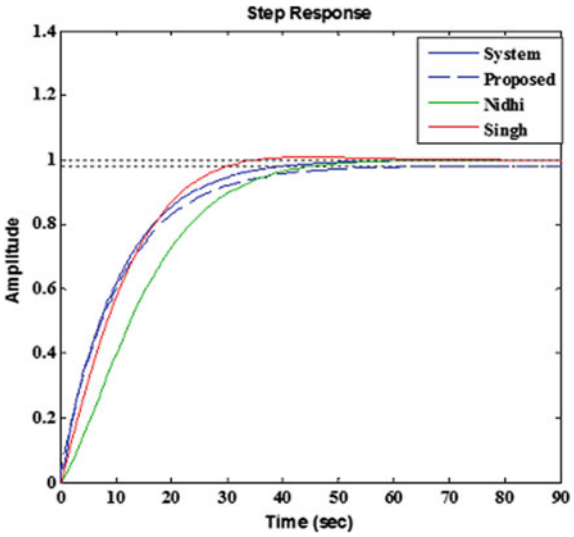


Fig. 1 Comparison of time response of Example 1 using proposed method and existing methods

Table 1 Comparison of existing methods

Method	Model	ISE
Original	$G(s) = \frac{2s^5+3s^4+16s^3+20s^2+8s+1}{2s^6+33.6s^5+155.94s^4+209.46s^3+102.42s^2+18.3s+1}$	–
Proposed	$G_2(s) = \frac{0.672s+1.5}{s^2+16.83s+1.53}$	4.865 e^{-004}
Singh [19]	$G_2(s) = \frac{5.9979s+1}{87.97s^2+15.96s+1}$	0.0650
Mahmoud [20]	$G_2(s) = \frac{0.0227s+0.0131}{s^2+0.22s+0.0131}$	8.7854

5 Conclusion

A novel method for model order reduction which relies upon flower pollination (FP) algorithm is provided in this piece of work. In addition to FP algorithm, Padé-based approximation which relies on giving equal weightage to Markov's parameters, as well as the time moments, is utilized to obtain diminished approximants of denominator polynomial. The time response curve of reduced model shown in the result section interprets that the proposed method provides excellent estimation of reduced model for the original higher order system. Integral square error (ISE) is utilized as a performance index. Low value of error is always desirable. Similarly, lower the value of ISE, more approximated will be the response for higher order system. The technique proposed in this paper has lower value of ISE than other existing methods. Stability is an important factor and a major concern in model order reduction. Reduced order model obtained using the proposed method in the given examples has appeared to be stable. Hence, it can be stated that the proposed method has potential and can be considered as an effective method for model order reduction.

References

1. E. Davison, A method for simplifying linear dynamic systems. IEEE Trans. Automat. Contr. **11**(1), 93–101 (1966)
2. M. Chidambara, Comments on the simplification of linear systems. IEEE Trans. Automat. Contr. **18**(1), 76–77 (1973)
3. G. Marchesini, S. Karni, N. Ahmed, On obtaining transfer functions from gain-function derivatives. IEEE Trans. Automat. Contr. **12**(6), 800 (1967)
4. A. Sikander, R. Prasad, A novel order reduction method using cuckoo search algorithm. IETE J. Res. **61**(2), 83–90 (2015)
5. B. Buttelmann, M. Lohmann, Model simplification and order reduction of non-linear systems with genetic algorithms, in *IMACS Symposium on Mathematical Modelling*, pp. 777–781 (2000)
6. S. Panda, S.K. Tomar, R. Prasad, C. Ardil, Reduction of linear time-invariant systems using Routh-approximation and PSO. Int. J. Electr. Comput. Energy Electron. Commun. Eng. **3**(9), 1769–1776 (2009)
7. Z.S. Abo-Hammour, O.M.K. Alsmadi, A.M. Al-Smadi, Frequency-based model order reduction via genetic algorithm approach, in *International Workshop on Systems, Signal Processing and their Applications, WOSSPA*, pp. 91–94 (2011)
8. S. Mukherjee, R.C. Mittal, Order reduction of linear discrete systems using a genetic algorithm, Appl. Math. Model. **29**(6), 565–578 (2005)
9. S. Lavania, D. Nagaria, Evolutionary approach for model order reduction. Perspect. Sci. **8**, 361–363 (2016)
10. O.P. Bharti, R.K. Saket, S.K. Nagar, Controller design of DFIG based wind turbine by using evolutionary soft computational techniques. Eng. Technol. Appl. Sci. Res. **7**(3), 1732–1736 (2017)
11. S.R. Desai, R. Prasad, A new approach to order reduction using stability equation and big bang big crunch optimization. Syst. Sci. Control Eng. **1**(1), 20–27 (2013)
12. Y. Shamash, Stable reduced-order models using Padé-type approximations. IEEE Trans. Automat. Contr. **19**(5), 615–616 (1974)
13. Y. Shamash, Model reduction using the Routh stability criterion and the Padé approximation technique. Int. J. Control **21**(3), 475–484 (1975)
14. A. Pati, A. Kumar, D. Chandra, Suboptimal control using model order reduction, Chinese J. Eng. 1–5 (2014)
15. S. Lavania, D. Nagaria, Pade approximation based moment matching technique for model order reduction, in *2015 International Conference on Computer, Communication and Control (IC4)*, pp. 1–4 (2015)
16. S. Lavania, D. Nagaria, Eigen spectrum based moment matching technique for model order reduction, in *2015 39th National Systems Conference (NSC)*, pp. 1–5 (2015)
17. X.-S. Yang, Flower pollination algorithm for global optimization, in *Lecture Notes in Computer Science*, pp. 240–249 (2012)
18. X.-S. Yang, M. Karamanoglu, X. He, Multi-objective flower algorithm for optimization. Proc. Comput. Sci. **18**, 861–868 (2013)
19. N. Singh, *Reduced Order Modelling and Controller Design, (Unpublished Thesis)* (Indian Institute of Technology, Roorkee, 2008)
20. M.S. Mahmoud, M.G. Singh, *Large Scale System Modeling* (Pergamon Press International Series on System and Control, First, 1981)
21. G. Parmer, R. Prasad, S. Mukherjee, Order reduction of linear dynamic systems using stability equation method and GA. Int. J. Electr. Comput. Energ. Electron. Commun. Eng. **1**(2), 236–242 (2007)
22. S.R. Desai, R. Prasad, A new approach to order reduction using stability equation and big bang big crunch optimization. Syst. Sci. Control Eng. **1**(1), 20–27 (2013)

Automated Classification of Cancerous Brain Tumours Using Haarlet Transform and Probabilistic Neural Network

Manmet Kaur and Balwant Prajapat

Abstract The classification of cancerous brain tumours is a complex task for physicians. Hence, automated machine learning techniques are being explored to do the same. This paper proposes an automated technique for cancerous brain tumour classification using Haarlet transform and probabilistic neural network. Data pre-processing uses threshold-based segmentation and binarization in addition to Haarlet transform. Feature extraction for the training data set yields twelve features for each of the training images which is in turn used to train a probabilistic neural network. It has been shown that the proposed system attains 96.3% classification accuracy. A comparative analysis reveals that the proposed system attains higher accuracy of classification compared to contemporary approaches.

Keywords Brain tumour classification · Haarlet transform · Segmentation
Principal component analysis · Probabilistic neural network

1 Introduction

Brain cancer is a lethal medical disorder which is extremely difficult to detect and classify, especially at early stages of diagnosis. The diseased are thereby dependent on the expertise of the physician. Off late, artificial intelligence-(AI) based systems are being explored for automated classification of brain tumour classification. Artificial neural networks are used to implement AI-based systems practically [1]. This paper presents a methodology to implement automated brain tumour classification using image pre-processing and probabilistic neural network (PNN) which is an ubiquitous data classification mechanism based on Bayes' theorem of conditional probability.

M. Kaur (✉) · B. Prajapat
Department of CSE, VITM, Indore, India
e-mail: manmeet1201@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_3

2 Data Pre-processing

The data utilized in this paper is magnetic resonance imaging images (MRIs) which have been classified into three categories for training the designed PNN. The pre-defined training categories are:

1. Normal
2. Benign (Non-cancerous)
3. Malignant (Cancerous).

Since images undergo several degradations during capturing, storage and transferring, hence de-noising the image using some image filtering technique becomes mandatory. Moreover, the tumour affected section (if any) needs to be separated from the rest of the image area [2]. This is done through threshold-based segmentation in this case. Since wavelet transform generates a large set of discrete wavelet transform coefficients, hence principal component analysis is used to remove redundancies of the DWT data and convert the data into a set of linearly uncorrelated data.

3 Feature Extraction

After the data pre-processing is over, important parameters or features of the images are computed termed as feature extraction. In this paper, twelve features are computed to train and subsequently test the designed PNN. The features computed are: (1) contrast (2) correlation (3) energy (4) homogeneity (5) mean (6) standard deviation (7) entropy (8) variance (9) root mean square (RMS) value (10) smoothness (11) kurtosis (12) skewness.

Although several features are seen to be instrumental in image classification, it is found that the above features affect the results the most [3].

4 Design of Probabilistic Neural Network

One of the most effective tools for multi-dimensional data classification is the probabilistic neural network. The probabilistic neural network is based on the Bayes' theorem of conditional probability given by:

$$P(A/B) = \frac{P(B/A) \cdot P(A)}{P(B)} \tag{1}$$

The several merits of PNN are underlined underneath [4]:

- It has the capability to map any number of inputs to any number of classifications.

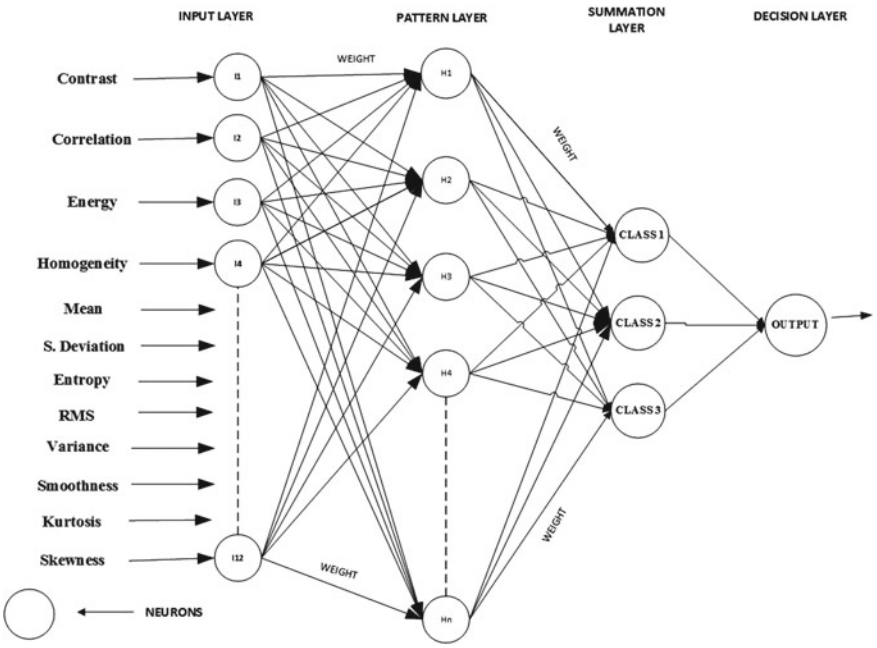


Fig. 1 Conceptual internal structure of proposed PNN

- It doesn't have a back propagation unlike most of the neural networks, making it a fast in grasping and learning.
- Once the neural network is trained, it doesn't require it to be retrained to add or remove a sample of data.
- Great amount of computational space is saved in its case.

The designed PNN in this case is shown in Fig. 1.

5 Proposed Algorithm

The proposed algorithm is a sequential implementation of the following steps:

(a) Categorize data set into training and testing data with a sub-classification of normal, benign and malignant images.
(b) Apply Binarization: This process removes small object patches from the images that may result in obstruction of the actual tumour detection from the MRI image.
(c) Apply Threshold-Based Segmentation: This will separate out the region affected by the tumour. The segmentation can be given by:

$$Y(I) = k1; mod(I) > T, elseY(I) = k2 \tag{2}$$

Here T refers to the threshold of the Segmentation. This may vary from system to system.

(d) Apply a subset of the discrete wavelet transform, which in this case is the Haarlet. The mathematical formulation for the same is given by:

The mathematical expression of the wavelet transform can be given as under:

$$C(S, P) = \int_{-\infty}^{\infty} \boldsymbol{f}(\mathbf{t})((\boldsymbol{S}, \boldsymbol{P}, \boldsymbol{t})) \tag{3}$$

Here S is taken for scaling, P refers to position, t denotes time shifts and C is the continuous wavelet transform (CWT). The main demerit of the CWT can be attributed to fact that is it consists if a huge amount of data. The sampled version of the CWT is the discrete wavelet transform (DWT). The DWT is a down-sampled form of the CWT and its characteristic nature is to smoothen out sudden fluctuations that can occur due to base functions abrupt changes and also because of down sampling.

The scaling function can be expressed as:

$$\mathrm{W}\,\Phi\,(\mathrm{Jo}, \mathrm{K}) = \frac{1}{\sqrt{M}} \sum_{n} S(n) \cdot \Phi(n)_j \tag{4}$$

The wavelet function can be expressed mathematically as:

$$\mathrm{W}\,\psi\,(\mathrm{j}, \mathrm{k}) = \frac{1}{\sqrt{M}} \sum_{n} S(n) \cdot \psi(n) \tag{5}$$

Here, W ψ (j, k) is the normalizing factor.

(e) Apply principal component analysis mathematically given by:

$$T_L = X W_{\mathrm{L}} \tag{6}$$

Here, X is the dataset to be mapped with dimensional reduction, W_L is the matrix of L loading vectors and T_L represents the uncorrelated transform.

(f) Train the designed PNN with 70% of the data and test the network using 30% of the data. Let d stand for the number of features of the input instance x, σ is the smoothing parameter, and x_{ij} is a training instance corresponding to category c. The summation layer neurons execute by computing the maximum likelihood of pattern x that is classified into c with the process of summarizing and averaging the output of every neuron that associates with the same class

$$\prod_{i}^{c} = \frac{1}{\left(2\prod\right)^{n/2}\sigma^{n}} \exp\left[\frac{-\left(x - x_{ij}\right)^{T}\left(x - x_{ij}\right)}{2\sigma^{2}}\right] \tag{7}$$

In the equation shown N_i refers to the total number of samples in class c. Supposing if the a priori probabilities for each class is similar, and also the measure of losses related to making a wrong decision for each of the classes is also the same, the pattern x is then classified by the decision layer according to the Bayes' Decision Rule that relies on the output of all the summation layer neurons that gives

$$C(x) = \text{argument-max}\left\{p_i(x)\right\}, \quad i = 1, 2, \ldots c \tag{8}$$

In the given equation where C(x) infers the estimated class of the pattern x and m is the total number of classes in the training samples. If the a priori probabilities for each class are different and vary along with the losses connected with making an incorrect decision for each class are different, the output of all the summation layer neurons can be given as

$$C(x) = \text{argmax}\left\{p_i(x)\, \text{cost}_i(x)\text{apro}_i(x)\right\}, \quad i = 1, 2, \ldots c \tag{9}$$

Where $\text{cost}_i(x)$ is, the cost associated with misclassifying the input vector and $\text{apro}_i(x)$. Therefore, PNN behaves like an efficient classifier for carrying out data classification into several categories.

6 Results

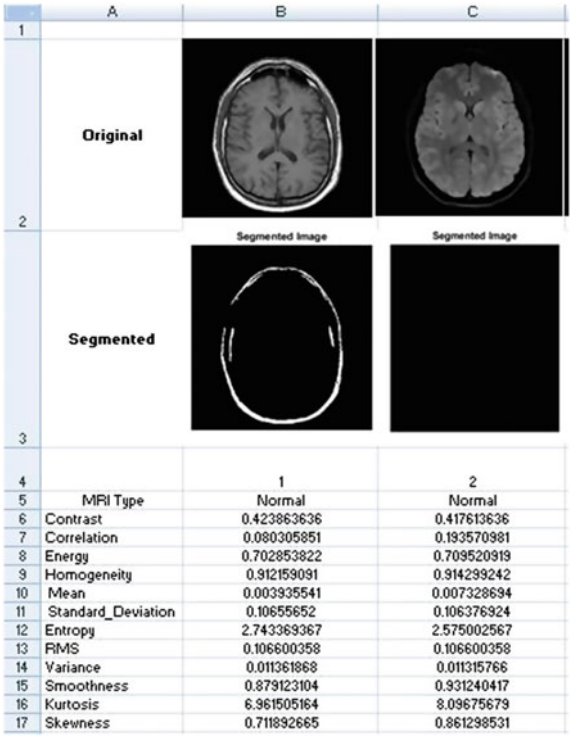
The designed PNN has 12 inputs corresponding to the 12 features extracted. The number of input neurons has been taken as 24, although it may be varied. More number of neurons can enhance the computational capability of the system but would increase system complexity. The accuracy for the system is computed as (Figs. 2 and 3):

Accuracy (Ac): It is mathematically defined as:

$$\frac{TP + TN}{TP + TN + FP + FN} \tag{10}$$

Here TP, TN, FP and FN represent true positive, true negative, false positive and false negative, respectively. In this case, the accuracy was found to be Ac = 17 + 9/17 + 9 + 0 + 1 = 0.9629 = 97% (approximately) for the used dataset which contained 9 normal, benign and malignant cases, respectively in the testing data set.

A comparative analysis with the different contemporary techniques given in "**Brain tumors detection and segmentation in MR images: Gabor wavelet versus statistical features, by Nooshin Nabizadeh and Miroslav Kubat, Elsevier 2015**" [5–8], in which the approach given by Gabor Wavelets attains a maximum accuracy of 95.3%, while the proposed approach attains an accuracy of 97%.



		1	2
5	MRI Type	Normal	Normal
6	Contrast	0.423863636	0.417613636
7	Correlation	0.080305851	0.193570981
8	Energy	0.702853822	0.709520919
9	Homogeneity	0.912159091	0.914299242
10	Mean	0.003935541	0.007328694
11	Standard_Deviation	0.10655652	0.106376924
12	Entropy	2.743369367	2.575002567
13	RMS	0.106600358	0.106600358
14	Variance	0.011361868	0.011315766
15	Smoothness	0.879123104	0.931240417
16	Kurtosis	6.961505164	8.09675679
17	Skewness	0.711892665	0.861298531

Fig. 2 Data Pre-processing and subsequent feature extraction

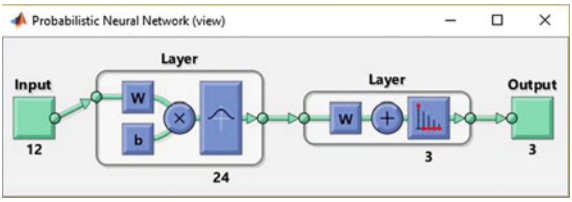


Fig. 3 Designed probabilistic neural network

7 Conclusion

It can be concluded from the above discussions and mathematical formulations that the proposed system achieves almost 97% classification accuracy, which is higher than contemporary approaches [5–8]. The system complexity is moderate due to the feed-forward nature of the PNN. Moreover, rigorous data pre-processing results in better training of the PNN. Further researchers can explore other pre-processing tools such as the MODWT and hybrid neural network designs.

References

1. A. Hasan, D. Dr. Jumaa, Dr. S. Bashk, Brain image classification based on discrete wavelet transform and probabilistic neural network. Int. J. Sci. Eng. Res. **7**(5) (2016)
2. N. Nabizadeh, M. Kubat, Brain Tumors Detection and Segmentation in MR Images: Gabor wavelet versus Statistical Features, Elsevier, Science Direct (2015)
3. L. Popescu, I. Sasu, Feature extraction, feature selection and machine learning for image classification: a case study, in *International Conference on Optimization of Electrical and Electronic Equipment (OPTIM)*. IEEE (2014)
4. B. Menzes, The multimodal brain tumor image segmentation benchmark (BRATS). IEEE Trans. Med. Imag. **34**(10)
5. S. Gaikwad, M. Joshi, Brain tumor classification using principal component analysis and probabilistic neural network. Int. J. Comput. Appl. **120**(3), 0975–8887 (2015)
6. K. Mala, V. Alagappan, Neural network based texture analysis of CT images for fatty and cirrhosis liver classification. Appl. Soft Comput. **32**, 80–86 (2015)
7. K. Dr. Kulhalli, V. Kolge, Primary level classification of brain tumor using PCA and PNN. Int. J. Recent Innovat. Trends Comput. Commun. **2**
8. D. Sridhar, M. Krishna M, Brain tumor classification using discrete cosine transform and probabilistic neural network, in *International Conference on Signal Processing, Image*

Solar Energy Prediction Using Backpropagation in Artificial Neural Networks

Abhishek Kumar and Ravi Khatri

Abstract This paper presents a mechanism to predict solar irradiation energy using artificial neural networks. Solar energy has been at the forefront of renewable sources of energy since a long time now and has found applications in diverse fields. To develop a system completely dependent on solar energy, one needs a good estimate of the amount of solar irradiation that can be obtained. This is very challenging though due to the dependence of solar irradiation on several parameters such as seasonal changes, cloud cover, moisture. Statistical techniques are prone to errors in prediction. In this paper, backpropagation is used to predict solar irradiation and an accuracy of 97.81% is achieved. An overall regression of 0.95 is obtained.

Keywords Solar irradiation · Artificial neural networks · Backpropagation Levengerg−Marquardt algorithm · Mean absolute error (MAE)

1 Introduction

With every passing day, the dependence on fossil fuels is decreasing and renewable sources of energy are being investigated. Solar energy has led the way in usage of non-conventional sources of energy. Yet, a large percentage of the power sector still relies on non-renewable sources. Earlier, only low power equipments and loads had solar energy as the driving source, but off-late, heavy industries and commercial applications are looking forward to solar energy for their power requirements. The major challenge in solar-powered heavy power-requirement industries is the fact that solar energy is not constant and fluctuates abruptly [1]. The dependence of solar energy on several parameters such as cloud cover, wind speed, topography of the area, time of the day makes it relatively unpredictable in nature as compared to conventional sources of energy. Still, large investments are being planned to establish solar power plants and solar-driven industries and a plethora of other applications. This

A. Kumar (✉) · R. Khatri
Department of CSE, VITM, Indore, India
e-mail: abhivitm15@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_4

needs a fair amount of estimate of the solar irradiation well in advance. Hence, solar irradiation forecasting has become an extremely active area of research. Previously, existing statistical models were prone to errors which meant less accurate prediction. With the advances in artificial intelligence, prediction problems came under the purview of artificial intelligence. Practically, implementing AI-based system using designed artificial neural networks (ANNs) could be used for a variety of prediction problems. The ones pertaining to solar irradiation forecasting are as follows:

1. Short-Term Forecasting: This is a forecasting model which could predict irradiation for only a few hours. It is used for daily consumption and declaration of power. This decides the cost of the power in the market.
2. Mid-Term Forecasting: Its done to predict weekly, monthly, or seasonal trends. It is necessary for regular maintenance.
3. Long-Term Forecasting: This is done in order to facilitate plans regarding the establishment of new power plants and related projects.

2 Introduction to Artificial Neural Network and Prediction Problems

ANNs have an extremely effective learning capability of learning from trends in the data and fast computing capabilities due to the parallel structure (Fig. 1).

The fact that the neural network can accept and process data in parallel ensures high speed of operation of the neural network. The learning and adapting capability

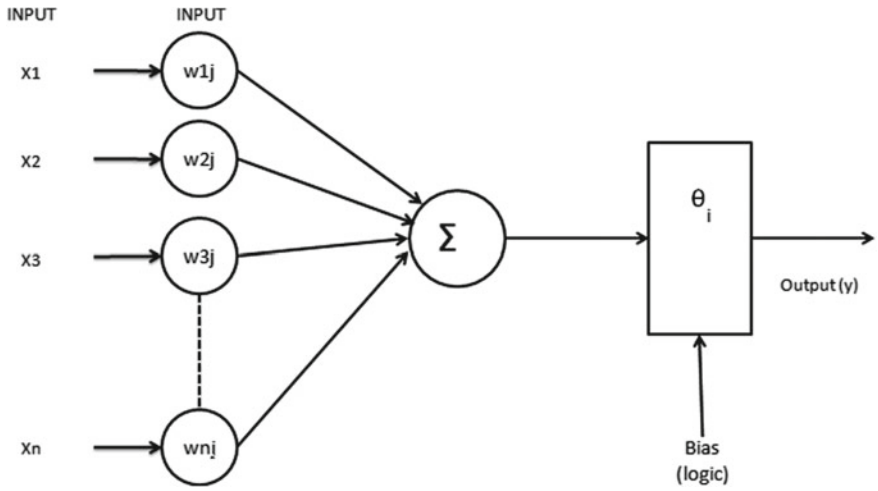


Fig. 1 Mathematical equivalent of an artificial neural network

of the neural network ensures the fact that data can be processed at a high speed [2]. The output of the ANN can be given by the following equation:

$$y = \sum_{i=1}^{n} x_{\mathrm{i}} \cdot w_{\mathrm{i}} + \phi \tag{1}$$

Here,

y denotes output of the ANN
x denotes the inputs to the ANN
w denotes the weights of the ANN
ϕ denotes the bias.

The neural network needs to be trained before it can be used to predict data and be tested for accuracy. There are several mechanisms to train an ANN but the one that is the most effective in reducing the errors quickly is the mechanism of backpropagation. In this paper, the Levenberg–Marquardt backpropagation approach is used.

3 Proposed Approach

(a) Collect raw solar irradiation data (hourly) along with related parameters.
(b) Check data for irregularities such as missing or inappropriate values. Enter average values if such data is found.
(c) Structure data into the following:

    (1) Past 1 h
    (2) Past 2 h, and
    (3) Past 24 h average.

The structuring of data helps in training the neural network regarding both short-term and relatively long-term variations in the data.

(d) Design a neural network and train it with the LM backpropagation algorithm described in Fig. 2:

The Levenberg–Marquardt (LM) algorithm has been implemented in the proposed approach to use backpropagation in the design of the neural network [3]. The mathematical model of the LM algorithm has been illustrated as follows:

Letting the errors in iteration as "i" be designated by e_i

The weight of iteration "i" be denoted by w_i

Then, the weight of the subsequent step w_{i+1} is expressed by the following equation [4]:

$$W_{i+1} = Wi - \left(J_K J_K^T - \mu I\right) e_{Ij} J_K^T \tag{2}$$

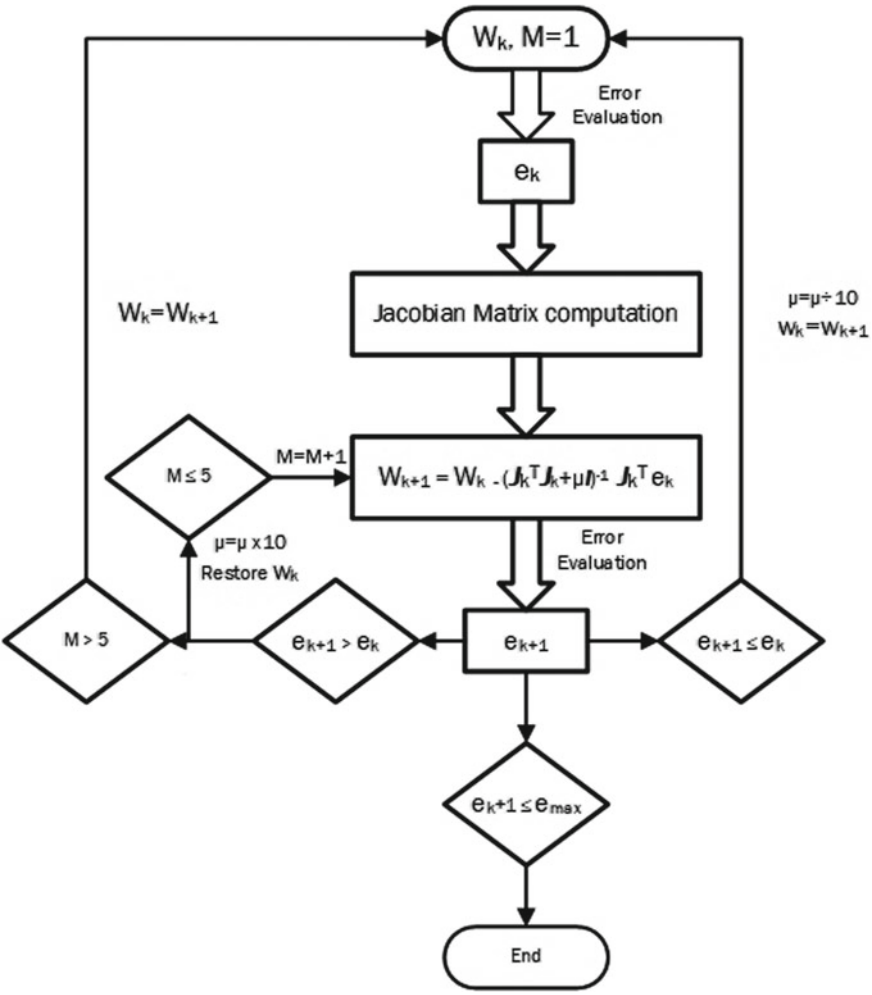


Fig. 2 Flowchart of Levenberg–Marquardt algorithm

Here,

J_K stands for the Jacobian matrix
J_K^T stands for the transpose of the Jacobian matrix
μ stands for step size
I stands for an identity matrix
J_K is the second-order derivative of error (e) with respect to weight (w)

$$\frac{\partial^2 e}{\partial w^2} = J_K \tag{3}$$

The flowchart of the Levenberg–Marquardt (LM) algorithm is shown below

LM is a balanced mixture of two different methods, one being the Hessian and other being the gradient.

Considering the sum of squares as the performance function, then the Hessian matrix and the gradient can be computed using the Jacobian matrix bypassing the need to compute the Hessian matrix directly. The Hessian matrix and the gradient in terms of the Jacobian matrix are given by the following equations:

$$H = J_k^T J_k \tag{4}$$

$$g = J_k^T e \tag{5}$$

(e) Test the network and compute the mean absolute error given by [5]

$$MAE = \frac{1}{N} \sum_{t=1}^{N} \left| A_t - \hat{A}_t \right| \tag{6}$$

4 Results

The data have been taken from the Texas Climate Division (http://mrcc.isws.illinois.edu/CLIMATE/Hourly/StnHourBTD2.jsp) for a period of 26 months (hourly) (Fig. 3).

It can be seen from the above graph that the MAPE is 2.19 only rendering 97.81% accuracy (Fig. 3).

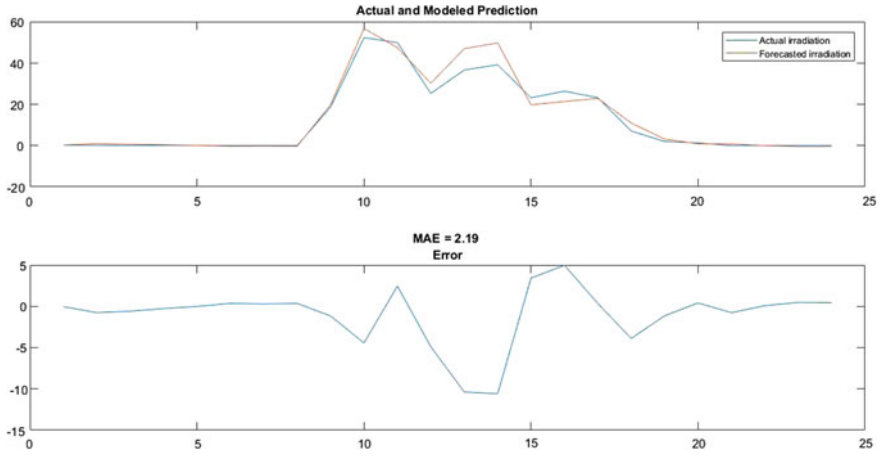


Fig. 3 Actual and predicted solar irradiation (upper graph) and the errors in predicted and actual values (magnified)

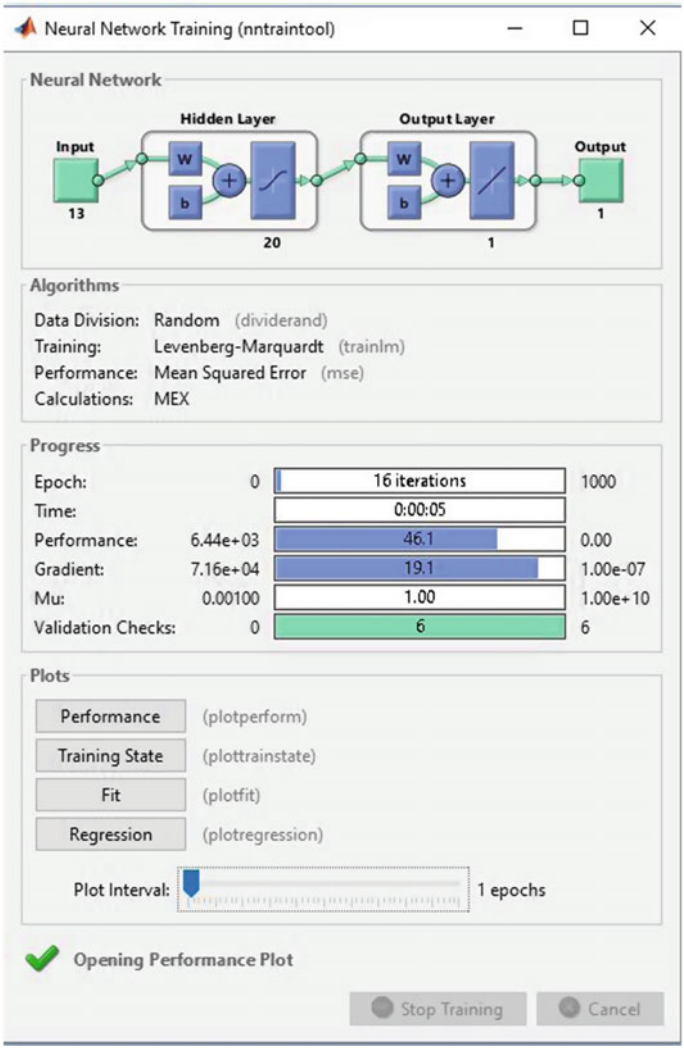


Fig. 4 Opening performance plot using NN training tool for 20 neurons ANN

Figure 4 shows opening performance plot using NN training tool for ANN with 20 neurons in the hidden layer, 13 neurons in the input layer, and one output layer. It can be seen from Fig. 5 that the overall regression of the proposed system is high (0.95) which validates the high accuracy obtained. Comparing with contemporary standard research mechanisms, it can be seen that the proposed system outperforms "**Fully Complex Valued Wavelet Network (CWN) for Forecasting the Global Solar Irradiation**" **by L. Saad Saoud et al. Springer, 2016**, which attains a prediction

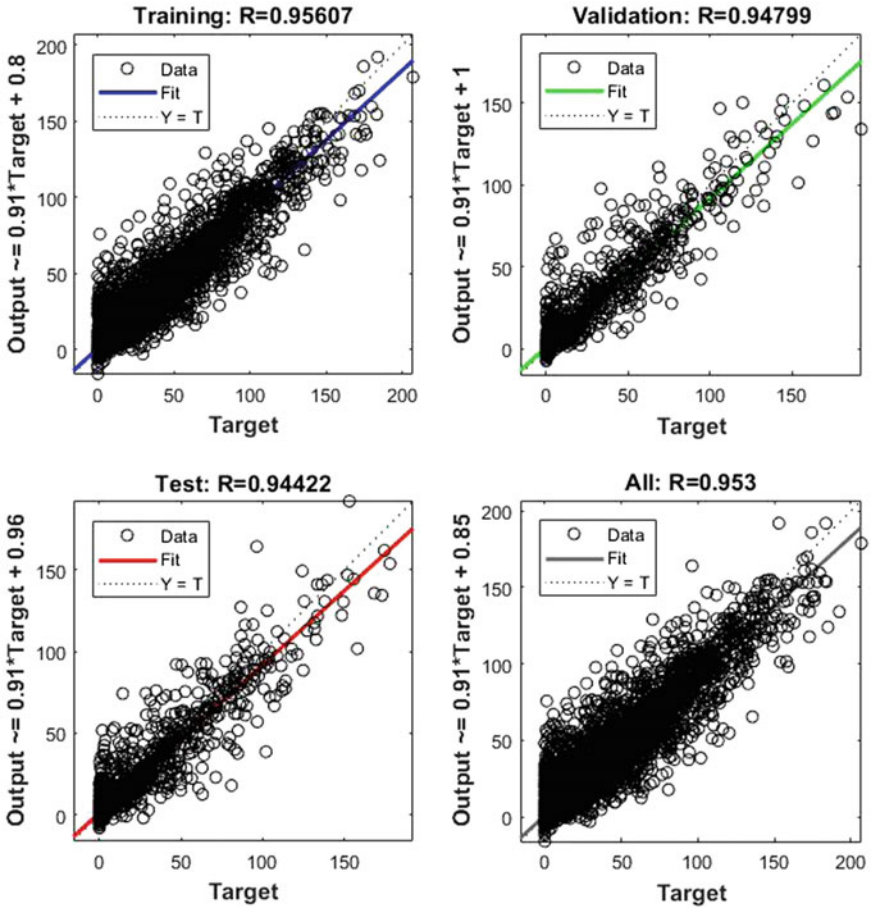


Fig. 5 Regression analysis for training, testing, validation, and overall cases

error of **5.21%**. Moreover, the proposed system attains better prediction accuracy compared to common techniques such as FCWN, CVNN, and CWN [6].

5 Conclusion

This paper presents an ANN-based approach for solar irradiation prediction using Levenberg–Marquardt backpropagation algorithm. It is important to note that the preprocessing of raw solar irradiation data is critical in helping the neural network train better. Structuring the data also plays a pivotal role in increasing the accuracy of the prediction. The overall regression achieved is 0.95 (approx), and the mean absolute percentage error is 2.19%, thereby rendering an accuracy of 97.81%.

References

1. S.L. Goh, M. Chen, D.H. Popovic, K. Aihara, D. Obradovic, D.P. Mandic, Complex-valued forecasting of wind profile. Renew. Energy **31**, 1733–1750 (2006)
2. T. Nitta, *Complex-Valued Neural Networks: Utilizing High-Dimensional Parameters Information Science Reference (an imprint of IGI Global)* (Hershey, New York, 2009)
3. S.L. Goh, D.P. Mandic, Nonlinear adaptive prediction of complex-valued signals by complex-valued PRNN. IEEE Trans. Signal Process. **53**(5), 1827–1836 (2005)
4. S. Suresh, N. Sundararajan, R. Savitha, *Supervised Learning with Complex-Valued Neural Networks* (Springer, Berlin, 2013)
5. B.K. Tripathi, B. Chandra, M. Singh, P.K. Kalra, Complex generalized-mean neuron model and its applications. Appl. Soft Comput. **11**, 768–777 (2011)
6. L. Saad Saoud, F. Rahmoune, V. Tourtchine, K. Baddari, *Fully Complex Valued Wavelet Neural Network for Forecasting the Global Solar Irradiation*

Person-Dependent Face Recognition Using Histogram of Oriented Gradients (HOG) and Convolution Neural Network (CNN)

Priya Tambi, Sarika Jain and Durgesh Kumar Mishra

Abstract Face recognition is not a new phenomenon, but still there is a lot which can be done. We are aiming in this work to develop an improved face recognition system for person-dependent and person-independent variants. To extract relevant facial features, we are using the convolutional neural network. These features allow comparing faces of different subjects in an optimized manner. The system training module, firstly recognizes different subjects of dataset, in another approach, the module processes a different set of new images. Thus, the predictions improve on previously trained data. The accuracy on Yale dataset of 15 people, 11 images per person with a total of 165 images, yielded 96.19% accurate results; the system can be easily scaled for more images and different datasets.

Keywords Convolution neural network · Feature extraction · Image database Preprocessing · Recognition accuracy

1 Introduction

In the last few years, with the growth of security-oriented applications such as access control, card identification, security monitoring, the importance of face recognition has been increased. Face recognition (FR) remains an actively studied topic in computer vision community and pattern recognition in comparison with other biometric recognitions [1].

In recent years, deep learning methods, especially the deep convolutional neural networks (CNNs), have achieved noteworthy successes in various computer vision applications, varying from image classification, segmentation, object detection, and many more. Deep learning method avoid preprocessing hazards and improves the evaluation technique [2]. Real world examples are auto-tagging in Facebook and mobile application face lock which opens the phone's lock using face recognition.

P. Tambi (✉) · S. Jain · D. K. Mishra
Sri Aurobindo Institute of Technology, Indore, India
e-mail: priyatambi20@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_5

If we get deeper into how it works than algorithms are able to recognize the images which are tagged a certain number of time. These tagged images now become the training data on which an algorithm works for recognition purposes. It seems miraculous to us due to high recognition accuracy of 98% achieved by Facebook.

These considerations have motivated us to develop a face recognition system that models human behaviors in common face recognition scenarios. Our work evaluates the methods used to determine if pairs of face images belong to the same individual or not. The method uses combination of HOG and convolution neural networks. The NN are trained and evaluated using the Yale dataset.

2 Proposed HOG-CNN Method

We have divided it into four small components:

1. Indoctrination of a picture using the HOG algorithm: It attempts to capture the shape of systems inside the region with the aid of shooting statistics approximately gradients. It does so by dividing the image into small (normally 8×8 pixels) cells and blocks of 4×4 cells.
2. Now, the pose of the face is being generated with main landmarks present in the face. A centered image is then engendered with the help of these landmarks.
3. This centered image is now passed through CNN to get 128 measurements.
4. The measurement of training data and testing data is then compared, and the images with similar measurements are considered to be of same subject.

2.1 Detecting Face Using HOG

Face detection came into existence at beginning of 2000s when Paul Viola and Michael Jones invented a very fast technique based on Haar-like features to detect the faces. The approach was implemented using low-resolution cameras and hence improved a lot in past 15 years. We're going to use a method invented in 2005 called the histogram of oriented gradients [3]. To locate the faces in an image, we'll first convert it into gray scale. Then, all pixels are scanned one at time. For every single pixel, we will look at the pixels available in the neighborhood. Once we'll get the HOG of the image, we'll compare it with the HOG of training images.

This technique will help us detecting the faces in any images of the dataset. Fig. 1 shows representation of the images using HOG.

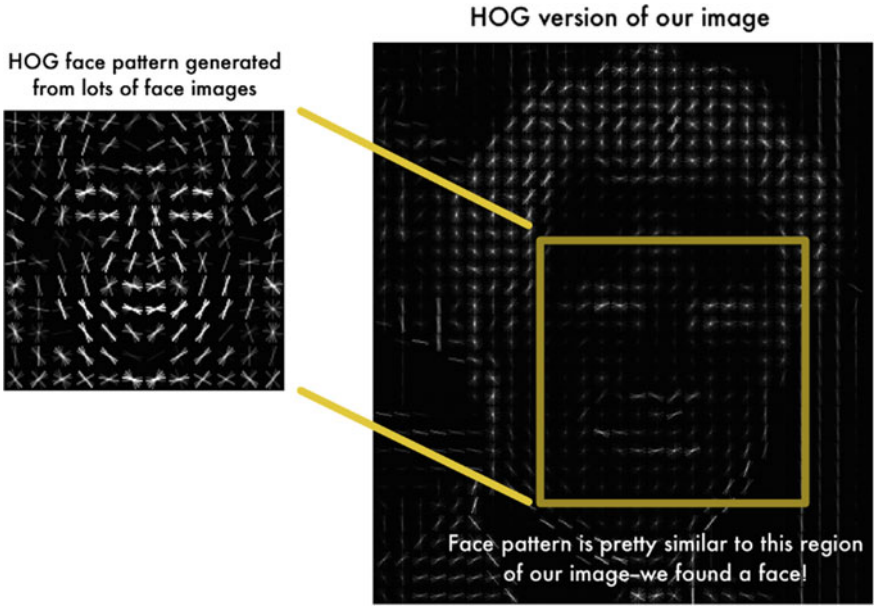


Fig. 1 HOG representations of images using the HOG

2.2 Pose-Variant Faces

Consider extraction of the facial part of an image. Images of same person can be taken from various angles and can in various poses. This creates difficulties for extracting the features. Figure 2 shows an example images of Indian Prime Minister Mr. Narendra Modi from various angles and in various poses.

Humans can easily recognize that all images are of same person, Mr. Narendra Modi, but computers would consider these images as four different persons.

(A) Front Face (B) Right Tilted (C) Back Tilted (D) Left Tilted

Fig. 2 Example of image from different angles and poses

All facial features wrapped and located in the specified subspace in the image. We estimate the facial landmarks, using method proposed in 2014 by Sherif et al. [4].

We'll now determine the landmark points which are on every face wiz are, comprised of points at chin, the edge of jawline reaching towards the eyebrow etc. Training will be done on the basis of our algorithm. This is the step which provides the needed 68 specific points on any face: Now, we know the exact location points for facial features; we can apply image operations such as rotation, scaling, and other to get the perfectly centered features on the image.

2.3 Encoding Faces Using CNN

Our requirement is to extract features for each face in the dataset. The features are compared in test and train set images. We can identify same person by differences in images and measurements. The lesser is the difference, the higher is probability of similarity in image. We use the 128 measurements instead of complete image for convolution neural network (CNN); CNN is a feed-forward network. In training. We use three images at a time, of which two images are of similar subject from the dataset and one is of other subjects. CNN extracts features from the raw image, and then, a classifier classifies the extracted features [5] Fig. 3 shows the method.

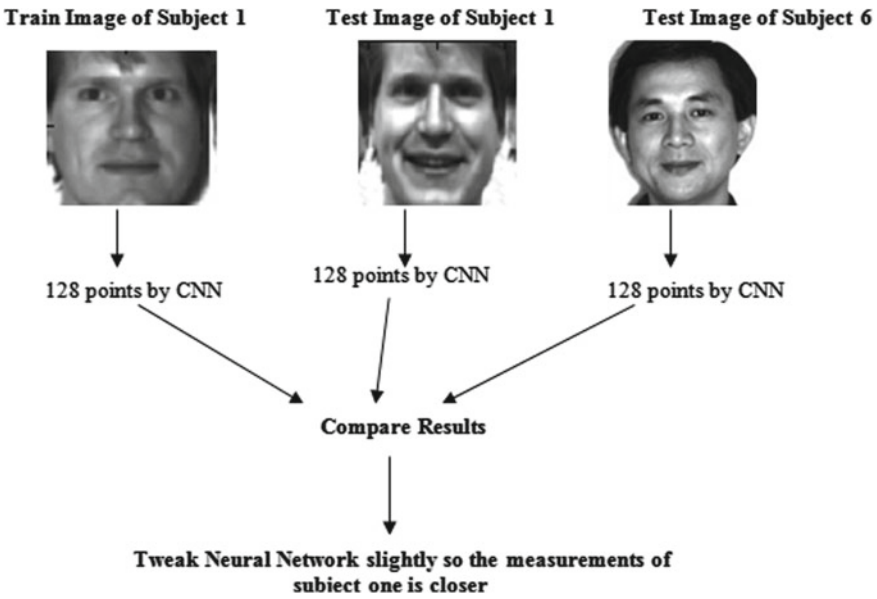


Fig. 3 CNN extraction method for extraction followed by classification

The step is rehashed for numerous iterations and distinctive confront pictures. The neural network organize and learns, and create 128 estimations for each individual. Any ten distinctive pictures of the same individual ought to deliver generally the 85 same estimations. The computer-generated numbers is a correct approach for faces recognition, suggested by analysts at Google in 2015 [6].

2.4 Person-Dependent Recognition

The final step is to discover the individual in our database of known individuals who have the closest estimations to our test picture. It can be done by utilizing any fundamental calculation [7]. We'll utilize a basic direct SVM classifier, prepare a classifier. Running the classifier on the top of Yale dataset will take few minutes. The outcome will be the label of the image. Title of the individual gives the identification in present case.

3 Experiment Setup

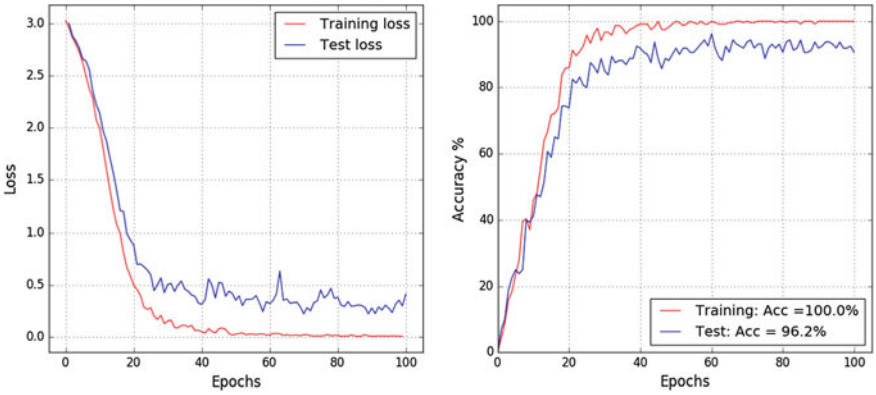
The handle CNN organizes to yield confront embedded parcel of information. We use Open-Face, and organize Python-based confront acknowledgment library for face_recognition. Overall time of the process reduces. We use pre-installed VM with all necessary tools installed. That can reduce the in general computation time (Table 1).

Table 1 Recognition accuracy for the different training and testing datasets

Count of data	Size of training set	Size of testing set	Train samples/class	Test samples/class	Recognition accuracy
165	105	60	7	4	73.21
165	120	45	8	3	83.03
165	135	30	9	2	91.79

4 Results of Simulation

We use two layers CNN with different Network parameters in each Layer from first layer. Filter size is 5 and 16 filters are used while maximum pooling is set to 2. Filters are increased for layer 2 up to 36 and pooling is negative 2. Regularization 1 dropout is 0.5. The outcome is displayed by a plot of the loss function and accuracy.



References

1. T. Priya, J. Sarika, IJournals: Int. J. Softw. Hardware Res. Eng. **5**(9) ISSN-2347-4890 (2017)
2. A. Krizhevsky, I. Sutskever, G.E. Hinton. Imagenet classification with deep convolutional neural networks, in *Advances in Neural Information Processing Systems*, pp. 1097–1105 (2012)
3. Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition. Proc. IEEE **86**(11), 2278–2324 (1998)
4. M. Sherif, M. Azab, Y. Emad, S. Sameh, F. Menisy, B. Kandil, Eyes in the sky, in *Companion Proceedings of the 10th International Conference on Utility and Cloud Computing–UCC '17 Companion* (2017)
5. S.B. Lo, H. Li, Y. Wang, L. Kinnard, M.T. Freedman, A multiple circular path convolution neural network system for detection of mammographic masses. IEEE Trans. Med. Imag. 150–158 (2002)
6. Y.H. Hu, J. Hwang, *Handbook of Neural Network Signal Processing*, CRC Press. Gepperth A. (ed.) Object detection and feature base learning by sparse convolutional neural networks, Lecture notes in artificial intelligence 4807 (Springer, Berlin, 2002), pp. 221–231
7. H. Khalajzadeh, M. Mansouri, M. Teshnehlab, Face Recognition using Convolutional Neural Network and Simple Logistic Classifier. WSC17 (2017)

Movie Rating Prediction System Based on Opinion Mining and Artificial Neural Networks

Somdutta Basu

Abstract The paper presents an artificial intelligence based approach for movie rating prediction which can be useful for producers, distributors and viewer. In the proposed system, six different parameters and along with their individual ratings are used as features to train an artificial neural network. The algorithm used to train the neural network is the Levenberg-Marquardt back propagation algorithm. It can be observed that the proposed algorithm attains steep descent in prediction errors due to the mechanism of back propagation. As a standard convention, 70% of the data has been used for training and 30% data has been used for testing. The data set contains of 150 movies and the data is collected from IMDB. The proposed system achieves an accuracy of 97.33% and a sensitivity of 98.63%. An overall regression of 0.9182 has been attained exhibiting the fact that actual and predicted data bear high resemblance. It has been shown that the proposed technique achieves higher accuracy compared to contemporary techniques.

Keywords Opinion mining · Artificial neural network (ANN)
Levenberg-Marquardt back propagation · Accuracy · Sensitivity

1 Introduction

The movie industry now has a global Diaspora and a lot of financial and time investments. Several approaches are being developed to predict box office revenues and movie ratings. With the emergence of artificial intelligence, box office prediction techniques are focussing towards artificial intelligence and associated probabilistic techniques [1]. This paper presents an approach based on the Levenberg-Marquardt back propagation algorithm for movie rating prediction. The data used has been taken from IMDB for the years 2016 and 2017. A total of 150 Hindi movies have been taken as the data set. The performance metrics used to evaluate the performance of

S. Basu (✉)
Information Technology, SSCET, Bhilai, CG, India
e-mail: somdutta1419@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_6

the proposed system are accuracy and sensitivity. The overall system performance of the designed neural network is evaluated using regression and mean square error.

2 Related Work

Several technique are employed to predict movie ratings based on opinion mining out of which the contemporary approaches are discussed in the paper titled "Predicting movie success with machine learning techniques: ways to improve accuracy" by Lee et al. [2]. This paper proposes an ensemble approach in machine learning and attains an average accuracy of 89.2%. Several other techniques such as Gradient Tree Boosting, Random Forests, and Support Vector Machine (SVM) etc. have also been compared. It has been found that the paper achieves higher accuracy compared to the standard techniques [3].

3 Materials and Methods

The main approach for the design and training of the neural network is the Levenberg-Marquardt back propagation algorithm. The training rule for the algorithm can be described by the mathematical relation [4]:

$$W_{k+1} = W_k - [J_k{}^T J_k + \mu I]^{-1} J_k{}^T e_k \tag{1}$$

Here,

W_K represents the weights of Kth iteration,
W_{K+1} represents the weights of the $(K+1)$st iteration
J_K represents the Jacobian Matrix
J_K^T represents the Transpose of the Jacobian matrix
I is an identity matrix
e_K represents the error in the Kth iteration.

The efficacy of the algorithm lies in the fact that it computes the Hessian matrix indirectly using the Jacobian Matrix which is the double differential rate of change of error with respect to the weights [5]. The maxima (point of inflexion) of the Jacobian matrix indicate the minima or steepest descent in the errors given by:

$$J_k = \frac{\partial^2 e}{\partial w^2} \tag{2}$$

It indirectly computes the Hessian Matrix given by:

$$H = J_k J_k{}^T \tag{3}$$

The approach results in the steepest descent of the error in each iteration with evading the complexity of computing the Hessian Matrix. This can inferred at hindsight form the value of mean square error and the number of epochs in the training of the algorithm.

The approach involves the following steps:

Step 1. Data mining of box office data (IMDB in this case).

Step 2. Structuring of data to extract feature values which in this case have been considered as:
(1) Actor (2) Actress (3) Director (4) Producer (5) Writer (6) Music Director.

Step 3. Convert textual features to numeric values by assigning individual parameter ratings, v.i.z.
$\mathbf{I_1}, \mathbf{I_2}, \ldots, \mathbf{I_6}$

Step 4. Prepare the target vector for the neural network using database's actual rating values. Divide the target vector into three classes based on thresholding i.e.

$\mathbf{0 < R < 40 \rightarrow flop}$
$\mathbf{40 < R < 70 \rightarrow average}$
$\mathbf{70 < R < 100 \rightarrow hit}$

Step 5. Design a Neural Network based on the Levenberg-Marquardt Back Propagation algorithm.

Step 6. Apply training and testing data samples to evaluate the predictive numeric value for each tested movie. Let the unique predictive value be designated by $\mathbf{P_{i.}}$

Step 7. Compute the final category or rating based on the following condition:

if $\mathbf{0 < M_i < 40 \rightarrow flop}$
else if $\mathbf{40 < M_i < 70 \rightarrow average}$
else $\mathbf{\rightarrow hit}$

Step 8. Compute the performance metrics for two categories:

1. **System performance evaluation**
   Compute Regression, Mean Square Error and Training States.
   This step validates the performance of the designed ANN structure.
2. **Overall Prediction performance evaluation**
   Compute Accuracy and Sensitivity.

The performance metrics used to evaluate the performance of the proposed system are:

(1) Mean square error (MSE) which is mathematically defined as:

$$MSE = \frac{1}{N}\sum_{k=1}^{N} {e_k}^2 \tag{4}$$

This metric evaluates the error in predicting the predictive numeric score ($\mathbf{M_i}$)

(2) **Accuracy** (**Ac**) defined as:

$$\text{Ac} = (\text{TP} + \text{TN})/(\text{TP} + \text{TN} + \text{FP} + \text{FN}) \tag{5}$$

(3) **Sensitivity** (**Se**) defined as:

$$\text{Se} = (\text{TP})/(\text{TP} + \text{FN}) \tag{6}$$

Here,

TP stands for true Positive
TN stands for true Negative
FP stands for False Positive
FN stands for False Negative.

4 Results and Discussions

The simulations are run on MATLAB 2017a. The results and their corresponding explanations are given below:

Figure 1 depicts the designed ANN structure with its performance metrics. The training stops after 8 iterations yielding an MSE of 0.00538. Six iterations for the validation are done after which the training stops. The six inputs signify the six features considered for training. The output signifies the final result in the form of a hit, average or flop movie (Fig. 2).

The obtained overall regression is found to be 0.91882 which is a relatively high value and indicates the similarity between actual and predicted values.

The Accuracy and Sensitivity obtained are based on the TP, TN, FP and FN values obtained.

Of the 150 values the accuracy is computed as:

$$\mathbf{Ac} = (\mathbf{72} + \mathbf{71})/(\mathbf{72} + \mathbf{71} + \mathbf{3} + \mathbf{1}) = \mathbf{97.33}\%$$

The sensitivity is computed as:

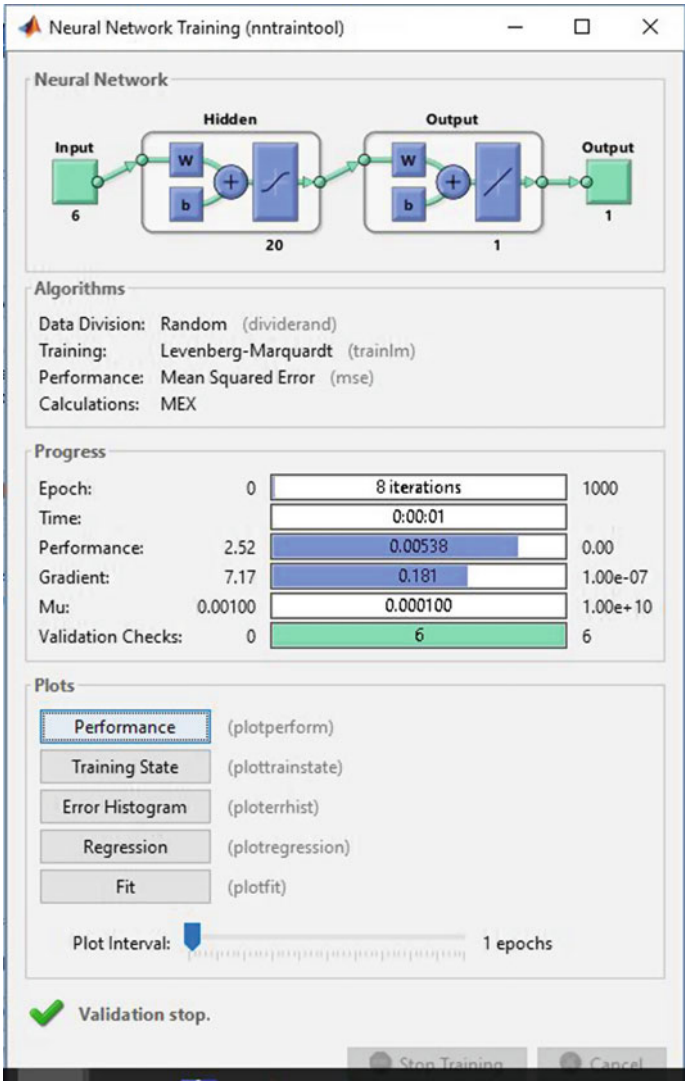


Fig. 1 Performance of the designed ANN structure

$$\mathbf{Se} = (\mathbf{72})/(\mathbf{72}+\mathbf{1}) = \mathbf{98.63}\%$$

Thus it can be seen that the proposed technique achieves batter accuracy compared to commonly used techniques described in [2, 6].

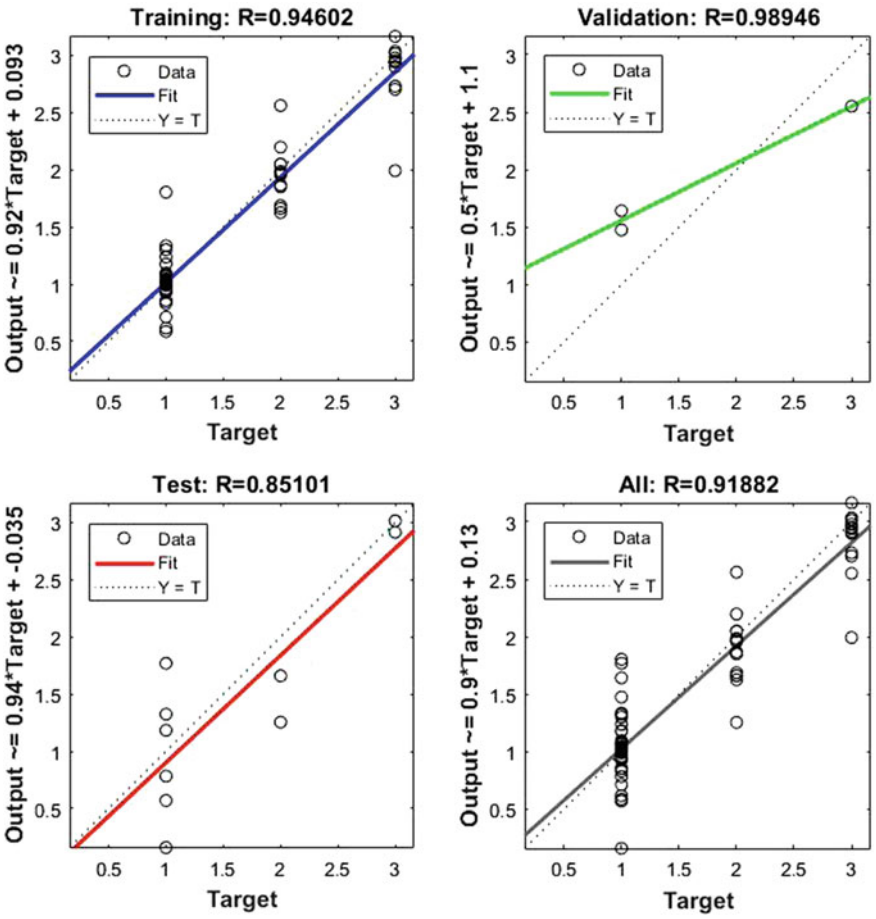


Fig. 2 Regression analysis of proposed system

5 Conclusion

This paper proposes a technique for movie rating prediction based on data mining and artificial neural networks. The mechanism uses individual ratings for the different feature values related to the movie's success. In this approach, six parameters or features have been considered which are thought to affect the results of a movie the most. The neural network architecture used here is the Levenberg-Marquardt back propagation algorithm due to speed and accuracy. It has been found that the proposed system attains a regression of 0.91882, an accuracy of 97.33% and a sensitivity of 98.63%. Future approaches may use more parameters to attain a more comprehensive prediction approach.

References

1. B. Pang, L. Lee, S. Vaithyanathan, Thumbs up?: sentiment classification using machine learning techniques, in *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing*, vol. 10. Association for Computational Linguistics (2002)
2. K. Lee, Predicting movie success with machine learning techniques: ways to improve accuracy (across local markets). Mark. Sci. **29**(5), 944–957. Springer 2016
3. T. Liu, X. Ding, Y. Cheng, H. Chen, M. Guo, Predicting movie Box-office revenues by exploiting large-scale social media content. Springer 201 (2014)
4. D.W. Marqaurdt, An algorithm for least squares estimation of non-linear parameters. J. Soc. Ind. Appl. Math. (1963)
5. B.E. Boser, I.M. Guyon, V.N. Vapnik, A training algorithm for optimal margin classifiers. ACM (1992)
6. J.S. Simonoff, I.R. Sparrow, Predicting movie grosses: winners and losers, blockbusters and sleepers. Chance (2000)

Classification of Satellite Images on HDFS Using Deep Neural Networks

Arushi Patni, Kunal Chandelkar, Rajesh Kumar Chakrawarti and Anand Rajavat

Abstract Nowadays, the artificial intelligence plays an important role in technological aspects. AI uses big data for performing its job, therefore it is necessary to store, analyze, and control the big data. AI needs to perform works like face detection and image classification. The classification process categorizes small pixels that are present inside a digital image into several different classes. Generally for classification, multispectral data were used by obtaining the spectral pattern present between image data and the numerical basis for this categorization are pixels. We propose an open-source Hadoop image classification platform. The goal of this research is to make a product that will, in future, be used to make image classification and processing tool on extensive number of images which will empower scholars and scientists to create applications.

Keywords Classification · Image · Neural network · Computer science · Hadoop

1 Introduction

There is a rapid increase in data in almost every field like social media, entertainment, and science. MapReduce frameworks like Hadoop are getting in use to handle big data-related problems [1].

A. Patni · K. Chandelkar (✉) · R. K. Chakrawarti · A. Rajavat
Shri Vaishnav Institute of Technology and Science, Indore, Madhya Pradesh, India
e-mail: kunalchandelkar@gmail.com

A. Patni
e-mail: arushi496patni@gmail.com

R. K. Chakrawarti
e-mail: rajesh_kr_chakra@yahoo.com

A. Rajavat
e-mail: anandrajavat@yahoo.co.in

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_7

We make our research with the aim of providing a platform for all image classification along with computer pixels or vision tools to withstand continual changes and provide improvements within Hadoop MapReduce system.

Image classification starts with the notion that we build a training set and that computers are equipped to recognize and categorize what they are processing. Amazon web services initiatives like public data sets and Indfochimps.org are always encourage the "information commons" in which data are accessible freely to download and analyze. For example, there is a project based on .net named Astrometry which keeps an eye on Astrometry group on Flickr for night sky photographs. It performs the analysis on images and detects which part of the sky it is from. [Hadoop A Definitive guide: O Reilly].

2 Literature Survey

Roshan Rajak and team have tried to come up with an approach as to how satellite images could be processed using Hadoop framework [2]. Soheil Bahrampour, Naveen Ramakrishnan, Lukas Schott, and Mohak Shah have tried to compare the various neural network frameworks such as Caffe, Neon, Theano, and Torch and have come up with a solution that Caffe has proved to be the best, as it has been found to be the easiest for evaluating the performance of standard deep architectures [3]. Renat has tried to apply HDFS processing and downloading of satellite images and their classification [4]. When we are dealing with millions of images then just download processes may take many number of days to accomplish on a single machine. To overcome this, we can perform these processes in parallel, by parallelizing execution on and within one machine using GPU and across many machines depending on Hadoop. The idea has been explained about how satellite image classification could be performed in remote sensing [5]. MapReduce delivers remarkably strong framework which works on data in-depth tools where the model for data handling is similar.

3 Problem Identified

Following are the points that cover problems identified:

- **Point of view variation**. A camera can show many ways of orientation of a object with single instance.
- **Size variation**. Variations are mainly present in visual classes.
- **Distortion**. Most of the objects are not rigid bodies and can be distorted in many ways.
- **Blockage**. Many times only a small amount of object could be visible and hence other portions are blocked.
- **Radiance conditions**. The pixel level is drastically affected by light.

- **Background jumbled**. Sometimes it is hard to identify the objects which are blend into their environment.
- **Internal variation**. There are a broad range of objects present in a single type of class.

4 Objectives

As per our research, there are the following objectives that we try to achieve:

- To create an environment that processes large image sets in a HDFS platform.
- To create an image classification platform for satellite images that could be used for research purpose for students, researchers as well as geologists.
- To understand neural network and work on its frameworks.

5 Solution Approach

5.1 Hadoop

Hadoop is designed to make servers capable of scaling themselves from single to hundreds of machines, and each of them provides services like local computation and data storage.

Standard Hadoop MapReduce programs face difficulties in showing images in a format that is useful for scientists. For example, to give a set of images to a set of mapper nodes would need a user to pass the images as a string and then decode every image in each map task before being able to get information [1].

5.2 Neural Network

An artificial neural network (ANN) is a type of information processing concept which is adopted from the biological nervous systems. It is organised in a specific manner as per the requirement, like data classification or pattern recognition by a learning process [6].

Due to the advancement in technologies like deep neural networks, it can be now possible to make automatic image classification with nearly human-like performance.

Projects like Theano, Caffe, DeepNet, and Toronto provide tools for implementing NNs. with [7].

6 Architecture Model

6.1 Data Storage

Hadoop tries to read images in a text file and fails to do that. To deal with this difficulty in configuring, Hadoop uses an Amazon's web service named Elastic MapReduce, i.e., EMR which provides APIs to install or configure a Hadoop cluster, so that it can run jobs [8]. The input format accepted by EMR is a file or a folder in which text files are present. When we are handling a very large amount of files or images, then we create set of files, where each file have around 1000 records in the following format:

photo_id s3://input_photo.jpg s3://output_classification.json

By default, EMR outputs data as one large list splits among many *part-xxxxx*files. Parsing is hard to split files since they could be very large and they are not arranged with new line boundaries. We store each image as a separate JSON file for our classification. This is why we explicitly specified destination location in the input for Hadoop so each worker can save results for each image.

6.2 Image Processing

There are two types of classes which can be used for implementation. They are RecordReader and InputFormat. InputFormat gets record in key-value pair format from RecordReader class. After that, key-value pair is passed to the Map function for processing [2].

Steps for image processing:

(a) **Hewing of image**: Image file is taken as input, and split operation is performed on it; after that, image gets stored.
(b) **Serialization of hewed image**: Each set of hew image files are converted to serialized form (.ser extension).
(c) **Serialization for HDFS block**: The .ser files are not optimized for HDFS. These are combined to get "bser" files.
(d) **Copy to HDFS**: The final obtained file, HFi.bser, is copied to the HDFS (Fig. 1).

6.3 Application Model

The user can choose a location of their choice on the satellite and download the image. The image size is set as per the application. The user chooses the areas that need to be studied such as land area, vegetation, and water bodies. The image is streamed

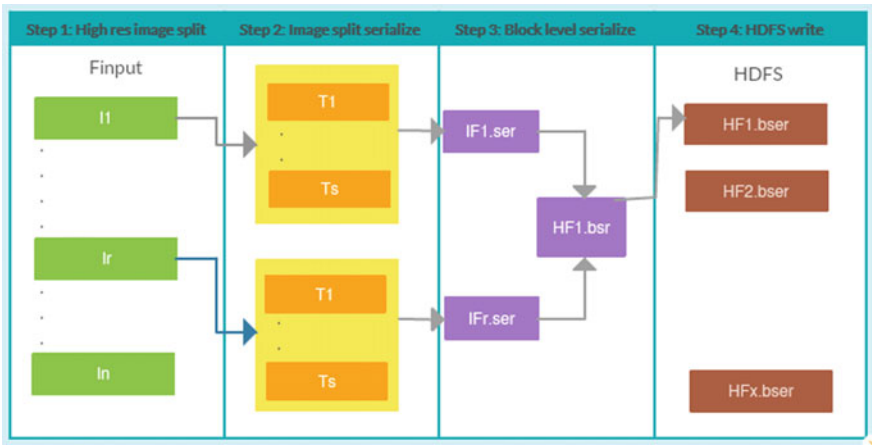


Fig. 1 Image processing

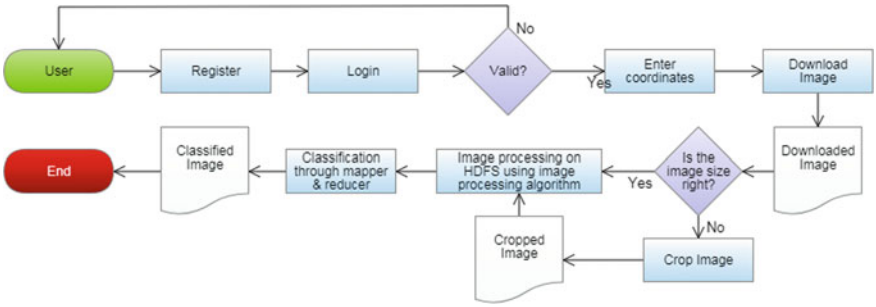


Fig. 2 Proposed architectural model

on HDFS cluster, and the data are evaluated and classified. Classified image is then sent to the user, which could be saved. In the figure, we have explained how the application will work (Fig. 2).

Algorithm

1. Train a classifier f from (Xl, Yl)
2. Utilize f to classify all unlabeled item xϵ Xu
3. Pick x* with upper most: Working Model assurance add (x*, f(x*)) to labeled data.
4. Repeat [3].

Where

Input x, label y Classifier $f : x \rightarrow y$

Labeled data (Xl, Yl) = {(x1, y1) … (xl, yl)}

Unlabeled data Xu = (xl + 1 … xn); usually = $n \gg l$.

6.4 MapReduce

When the image is on HDFS, its processing is done by MapReduce programs. Hadoop MapReduce does distributed processing by parallel programming technique that is implemented on top of HDFS [9]. The Hadoop MapReduce engine consists of a JobTracker and several TaskTrackers. TaskTrackers maintain the progress and status on a particular data node and report regularly to JobTrackers. The JobTracker splits the job executed by MapReduce into smaller tasks that are later handled by TaskTrackers. In the Map step, the master node takes the input, divides it into smaller subproblems, and distributes them to worker nodes. The values with the same key are processed by the same machine to make the final output. This process is done in Reduce step. The image data are first sent to Feature Mapper function, which extracts information out of the images data. Classification Mapper and Classification Reducers perform image classification.

7 Outcome

As per the research, we can create software which can work in Hadoop environment that processes large image sets in a HDFS platform. We are able to create image processing software that could stream image data on HDFS also by HIPI [10]. By this, we have tried to classify images based on their image pixel data. We can create software that uses neural network technology and could be used as a tool in AI.

8 Benefits

The research provides an easy approach for classification of image. The study of research can be used to make project software for classification of. It can handle large number of images since it is in Hadoop ecosystem.

Following are the widely uses image classification sectors:

1. **Research**: The proposed technology could be used by researchers and geologists to understand the soil, water, and land compositions of an area.
2. **Remote Sensing**: This technique could be used to understand the various geographical images that are detect by remote sensors [11].
3. **Space Research**: This could be used in space research to understand the area on any planet.
4. **Weather Forecast**: This could be used to estimate, that if a calamity strikes a particular area, what amount of damage could occur.
5. **Artificial Intelligence**: This technology could further be used in robotics.

9 Limitations

The proposed project has many benefits, but there are also a few limitations:

- It requires high-speed data transfer rate since it is Hadoop application [12]. The modern technology has evolved, but could take days to download image.
- It has slow processing as it uses Hadoop technology. This project is only to be used for big data satellite images.

10 Conclusions and Future Scope

In future, MapReduce application programming interface could be combined for a lot of more images giving out and computer vision modules.

The proposed project could be very useful in future in robotics to perform image classification. It could be further expanded to predict the outcomes if a natural calamity occurs, and compare the change in geography of an area.

References

1. R.S. Ferreira, P. Happ, D. Oliveira, R. Feitosa, G. Costa, P. Plaza, Classification algorithms for big data analysis, a MapReduce approach, in *Joint ISPRS Conference*, Munich, Germany (2015)
2. R. Rajak, D. Raveendran, M. Chandrasekhar, S.S. Medasani, High resolution satellite image processing using Hadoop framework, in *IEEE International Conference on Cloud Computing in Emerging Markets* (2015), pp. 16–21
3. S. Bahrampour, N. Ramakrishnan, L. Schott, M. Shah, Comparative study of Caffe, Neon, Theano and Torch for deep learning, in *Workshop track—ICLR 2016*. Bosch Research and Technology Center (2016)
4. Renat, Image classification with Hadoop Streaming, https://developers.500px.com/image-classification-with-hadoop-streaming-1aa18b81e22b
5. Supervised classification of land cover using Sentinel-2 images, https://fromgistors.blogspot.com/2016/09/basic-tutorial-2.html
6. C. Stergiou, D. Siganos, https://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html
7. Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, T. Darrell, Caffe: convolutional architecture for fast feature embedding, in *EECS* (Berkeley, CA 94702)
8. M. Adnan, M. Afzal, M. Aslam, R. Jan, A.M. Martinez-Enriquez, Minimizing big data problems using cloud computing based on Hadoop architecture
9. G. Yang, The application of mapreduce in the cloud computing
10. C. Sweeney, L. Liu, S. Arietta, J. Lawrence, *HIPI: A Hadoop Image Processing Interface for Image-Based MapReduce Tasks*, https://www.researchgate.net/publication/266464321_HIPI_A_Hadoop_Image_Processing_Interface_for_Image-based_MapReduce_Tasks
11. *Image Classification Techniques in Remote Sensing*, http://gisgeography.com/image-classification-techniques-remote-sensing/
12. T. White, *Hadoop the Definitive Guide: Storage and Analysis at Internet Scale*, 4th ed. (O'Reilly', 2015)

Enhanced Entity Extraction Using Big Data Mechanics

Adyasha Dash, Manjusha Pandey and Siddharth Rautaray

Abstract With the advancements of new technologies, a large volume of digital data is getting generated every second from various internal and external sources like social networking, organizations and any business applications. Big data refers to enormous digital data that are high in volume, velocity, varieties. The traditional conventional approach fails to handle large data sets using their tools and techniques. Big data proved to be an effective mean for collecting, analyzing and processing data despite their size and data formats structured, semi-structured or unstructured. Large set of information and data are produced from different organizations and social activities. Text mining or text analytics plays a significant role in deriving relevant information from text in digital environment. Text mining includes technique like entity extraction which automatically extracts structured information from unstructured or semi-structured documents. This paper details how entity extraction is useful in processing human language texts by using natural language processing. Entity extraction based on method like part-of-speech tagging which helps in determining the noun, verb, adverb and adjectives associated with a sentence. Enhanced entity extraction method will be mainly useful for filtering entities based on their part-of-speeches by removing any ambiguities. Entity extraction focuses on relevant parts of a document and represents them in a structured manner.

Keywords Big data · Digitization · Entity extraction · Named entity recognition Natural language processing

A. Dash (✉) · M. Pandey · S. Rautaray
School of Computer Engineering, Data Science Center of Excellence,
Kalinga Institute of Industrial Technology (KIIT) University, Bhubaneswar, India
e-mail: adi92dash@gmail.com

M. Pandey
e-mail: manjushapandey82@gmail.com

S. Rautaray
e-mail: sr.rgpv@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_8

1 Introduction

In the current generation, the process of digitization helped the mankind in processing, storing and transmitting data in an efficient way. Big data is the most promising trend among the researchers because of its magnificent way in making the immense and complicated data simpler and structured. The traditional relational database management system like DB2, SQL server was incapable of processing large and unstructured data sets. A simple comparison states that the amount of data produced in beginning of time till 2005 was 5 billion gigabytes and the same amount of data was generated in every twenty days in 2011 and in the growing era the same volume of data is coming from various sources in just one day. Big data analytics is the field which deals with collecting voluminous big data from different data sources and analyzes the data and organizes it to infer useful information patterns [1]. The researchers are making a lot of efforts to find new advanced tools and techniques for analysis of data. The most important feature of big data is that it can handle unstructured data which was a limitation in the traditional approach. Text analytics in simple term is the ability to process unstructured text from a very large set of documents interpret the meaning and automatically identify and extract the concepts hidden in it and also the relationship between concepts [2]. Entity extraction is important in text analytics because it finds relevant entities from text and gathers information from pieces of text and finally produces structured representation of those related information. The advantage of entity extraction process is that it tends to classify entity in a text in categories like noun, pronoun, and verb for simplification of further natural language processing (NLP)-based tasks. The concept of knowledge discovery is applied on entity extraction [3] (Fig. 1).

The research paper is organized in the following manner. Section 1 is an introduction to text analytics, entity extraction in context of big data analytics. Section 2 discusses literature review of text analytics. It gives brief idea on different approaches for data analytics. Section 3 deals with proposed enhanced entity extraction framework. Section 4 sheds light on advantages and applicability of proposed frameworks in multitude of domains.

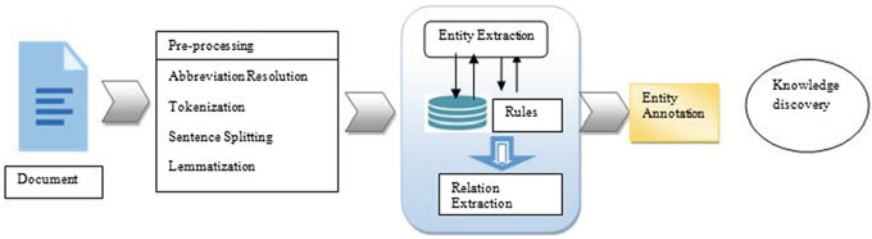


Fig. 1 Overview of entity extraction process

1.1 Text Analytics

Text analytics is commonly known as text mining or text data mining. It is the process in which high-quality information are derived from text data. Text analytics usually involves the process of structuring the input text by adding some linguistic features and removal of irrelevant information, deriving patterns within the structured data. Text analytics involves sub tasks like text categorization, text clustering, concept/entity extraction, sentiment analysis, document summarization and entity relation modeling. Text analysis turns text into data for analysis by using the analytics method and NLP. A general text contains entities, attributes of different forms. An entity is a real-world object. An entity in a text is a proper noun such as a person, place, and product. Most text analytics apply natural language processing such as part-of-speech (PoS) tagging, syntactic parsing and few linguistic analyses. Text analysis also identifies the association among different entities. Many unstructured text data are mounting up in social media and emails.

Many renounced organizations rely on such text-based information but nowadays text analytics along with the use of NLP is the appropriate platform for utilization of large data. NLP makes natural languages accessible to machines. Text analytics has drawn remarkable changes in different sectors like retail industry for product analytics based on the feedback given by its customers, in finance sector for entity matching, for analyzing the overall content of any audio or video files in social websites, in medical science for content extraction of patient record and also in government sectors. NLP with data analytics has major contribution on entity extraction and finding appropriate relationships among the entities and part-of-speech.

1.2 Stages of Entity Extraction Using NLP

There are three ranges of natural language processing for linguistic analysis.

- Syntax Analysis: Syntax analysis focuses on the grammatical knowledge of a given text.
- Semantic Analysis: It deals with the internal meaning given in any text.
- Pragmatic Analysis: Pragmatic analysis says the purpose of a text.

NLP comprises of two stages as (i) understanding the natural language and (ii) Generation of the Language. The meaning of a given text will be well understood by NLP, the nature, structure and type of a word in a text is handled by NLP, Different words and sentences has multiple meanings, so NLP removes all ambiguities related to such text and generate refined text data.

Figure 2 depicts different stages of entity extraction. In natural language processing, some of the major tasks are performed. Initially, lemmatization process collects the unformatted data and groups it and identifies its dictionary form. Every word has some internal meaning and segmentation of sentence finds the intended POS

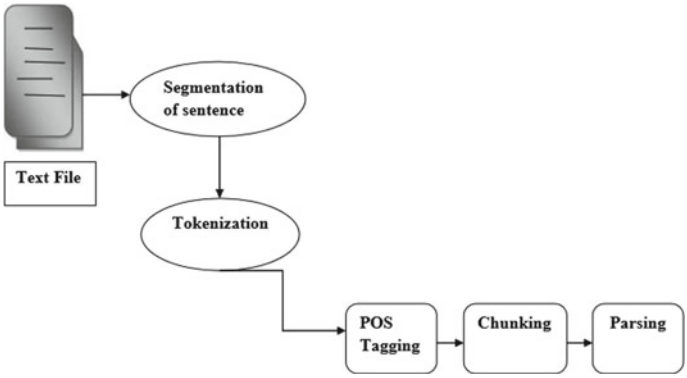


Fig. 2 General framework for natural language processing

and meaning of words in a complete sentence. Text segmentation is the process in which continuous text are separated into multiple words. It is quite easy for English language but in languages like Chinese, Thai and Japanese language sentences are written in completely different format. So text segmentation is the biggest challenge. Entity recognition also determines the proper names such as person location any institution. Entity extraction process takes a given text as an input and locate the words in a text belongs to which part-of-speech category. POS tagging is also called word categorization which removes all the ambiguities related to the text. It primarily reads the text in any specific language and assigns part-of-speech to each word as a noun, pronoun or verb. It also finds the relationship between inter related words and adjacent words in a sentence or paragraph. Some words in a paragraph or a sentence refers to more than one POS at altered times. It is not only important to identify the part-of-speeches, but classification plays a vital role in entity extraction process.

1.3 Named Entity Reorganization

In spite of the text format whether the text is in document record based, spreadsheet or web posts, the named entity extraction distinguishes it as a place-, person- or location-based entity. Some temporal and numerical expressions can also be derived from the text [4]. Several methods have been used for extraction of text of different domain. Regex extraction is suitable for email-address, credit card number and ID number extraction. Dictionary extraction basically uses dictionary of token sequences and identifies when those sequences occur in the text. Similarly statistical extraction is another method for extracting texts [5] (Fig. 3).

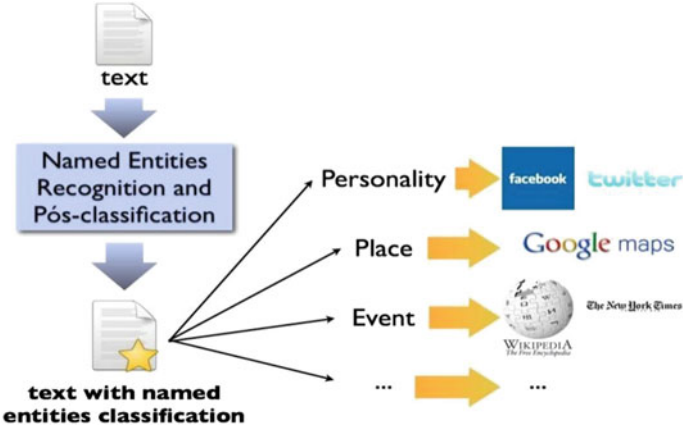


Fig. 3 Example of named entity extraction

2 Literature Review

See Table 1.

3 Proposed Framework

An enhanced entity extraction framework is shown in Fig. 4. This proposed framework is useful mainly on extracting useful and meaningful information from various categories of data. The Entity Extractor follows the following three steps as

1. Parts-of-speech trainee
2. POS Extractor
3. Classifier for POS.

Through the process of filtration, noun, verb, adverb and adjectives in a sentence can be determined. Entity extraction is a process of finding specific entity-based POS out of the raw data.

The system is quite effective in aspect of digitization, training and classifying with high performance. This system will significantly help user in accessing data irrespective of its veracity, volume.

Data can be collected from different internal and external sources. The data can be collected in two ways either by manually or by the process of digitization. The sources of manual data, simple text data, are in the form of one sentence or a whole paragraph or several pages. We can convert the manual data to the digital data or the digital data can be directly collected from the internet, social sites, different enterprises or media and educational institutes.

Table 1 Comparative analysis of entity extraction approaches

Author, year	Title	Proposed mechanism	Contribution	Challenges
Christian Bøhn and Kjetil Norvag	Extracting named entities and synonyms from wikipedia	Generic NE (named entity) recognition algorithm proposed, generalized vector space model	Approaches for using wikipedia to automatically build a dictionary of NEs and their synonyms. The intended usage of this dictionary is in search by helping the users find articles about the entity independent of which entity name is used in the article with high precision	An additional wikipedia structure and contents are developed to improve NE reorganization and categorization
Srinivasa Rao Kundeti, J Vijayananda, Srikanth Mujjiga, Kalyan M [6]	Clinical named entity recognition: challenges and opportunities	NER in the medical domain involves handling of a number of vital tasks such as identification of medical terms, attributes such as negation, severity, identification of relationships between entities and mapping terms in the document to concepts in domain specific ontologies	Clinical NER is a sub task of information extraction that seeks to extract and classify named entities in clinical report. Previous clinical history summarization to the doctor during subsequent patient visits	There are lots of cases where the medical report text is available as an image of typed text or free handwritten. In a typical medical use case, the radiologist investigates the diagnostic medical images and summaries them in the text format

(continued)

Table 1 (continued)

Author, year	Title	Proposed mechanism	Contribution	Challenges
Rahul Sharnagat [7]	Named entity recognition: a literature survey	Supervised methods, hidden Markov models, maximum entropy-based model, support vector machine model, conditional random field-based model	Various supervised learning techniques are proposed for improving NER. The NER system is improvised for Indian Languages using supervised and unsupervised methods. A fine grained entity tag set generated and state diagram is proposed for identifying entities	NER system for Indian language will be more efficient by using deep learning techniques
Nita Patil, Nita Patil, B. V. Pawar [8]	Survey of named entity recognition systems with respect to Indian and foreign languages	This paper presents a survey regarding NER research done for various Indian and Non-Indian language. The study and observations related to approaches, techniques and features required to implement NER for languages is reported	linguistic approach rules are designed by grammar expert with help of knowledge derived from language, observations of samples, dictionaries	More techniques and algorithms will be developed to deal different language structure

(continued)

Table 1 (continued)

Author, year	Title	Proposed mechanism	Contribution	Challenges
P. Sutheebanjard and W. Premchaiswadi [9]	Thai personal named entity extraction without using word segmentation or POS tagging	Front and rear context extraction module technique, module for removal of numeric, non alphabet such as parenthesizes and escaped characters	This paper proposes the method to extract Thai personal named entity without using word segmentation or POS tagging. The proposed method is composed of three steps. Firstly, pre-processing, this process is used to remove non alphabet such as parenthesizes and numerical. Then, personal named entity is extracted by using contextual environment, front and rear, of personal name. Finally, post-processing, a simple rule base is employed to identify personal names	Precision, measure and F-measures are computed more efficiently for better performance analysis and apply in different domain like business, medical and sports

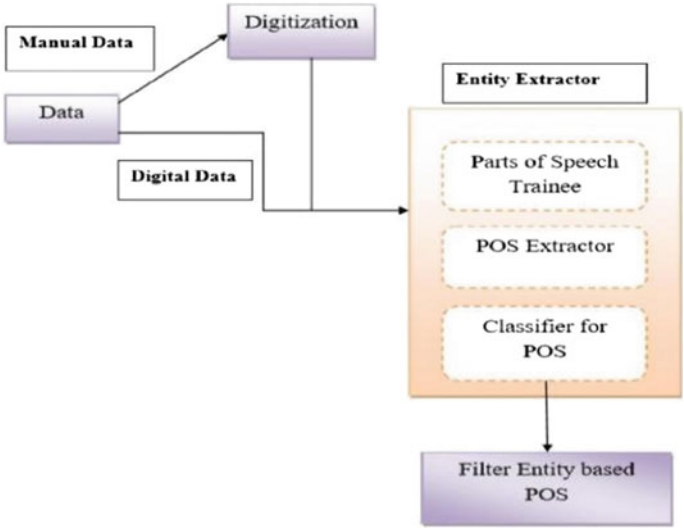


Fig. 4 Enhanced entity extraction framework

Collection of manual data and processing has advantage like accuracy that means an individual can identify and correct the errors while taking the data as input to the system, but it leads to several disadvantages like time consuming, expensive. So manual data need to be converted to digital format.

Digitization is the process through which we can represent data in digital form which assist computer processing and relevant operations. Digital data does not lose its quality even if it comes in all formats. Out of the enormous data, it is very much essential to collect only the relevant data. Digitization follows an appropriate way in collecting, transmitting and storing diverse data in form of text, image, sound or a simple recording.

Optical Character Recognition (OCR) is a highly developed technology which is useful in converting the manual documents in the form of handwritten papers into digital documents. The information can be easily stored and accessed by using OCR technology. Most of the organizations are now converting their printed manual business documents into digital form so that they can efficiently organize their relevant documents. OCR technology digitizes texts in fields like cognitive computing, text mining and pattern recognition. Optical word recognition and intelligent word recognition process make the digitization process much faster.

The Entity extractor system takes the digital data as an input. Then, it is the task of the trainer to train the part-of-speech. Part-of-speeches are noun, pronoun, adverb, conjunction and adverb etc. It will give much more information on a word and useful in determining entities whether refers to a person or organization in a specific digital text data. POS extractor is the way of assigning a part-of-speech to each individual word in a simple text. It is the process of categorizing whether a

word is a noun, pronoun or adverb. POS extractor followed by tokenization where we separate quotation and commas.

Classification of POS is an important task in enhanced entity extraction system. Words are classified into different types of form completely based on its functionality and utility. Classifier will differentiate nouns of different kind whether it's a proper noun, collective noun or common noun. Pronoun acts as a substitute for a noun so classifier has the capability to separate noun and pronoun. One of the part-of-speeches is an adjective. Adjective can detail the size and number of noun or pronoun. Verb plays an important role in a sentence. Sentence will have no sense without a verb so classifier will identify a word as a verb by visualizing its action performed in a sentence. Similarly, adverb has categorized on different aspect like manner, timing, place and degree. So classifier does the major task.

Filter entity-based POS is the last phase of enhanced entity extraction framework. In this stage, we only take entity-based part-of-speeches. Entities are filtered on the basis of different features.

As a single word in a text can have multiple sense which leads to ambiguity in the extraction process. Enhanced POS system can resolve numerous ambiguities like semantic, Pragmatic and syntactic through rule-based POS tagger.

In future work, we will be making our system more efficient so that it will extract meaningful content out of unstructured and semi-structured text.

4 Discussion

The named entity extraction is a better and advanced method for tracing a word or any phrase for finding a specific entity in text either in document or in paragraph format. Classification of entities to the appropriate part-of-speeches is the crucial part of entity extraction method. Entity extraction helps classifying different entities as a person, location and organizational name. POS tagger and classifier are effective and efficient tool for analyzing texts and find any relations if exists among them.

Different classification approaches are used to classify different POS as a noun, pronoun, verb, adjective, conjunction. Our enhanced entity extraction framework i n will efficiently work for python written code and Node Js for Unix environment which will extract meaningful content out of PDF and doc file of web blogs and semi-structure data, i.e., email by use of open-source library Textract.

In future work, we will use this framework for extracting technical skill, work experience and educational qualification of a particular student and filter out some words by using NLP and regular expressions function in finding the overall performance of any person and how capable that person is for any competitive examination.

References

1. M.K. Shilpa, Big data and methodology-a review. Int. J. Adv. Res. Comput. Sci. Software Eng. **3**(10), 991–995 (2013)
2. M. song et al., PKDE4J: Entity and relation extraction for public knowledge discovery. J. Biomed. Inf. **57**, 320–332 (2015)
3. N. Ranjan, A. Gupta, I. Dhumale, Text analytics and classification techniques for text document. IJDR **5**, 5953–5955 (2015)
4. D.D.A. Bui, G. Del Fiol, S. Jonnalagadda, PDF text classification to leverage information extraction from publication reports. J Biomed. Inf. **61**, 141–148 (2016)
5. C. Bøhn, K. Nørvåg, Extracting named entities and synonyms from wikipedia. *2010 24th IEEE International Conference on Advanced Information Networking and Applications (AINA)*. IEEE (2010)
6. S.R. Kundeti, J. Vijayananda, S. Mujjiga, M. Kalyan, Clinical named entity recognition: challenges and opportunities. *2016 IEEE International Conference on Big Data (Big Data)*. IEEE (2016)
7. R. Sharnagat, Named Entity Recognition: A Literature Survey. *Center For Indian Language Technology* (2014)
8. N. Patil, N. Patil, B.V. Pawar, Survey of named entity recognition systems with respect to Indian and foreign languages. Int. J. Comput. Appl. **134**(16) (2016)
9. P. Sutheebanjard, W. Premchaiswadi, Thai personal named entity extraction without using word segmentation or POS tagging. Natural Language Processing, 2009, SNLP'09, Eighth International Symposium on. IEEE (2009)

Sharing Organizational Data Outside Its Domain Using ABE in Cloud Environment

Reetu Gupta, Priyesh Kanungo and Nirmal Dagdee

Abstract In this era of Internet, users and organizations are sharing a lot of data quite frequently. The development of cloud technology has enabled the users and organizations to put their data online. Organizations are outsourcing their shared data on cloud storage systems for low maintenance cost and flexibility. The outsourced data may be sensitive, so it is stored in encrypted form. There are a large number of users with different privileges and authorities for data access according to their attributes and cadre in the organization. When the data owner or an organization shares its data, the user can be from the organization itself or from outside the organization, i.e., can be known user (closed-domain user) or unknown user (open-domain user). In this paper, we propose a framework of an access control system, which allows the data sharing to closed- as well as open-domain users depending upon their attributes issued by the data owner and various attribute authorities. This kind of system can be used to promote easy sharing of useful data with large number of users.

Keywords Cloud storage systems · Closed and open domain
Access control system · Attribute authorities

1 Introduction

In recent years, the information communication technology and internet-based services have become the backbone for the society. Various business organizations, healthcare organizations, research institutes, commercial bodies, etc., are producing

R. Gupta (✉) · N. Dagdee
S.D. Bansal College of Technology, Indore, India
e-mail: reetu.gupta@sdbct.ac.in

N. Dagdee
e-mail: director@sdbct.ac.in

P. Kanungo
MPSTME, NMIMS Shirpur, Shirpur, Maharashtra, India
e-mail: priyesh.kanungo@nmims.edu

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_9

a bulk of data everyday, and they are putting this data online for sharing and exchange of information. This type of data sharing provides many advantages such as higher productivity, flexibility, and low maintenance cost. Users of the data also get the required data at fast speed and low cost. Cloud systems are the obvious choice for data storage and sharing because of their advantages. Secure and efficient data sharing in cloud environment is a challenging issue [1, 2]. Data owner wants the outsourced data to be very secure. Even the party managing the data called Cloud Service Provider (CSP) is not considered as trusted. So the data is stored in encrypted form on the cloud. Also there is requirement of access control so that the data owner could specify the users to whom he wants to share the data. Various approaches have been proposed for access control [2, 3]. For cloud-based storage systems attribute-based access control approaches have been proposed [3–7], where data owner outsources his sharable data on cloud and defines his own access control policies over that data. Data owner may or may not be aware of the user's identity to whom he wants to share the data. When data owner knows the users or users are registered to the organization, that type of environment is called closed environment. When he does not know the identity of the users but he recognizes them based on their characteristics, that type of system is called open system. In case of organizational data sharing, a large number of users work under a domain and generally form a hierarchy governed by various administrative levels. There may be users who work in collaboration with the organization or who are not known to the organization, but need the shared data. So there is a need of development of access control methodologies for data sharing, which can be applied in open environment as well as closed environment. Attribute-Based Encryption (ABE) [8] is regarded as a promising cryptographic technology for providing fine grained access control. It allows data owner to frame the policies on encrypted data based on various attributes of the targeted user. If a user possesses sufficient attributes to satisfy the access policy, he can decrypt the encrypted data.

In this paper, we have proposed a framework of an access control system, which allows sharing of organizational data not only to the organizational users but also to the users who are interested in the data in open domain. The paper is structured as follows: Section 2 discusses related work in this area. Section 3 presents the problem formulation and outline of proposed access control system. In Sect. 4, system framework and functionalities are discussed, and Sect. 5 presents the security analysis of the proposed system. Finally, Sect. 6 draws the conclusion.

2 Related Work

Access control is one of the compulsory requirements for secure data sharing. It allows only the eligible users to access the data. The eligible users are the users who satisfy the access control policy stated by the data owner. Considerable work has been done for effective data access control on cloud-based data storage systems [3–7, 9, 10]. They make use of a cryptographic primitive of attribute-based access control (ABAC) known as attribute-based encryption (ABE). It is the technique to

provide fine grained access control over encrypted data, where access policies are defined by using various attributes. The two major categories of ABE are ciphertext policy ABE (CP-ABE) [8] and key policy ABE (KP-ABE) [11]. In the CP-ABE scheme [8], the access structure is defined by the data owner or encryptor of the data. This access structure is incorporated into the ciphertext. For example, a policy can be (Doctor AND (Physician OR Orthopedist)). The user can acquire the secret keys for his attribute set, e.g., Doctor and Physician. As these secret keys satisfy the ciphertext policy, the user can decrypt the data. In the KP-ABE scheme [11], the access structure is defined by the authorities and attached in the user's secret key. The encrypted data called ciphertext is bundled with a set of attributes. A user can decrypt the ciphertext if and only if the access structure in his secret key satisfies the mentioned attributes in the ciphertext.

In all above approaches, single trusted authority (TA) was used, which cannot maintain the large numbers of users in an open distributed environment. A user may get the attributes from multiple authorities of multiple domains and the data owner can use various attributes from these authorities in his encryption. To solve the problem of multi-authority ABE, a number of access control schemes have been proposed in [6, 7, 12–14]. The solution proposed by Chase [12] allows data owner to specify a number of attributes issued by multiple attribute authorities (AA). In this scheme, a central authority (CA) masters the multiple AAs. The CA holds the master key of the system, so it can decrypt all the ciphertexts. This creates problem of vulnerability attacks and may become performance bottleneck for large systems. To address these drawbacks, various schemes without the existence of central authorities were suggested. Lewko-Waters [15] proposed decentralized ABE, where the concept of global identifier (GID) was used to replace the CA. The use of GID was to link the keys issued to same user by different authorities. Due to presence of hierarchical structure of user's role and attributes in modern organizations hierarchical attribute-based encryption (HABE) [16] was proposed. This introduced flexibility, high performance, delegation capabilities, and scalability. In multi-authority systems, attributes issued to user may expire or he can gain new attributes. This situation changes the user's current access permissions. Various revocable multi-authority schemes have been proposed in [6, 7, 14]. All these approaches addressed the issues of key management, revocation of access rights, flexibility in policy management, etc., but the limitation is to address a large group of users from both closed domain and open domain. In this paper, we introduced a framework for hybrid access control system which addresses these shortcomings and enables us to build an access control supporting open- and closed-domain users.

3 Problem Formulation

Organizations share their data to various users for various reasons. E-healthcare organizations can share the data with other organization or individual for patient's treatment. Research institutes can share the data to promote the use of new findings

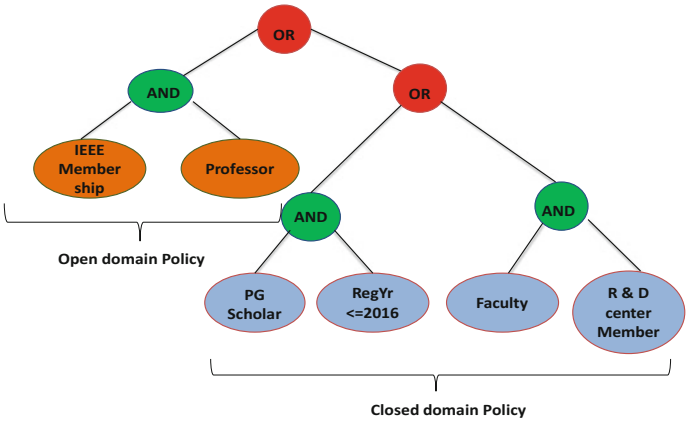


Fig. 1 Example access control policy

and to further advance the research. Business organizations can share the data to their collaborators to provide fast service to the consumers. The organization's data owner can frame the access control policy by using the attributes of the closed-domain users as well as the open-domain users. This hybrid nature will expand the scope of data sharing. The closed-domain users can be authenticated by the organization or data owner itself. The open-domain users get their attributes for their characteristics from various domain authorities. To further illustrate the concept, we take an example of research data sharing by a professor. The professor wants to share his data to a large group of users to benefit them. He shares the data by the name of the organization he is working with. The data can be shared with following users who are either the PG scholars of the organization who have registered in the year before 2016 or the faculty members of the organization who are deputed in the R&D centre of the organization or all the professors who are having IEEE membership. Figure 1 depicts the access control policy.

The closed-domain authorities are the registration office and the administrative section of the organization. The authorities in the open domain are any university distributing credential of professor and the IEEE membership office. A flexible and scalable access control system is needed to implement varied kind of requirements for such a data sharing system in multi-domain environment.

4 System Framework and Functionalities

In this section, we present the framework of access control system and system functionalities. The framework is shown in Fig. 2.

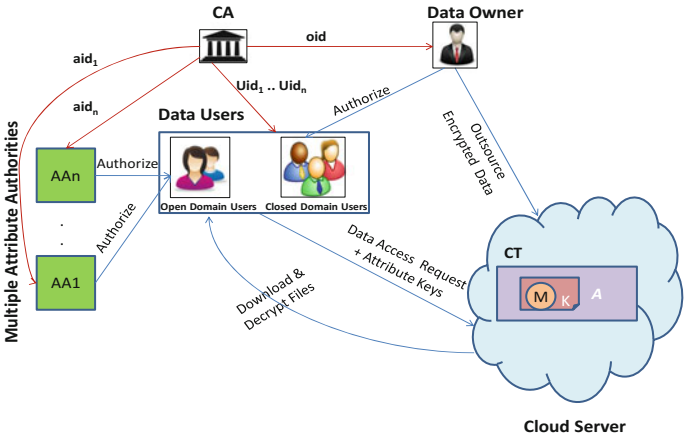


Fig. 2 Framework of access control system for data sharing

4.1 System Framework

There are four entities which are involved in this system. The central authority (CA) sets up the system and assigns unique aid to attribute authorities, uid to users and oid to data owner. The attribute authorities (AA) are responsible for key management tasks. The data owner (DO) outsources the data with access control policies at cloud servers so as to enable the access to eligible users only. DO also authenticates the closed-domain users by issuing them suitable attributes according to their role or identity in the organization. Cloud server stores the encrypted data and performs appropriate searches when required. The data user submits request to retrieve the file stored at cloud server. They can be from closed domain or open domain. Closed-domain users are identified and authenticated by data owner, and open-domain users are issued attributes from various attribute authorities.

4.2 System Functionalities

In order to provide a hybrid access control system, we propose a scheme that is based on multi-authority ABE [12, 13]. We combine it with the scheme where CP-ABE [8] is used by data owner to assign secret key to organization's user based on his attributes and identity. Here we assume that the CA is fully trusted party. The attribute authorities are also trusted. They never collude with any of the users. The system consists of following steps:

4.2.1 CA Setup

The algorithm for the setup of the system is run by CA. It takes input parameter λ. The CA chooses two multiplicative groups G and G_T of prime order p. The generator of G is g and $\boldsymbol{e} : \boldsymbol{G} \times \boldsymbol{G} \rightarrow \boldsymbol{G_T}$ is a bilinear map. Let H :{0, 1}* → G be a collision-resistant hash function, which maps an attribute to a random group element. The global parameters of system are {p, $\boldsymbol{g}$, G, G_T, e, H}.

CA also registers all the attribute authorities with unique ids 'aid' and also assigns a unique 'oid' to data owner/organization.

4.2.2 AA Setup

Each AA takes global parameters and the attribute set U_{aid} as input to setup algorithm. The AA chooses a random variable $x_{aid} \in Z_p^*$ and computes $e(g, g)^{x_{aid}}$. Then it chooses a random number $y_{aid_{i,j}} \in Z_p^*$ for each attribute and computes $g^{y_{aid_{i,j}}}$. The public key of AA isw $\{e(g, g)^{x_{aid}}, g^{y_{aid_{i,j}}}\}$ and secret key is $\{x_{aid}, y_{aid_{i,j}}\}$.

4.2.3 DO Setup

The data owner with identifier oid works same as one of the attribute authority. It does have a set of attributes U_c. The public key and secret key for data owner are generated as: $\{e(g, g)^{\alpha_{oid}}, g^{\beta_{oid_k}}\}$ and $\{\alpha_{oid}, \beta_{oid_k}\}$.

4.2.4 Data Encryption

The data owner encrypts the data m under the access policy $\mathbb{A}$. The inputs to the algorithm are the public keys of AA and the data owner. The data owner randomly chooses $s \in Z_p^*$ and creates ciphertext $CT_{\mathbb{A}} = (\mathbb{A}, C_1, C_2, C_3)$. Here C_1, C_2, C_3 are semi-functional ciphertexts components, which are used in decryption process for getting the original data m by canceling out each other.

$$C_1 = m \cdot ((\prod_{i=1}^{na} (\prod_{j=1}^{no} e(g, g)^{x_{aidi,j}})) \cdot (\prod_{k=1}^{nc} e(g, g)^{\alpha_{oid,k}}))^s \tag{1}$$

where

n_a is the attribute authorities involved in access policy $\mathbb{A}$,
n_o is the number of attributes managed by each AA,
n_c is the number of attributes managed by data owner

$$C_2 = g^s \tag{2}$$

$$C_3 = \left(\left(\prod_{i=1}^{\text{na}} \left(\prod_{\text{j}=1}^{no} g^{y_{aid_{i,j}}} \right) \right) \cdot \left(\prod_{k=1}^{nc} g^{\beta_{oid_{\text{k}}}} \right) \right)^{s} \tag{3}$$

4.2.5 Key Generation

(a) **For Open-Domain User**: Each user gets a global unique identifier uid from the CA. The user gets various attribute keys issued from AAs after authentication. The user's secret key is in the form $\{g^{x_{aid_i}} \cdot H(uid)^{\text{y}_{\text{aid}_{\text{i,j}}}}\}$.

(b) **For Closed-Domain User**: Here the user gets the secret keys issued from data owner oid in the form $\{g^{\alpha_{oid}} \cdot H(uid)^{\beta_{oid_j}}\}$.

4.2.6 Data Decryption

At the receiver side, the user runs the decryption algorithm with his unique uid. Under the access policy $\mathbb{A}$, if the user has sufficient secret keys either from the closed or open domain, then $\mathbf{C1} \cdot \mathbf{e}(\mathbf{H}(\mathbf{uid}), \mathbf{C3})$ cancels out from $\boldsymbol{e}(\mathbf{C2}, \mathbf{SK_{OD}})$ or from $\mathbf{e}(\mathbf{C}2, \mathbf{SK_{CD}})$, and we get the original data m. Here SK_{OD} and SK_{CD} are the secret keys obtained by the user from open or closed domain.

$$\text{SK}_{\text{OD}} = \prod_{t=1}^{no} \text{SK}_{\text{aid}} \tag{4}$$

$$\text{SK}_{\text{CD}} = \prod_{p=1}^{nc} \text{SK}_{\text{oid}} \tag{5}$$

After decryption data m is retrieved as:

$$\text{m} = \frac{\text{C1} \cdot \text{e(H(uid), C3)}}{\boldsymbol{e}(\text{C2, SK}_{\text{OD}})\text{e(C2, SK}_{\text{CD}})} \tag{6}$$

5 Security Analysis of Proposed System

The proposed system guarantees the confidentiality of outsourced shared data against CA, AA, cloud server, and unauthorized users. There is no collusion attack in the system, which can be understood as follows.

In our system, CA issues aid, oid and uid to AA, owner, and user, respectively, but cannot gain access to any part of the attribute secret keys. The attribute authorities are also distinguishable because of the use of independent aid in generation of secret keys. Cloud server cannot access the stored data as it is not possible for server to get the secret keys which are in the form of $\left\{g^{x_{aid_i}} \cdot H(uid)^{\text{y}_{\text{aid}_{\text{i,j}}}}\right\}$ and $\{g^{\alpha_{oid}} \cdot H(uid)^{\beta_{oid_j}}\}$.

Any unauthorized user cannot get the access to shared data as he/she cannot provide all the secret attribute keys corresponding to required attributes for calculating $e(g, g)^s$. Two or more users, who are not allowed to access the data individually, cannot collude for getting access because different users have different e(H(uid), g) values, which are further used in the process of decryption.

6 Conclusions

In this paper, a framework for an access control system is been proposed. The access control system allows sharing of organizational data not only to the organizational users but also to the users who are interested in the data in open domain. The system permits the access to the data based on the attributes issued by multiple attribute authorities as well as the attributes issued by the data owner. Under the assumption that the CA and the AAs are fully trusted parties, we have designed the entire scheme that includes setup of the CA and the AAs, key generation of users in closed and open domains, and the encryption and decryption algorithms. We conclude that the proposed ABE scheme will be beneficial for an organization to realize proper control of access to its data by its internal users and at the same time allow easy access of the useful data selectively to the users outside the organization based on the attributes they hold.

References

1. J. Tang, Y. Cui, Q. Li, K. Ren, J. Liu, R. Buyya, Ensuring security and privacy preservation for cloud data services. ACM Comput. Surv. **49**(1), 13:1–13:39 (2016)
2. A. Gholami, E. Laure, Security and privacy of sensitive data in cloud computing: a survey of recent developments, in *Seventh International Conference on Network & Communications Security (NCS 2015)* (2015), pp. 131–150
3. X. Dong, J. Yu, Y. Luo, Y. Chen, G. Xue, M. Li, Achieving an effective, scalable and privacy-preserving data sharing service in cloud computing. Comput. Secur. **42**, 151–164 (2014)
4. D. Thilakanathan, S. Chen, S. Nepal, R. Calvo, SafeProtect: controlled data sharing with user-defined policies in cloud-based collaborative environment. IEEE Trans. Emerg. Top. Comput. **4**(2), 301–315 (2016)
5. S. Ruj, M. Stojmenovic, A. Nayak, Decentralized access control with anonymous authentication of data stored in clouds. IEEE Trans. Parallel Distrib. Syst. **25**(2), 384–394 (2014)
6. K. Yang, X. Jia, K. Ren, B. Zhang, DAC-MACS: effective data access control for multi-authority cloud storage systems, in *2013 Proceedings IEEE INFOCOM*, Turin (2013), pp. 2895–2903
7. Q. Li et al., Secure, efficient and revocable multi-Authority access control system in cloud storage. Comput. Secur. **59**, 45–59 (2016), https://doi.org/10.1016/j.cose.2016.02.002
8. J. Bethencourt, A. Sahai, B. Waters, Ciphertext-policy attribute-based encryption, in *Proceedings of the 2007 IEEE Symposium on Security and Privacy* (2007), pp. 321–334
9. R. Wu, G.J. Ahn, H. Hu, Secure sharing of electronic health records in clouds, in *8th International Conference on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom)*, Pittsburgh, PA (2012), pp. 711–718

10. M. Li, S. Yu, Y. Zheng, K. Ren, W. Lou, Scalable and secure sharing of personal health records in cloud computing using attribute-based encryption. IEEE Trans. Parallel Distrib. Syst. **24**(1), 131–143 (2013)
11. V. Goyal, O. Pandey, A. Sahai, B. Waters, Attribute-based encryption for fine-grained access control of encrypted data, in *Proceedings of the 13th ACM Conference on Computer and Communications Security* (2006), pp. 89–98
12. M. Chase, Multi-authority attribute based encryption, in *Proceedings of the 4th Conference on Theory of Cryptography* (2007), pp. 515–534
13. M. Chase, S.S.M. Chow, Improving privacy and security in multi-authority attribute-based encryption, in *Proceedings of the 16th ACM Conference on Computer and Communications Security* (2009), pp. 121–130
14. K. Yang, X. Jia, Expressive, efficient, and revocable data access control for multi-authority cloud storage. IEEE Trans. Parallel Distrib. Syst. **25**(7), 1735–1744 (2014)
15. A. Lewko, B. Waters, Decentralizing attribute-based encryption, in *Advances in Cryptology* (Springer, Berlin, Germany, 2011), pp. 568–588
16. G. Wang, Q. Liu, J. Wu, Hierarchical attribute-based encryption for fine-grained access control in cloud storage services, in *Proceedings of the 17th ACM Conference on Computer and Communications Security* (2010), pp. 735–737

Improving the Cloud Server Resource Management in Uncertain Load Conditions Using ACO and Linear Regression Algorithms

Nikita Baheti Kothari and Ajitab Mahalkari

Abstract The cloud is new generation computing technology. The technology is a huge infrastructure that is developed using networks, computational resources and memory units. These infrastructures are not working alone they are always shared with the other cloud infrastructures. Therefore these services can deal with the fluctuating loads on servers. Additionally, scaled when more and more resource requirements appears. Therefore any of the servers can face the issue of resource availability any time due to lake of computational resources. This event in cloud infrastructure is also termed as the uncertain load appearance because the significant amount of load appeared and to execute these request less amount of resources are available by which the waiting time of the jobs increases and server running cost also increases. In order to deal with such kind of situation, the proposed work introduces a two-phase scheduling technique that helps to monitor the load appearance and based on the load patterns optimization of resource allocation is performed. In this context, two popular algorithm namely regression analysis and ACO (ant colony optimization) algorithms are applied. The simulation and modeling of the proposed approach is performed on CloudSim simulator. The experiments of the given technique demonstrate the efficient and enhance resource management.

Keywords VM scheduling · Load balancing · Resource allocation · Processes Cloud simulation · ACO algorithm

1 Introduction

Cloud computing becomes popular due availability, reach ability and scalability of the resources when the application required [9]. Nowadays, most of the applications are developed for targeting a large number of worldwide users additionally the traffic

N. B. Kothari (✉) · A. Mahalkari
Department of Computer Science and Engineering,
S.D. Bansal College of Technology, Indore, Madhya Pradesh, India
e-mail: nikitabaheti89@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_10

on these applications are not in regular states, sometimes more and more resources are required for execution of these applications. In this context, the cloud is a reliable and effective technology for hosting the data and applications [8]. On the other side, these computing infrastructures are not working individually to serve the dynamic services; they are also collaborated or shared with some other providers to scale their abilities. In this scenario, when a server consumes the resources from the other server it is called as rent of resource. The rents of resources are increasing the running cost of servers because of additional overhead on the running processes. The processes waiting time is increases in both the cases of the cloud servers are much effective indicates low-performance ability of servers. Therefore, the resource management under such conditions is required when the load appeared uncertainly.

Basically, the load on servers is not fluctuated enough most of the time. It is observed in regular limits but sometimes due to instantly increment in popularity of an application or other factors, the workloads increases. Due to this, the consumption of resources becomes uncertain and this condition of workload appearance is needed to be deal for improving the cloud server performance. A new predictive and optimization technique that first finds the possible upcoming load on server and according to the load, the resources is managed with the help of ACO algorithm is proposed for design and implementation.

2 Literature Survey

The given section introduces the different techniques and methods that are recently developed for optimizing the solutions for effective cloud resource provision algorithm and methodology for optimization of resource allocation. These techniques help to develop an effective methodology for resource allocation strategies.

Tchernykh et al. [1] address non-preemptive scheduling issues on hetero-generous P2P frameworks. The Main focus is to set replication thresholds, and dynamically adapt them to cope with different objective preferences, workloads, and grid properties. Author considered three groups of strategies: knowledge-free, speed-aware, and power-aware that show that two replicas for knowledge-free algorithms and one replica for speed-aware algorithms provide the best energy and performance trade-offs in the scheduling. In this paper, online scheduling problem without complexity can be reduced. Tchernykh et al. [2] centers around the bi-objective trial examination of web-based scheduling for the infrastructure as a service model of cloud processing. In this paper, various service levels are available for the client and each service level is having a relation with slack factor and cost for every unit of job execution time that find out the maximal time span to convey the computing resources for requested amount. This scheduling algorithm performs well in various situations with an assortment of workloads and cloud setup.

Schwiegelshohn and Tchernykh [3] address scheduling issues for infrastructure as a service (IaaS). In this model, each a service level is depicted by a slack factor and a cost for a handling time unit. In the event, that the provider acknowledges work

it is ensured to complete by its due date, that is its accommodation time in addition to its preparing time times the slack factor of relegated services level. After a job has been presented, the provider must choose instantly and irreversibly whether he acknowledges or rejects the job. Seenivasan [4] has proposed virtual machine repository (VMR) for cloud storage to such an extent that the issues of over and under provisioning can be settled to a more noteworthy degree. It reduces the time and cost of clients.

Chen et al. [5] have proposed and implemented proactive and reactive scheduling (PRS) algorithm for addressing the issue of minimizing uncertainty propagation in real-time workflow scheduling. Initially, author presented an uncertainty-aware scheduling architecture to alleviate the effect of uncertainty factors on the nature of workflow schedules. After that by using PRS algorithm, authors conducted extensive experiments using real-world workflow. Exploratory outcomes demonstrate that PRS algorithm beats two representative scheduling algorithms as far as costs, resource utilization and deviation. Jamshidi et al. [6] discussed cloud-based software that utilizes fuzzy logic for empowering subjective determination of elasticity rules and control hypothetical approach that utilizes type-2 fuzzy logic frameworks to deliberate elasticity under uncertainties. After performing lot of experiments, Authors concluded that cloud-based software aided with such elasticity controller can powerfully deal with surprising spikes in the workload and give adequate client encounter. This converts into expanded benefit for the cloud application proprietor.

Kumar and Vadhiyar [7] discussed production parallel systems in which submitted job has to stay before execution. Anticipation of waiting time helps in estimating to user and metaschedulers for making decision. In this paper, an integrated system is proposed that distinguish and foresee quick starters by utilizing the job qualities and conditions of the queue and processor occupancy and utilize the current methodologies to anticipate occupations with long queue holding up times. Proposed prediction methodologies can prompt right recognizable proof of up to 20 times more quick starters and give more tightly limits to these jobs, and in this manner result in up to 64% higher general forecast accuracy than existing techniques.

3 Proposed Work

The proposed work performed for enhancing the cloud resource management. In this scenario, a new model for resource provisioning is proposed that is intended to improve the resource allocation strategy.

3.1 System Overview

The proposed work is a cloud resource scheduling technique that optimizes the resource consumption for improving the performance of server. The cloud servers

manage their resource dynamically when the servers are less loaded then they provide the computational resources to other servers and when they are high loaded then they rent the resources from other servers. But the renting of resources is much costly therefore needs to find a methodology that prepares the resources for upcoming load scenarios. The key hypothesis of the proposed work is to analyze the historical workload and find the different pattern that demonstrate how and when the load on the cloud servers are fluctuating. These patterns are compared with the real workload appeared on the cloud server for monitoring the load patterns. That process is an alert for upcoming workload. On the other side for managing the upcoming workload, soft computing technique is employed that reschedule the workload according to the new scenarios.

In this situation, a workload file is required as the basic input for the system. The proposed algorithm extracts the loads available in the historical database. In order to analyze the workload with the current appeared load and statistical model, linear regression analysis technique is employed that accept the previous load and new load patterns for finding the next possible load on server. Now the available resources are rescheduled according to the new generated pattern of the possible workload. In order to find the most suitable work assignment schedule, the ACO (ant colony optimization) algorithm is employed. The proposed approach is promising for optimizing the performance of the server when the load fluctuations of servers are higher.

3.2 Proposed Methodology

The proposed model consist of two major module first is used for evaluation of the workload data for making futuristic decisions and second is the resource allocation technique that improves the performance of existing scheduler. Both the concepts are described as follows:

Predictive load analysis. The predictive load analysis and model for obtaining the fluctuating patterns from data is described in Fig. 1.

Historical workload: This is the initial and key input of the proposed cloud resource management system. In this phase, a previous workload trace file is produced as input

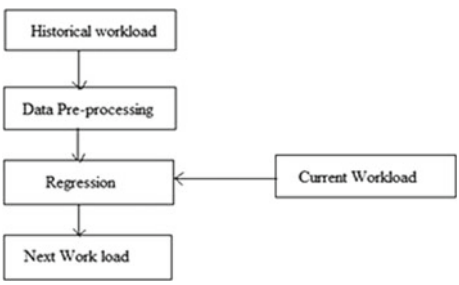


Fig. 1 Proposed predictive model

that contains a significant amount of data historical workload. That data is available in raw format which is processed and in require attributes form the data is extracted. This data is consumed in next phase to refine the data.

Data pre-processing: The data pre-processing is performed for improving the quality of data. In this context, the different approaches of data cleaning and refinement are applied on data. In this presented work, the aim is to extract the workload data from the trace file in addition of that removal of the instances that contains 0 workload or null workload. This extracted data is preserved in a matrix to perform the regression analysis.

Current workload: The current limited patterns are also collected in a matrix which is similar to the size of the historical workload. Using these two matrixes, the regression is performed for finding the next expected outcome as the prediction of upcoming load pattern.

Next workload: That is the workload value predicted by the system which tends to occurred.

Resource scheduling. The above phase is used to produce a value which notes the load which is occurred in next step. Therefore using this value, system is prepared to make some decision. Let the obtained predicted value from the previous step is P_r. Now need to approximate a threshold value which is used to make decision. Therefore, the maximum workload on list and minimum workload on list is computed first and denoted as max and min. Using both the values, a threshold is computed which is denoted as T. Therefore,

$$T = \frac{max - min}{N}$$

where, the N is the number of sample in list.

If the $P_r < T$, then the load in under wearable condition and no rescheduling of job is required otherwise if $P_r \geq T$, then the jobs in queue are again rescheduled using ACO algorithm. In this context, the current workload list available in job queue is rescheduled by including an additional load which is predicted by the system. Therefore, the current job list is extended to the size of J + 1 and the resource list is kept similar to the normal conditions. This job list is provided to ACO algorithm for finding the best matched resource combination to execute the jobs. The working of ACO algorithm is described as:

1. Randomly position the ants on N solutions
2. For i = 1 to N
    a. For ant m = 1 to M
        i. Each ant build solution by comparing the resources to the jobs
        ii. Select new load state according to steps
        iii. Apply local update
    b. End for
3. Apply global update using best ants
4. End for

Table 1 Proposed algorithm

Input: historical load dataset H, current load patterns C Output: scheduled jobs SJ
Process: 1. $H_m = read\ Load\ Dataset(H)$ 2. $P = preprocess\ Data(H_m)$ 3. $P_r = Regression.Predict(P, C)$ 4. $min = FindMinimum(C)$ 5. $max = findMaximum(C)$ 6. $T = \frac{max-min}{N}$ 7. $if\ (T > P_r)$ a. *do nothing* 8. Else a. $C = C + P_r$ 9. *end if* 10. JS $= ACO.getOptimal(C, R_S)$ 11. Return JS

3.3 Proposed Algorithm

This section combines both the efforts for providing the optimum scheduling for the cloud resource management. Table 1 contains the algorithm steps.

4 Result Analysis

This work provides evaluation and performance analysis of the proposed and traditional algorithm for their impact on the resource management. Additionally, a comparative performance analysis using essential performance factors are reported.

4.1 CPU Utilization

In order to execute an individual process for that an amount of time is consumed. This time of execution is termed as CPU Utilization. The CPU Utilization time can be calculated by using following formula:

$$\text{CPU}_{\text{ut}} = 100\% - (\%\ \text{of time spent in idle task})$$

where, CPU_{ut} = CPU Utilization Time.

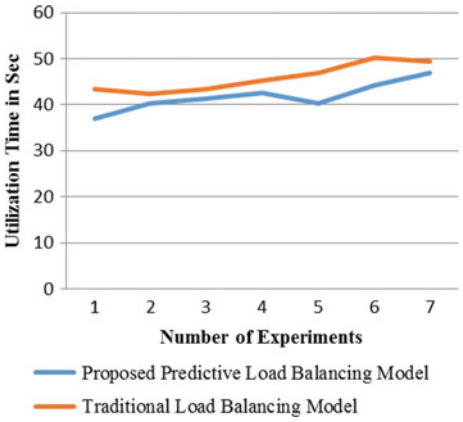


Fig. 2 Comparative CPU utilization

Table 2 Form of CPU utilization

Number of experiments	Proposed predictive load balancing model	Traditional load balancing model
1	37.0412	43.2865
2	40.3215	42.3362
3	41.3321	43.3691
4	42.6591	45.2261
5	40.3691	46.9856
6	44.2549	50.2132
7	47.0021	49.2862

Figure 2 and Table 2 show the CPU utilization time of both implemented algorithm, i.e., proposed HBB and traditional HBB of load balancing.

4.2 Execution Time

The execution time sometimes also called the runtime can be defined as the amount of time required during a program is executing, in context to other program life cycle such as compilation time, and/or load time. The traditional and proposed technique of load balancing in cloud simulation is demonstrated in Fig. 3 and Table 3.

4.3 Waiting Time

The waiting time is the amount of time required to schedule the job for execution after placing the job execution request in the waiting queue. The waiting time of the

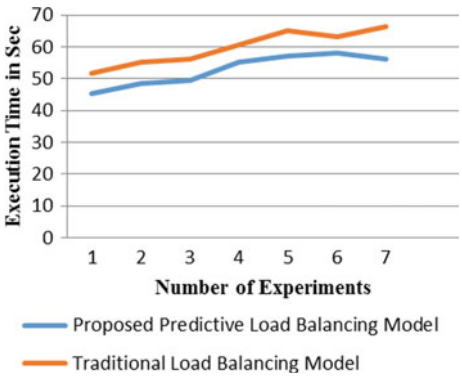


Fig. 3 Comparative execution time

Table 3 Execution time

Number of experiments	Proposed predictive load balancing model	Traditional load balancing model
1	45.332	51.621
2	48.561	55.261
3	49.336	56.002
4	55.214	60.519
5	57.012	65.221
6	58.233	63.221
7	56.147	66.321

Table 4 Waiting time

Number of experiments	Proposed predictive load balancing model	Traditional load balancing model
1	21.25	30.22
2	20.33	37.02
3	24.21	38.26
4	26.45	43.86
5	25.17	44.21
6	27.33	49.06
7	28.53	55.79

process execution is reported in this section, Fig. 4 shows the waiting time of the processes using the traditional approach of load balancing and proposed predictive schemes. Table 4 shows the numeric values of proposed technique.

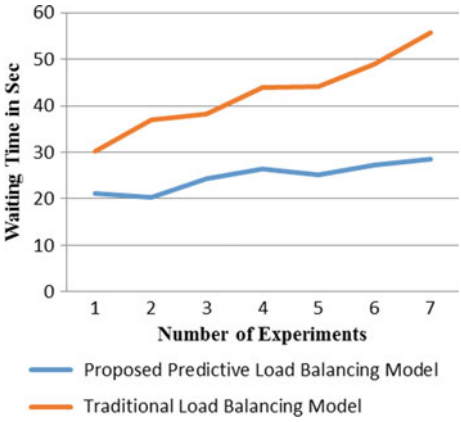


Fig. 4 Comparative waiting time

5 Conclusion

The cloud computing offers the elastic services of computing and storage therefore these computational resources are in higher demands. In addition to that these services are reliable for providing the resources to application at any point of time when the resources are required. The proposed technique first analyzes the load patterns which are previously occurred on different situations. Further, the currently load scenarios are also considered for finding the current pattern matched with the previous patterns. In this context, a linear regression model is implemented for finding the approximate workload in next step. After computing the approximate upcoming work load on server, the decision threshold value is calculated. Using this threshold value, the decision is made to reschedule the task list or not. If the rescheduling of task is called ACO (ant colony optimization), algorithm is called for process the resource list with respect to the job list.

The proposed technique is implemented using JAVA technology-based simulation system which is popularly known as CloudSim simulator. After the implementation of proposed technique for resource management, the performance of the system is computed in different performance factors these are reported in Table 5.

The computed results of both the system namely proposed and traditional available resource scheduling systems demonstrate the proposed technique is efficient and provides higher outcome with respect to the traditional methodology. Therefore, the proposed approach is acceptable for resource scheduling task in cloud computing.

Table 5 Computed performance

S. No.	Parameters	Proposed technique	Traditional technique
1	Waiting time	Low	High
2	Execution time	Low	High
3	CPU utilization	High	Low

References

1. A. Tchernykh, J.E. Pecero, A. Barrondo, E. Schaeffer, Adaptive energy efficient scheduling in Peer-to-Peer desktop grids. Future Gener Comput Syst **36**, 209–220 (2014)
2. A. Tchernykh, L. Lozano, J.E. Pecero, S. Nesmachnow, Bi-objective online scheduling with quality of service for IaaS clouds, in *2014 IEEE 3rd International Conference on Cloud Networking (CloudNet)*, pp. 307–312
3. U. Schwiegelshohn, A. Tchernykh, Online scheduling for cloud computing and different service levels, in *2012 IEEE 26th International Conference In Parallel and Distributed Processing Symposium Workshops & PhD Forum (IPDPSW)*, pp. 1067–1074
4. D. Seenivasan, Optimization of resource provisioning in cloud. Int. J. Comput. Appl. (IJCA) 14–16 (2014)
5. H. Chen, X. Zhu, D. Qiu, L. Liu, Uncertainty-aware real-time workflow scheduling in the cloud, in *2016 IEEE 9th International Conference on Cloud Computing (CLOUD)*, pp. 577–584
6. P. Jamshidi, A. Ahmad, C. Pahl, Autonomic resource provisioning for cloud-based software, in *Proceedings of the 9th International Symposium on Software Engineering for Adaptive and Self-Managing Systems* (ACM, 2014), pp. 95–104
7. R. Kumar, S. Vadhiyar, Identifying quick starters: towards an integrated framework for efficient predictions of queue waiting times of batch parallel jobs, in *JSSPP* (2012), pp. 196–215
8. A. Tchernykh, U. Schwiegelsohn, E. Talbi, Towards understanding uncertainty in cloud computing resource provisioning. Procedia Comput Sci **51**, 1772–1781 (2015)
9. K. Kaur, A review of cloud computing service models. Int. J. Comput. Appl. (IJCA) **140**(7) (2016)
10. Kelvin, Basic Overview on Cloud Computing, https://www.hostdepartment.com/blog/2014/08/05/cloud-computing/. Accessed 23 Oct 2017
11. D. Barley, Cloud Computing Effect on Enterprises-in terms of Cost and Security, Lund University, Jan 2011, Number 1 (2011)
12. What is cloud computing? A beginner's guide, https://azure.microsoft.com/en-in/overview/what-is-cloud-computing/
13. The Benefits of Cloud Computing, https://www.ibm.com/ibm/files/H300444G23392G14/13Benefits_of_Cloud_Computing_634KB.pdf, Dynamic Infrastructure, July 2009
14. D. Karger, C. Stein, J. Wein, *Scheduling Algorithms. Algorithms and Theory of Computation Handbook: Special Topics and Techniques* (Chapman & Hall/CRC, 2010)
15. D. Magalhães et al., Workload modeling for resource usage analysis and simulation in cloud computing. Comput. Electr. Eng. **47**, 69–81 (2015)
16. Resource Provisioning, https://www.tmforum.org/Browsable_HTML_Frameworx_R14.5/main/diagram70031aa2d7f211db943e000e3573f0d3.htm
17. B.H. Bhavani, H.S. Guruprasad, Resource provisioning techniques in cloud computing environment: a survey. Int. J. Res. Comput. Commun. Technol. **3**(3) (2014)
18. U. Sharma, P.J. Shenoy, S. Sahu, A. Shaikh, A cost-aware elasticity provisioning system for the cloud, in *Proceedings of the International Conference on Distributed Computing Systems*, July 2011, pp. 559–570
19. K. Tsakalozos, H. Kllapi, E. Sitaridi, M. Roussopoulous, D. Paparas, A. Delis, Flexible use of cloud resources through profit maximization and price discrimination, in *Proceedings of the 27th IEEE International Conference on Data Engineering (ICDE 2011)*, Apr 2011, pp. 75–86
20. R.F. de Mello, L.J. Senger, L.T. Yang, A routing load balancing policy for grid computing environments, in *20th International Conference on, Advanced Information Networking and Applications*, vol. 1 (2006)

Comparative Performance Evaluation Using Hadoop Ecosystem –PIG and HIVE Through Rendering of Duplicates

Pragya Pandey and C. S. Satsangi

Abstract Traditionally, for analysis and decision making, preprocessed data have been stored on data warehouse and various operations are performed on those stored data. With the rapid growth in cloud applications and IoT-based systems, data get generated with high velocity and increased volume. Thus, big data, which get generated by variety of structured and unstructured data sources, are heterogeneous. There is a need to integrate variety of data and analyze the large-scale data. Hadoop provides a solution for such processing needs. Inherently, it is designed for high-throughput batch processing jobs and for handling complex queries for streaming data. This paper presents the MapReduce model of Hadoop framework with two analytical ecosystems PIG and HIVE. Here, we also present performance evaluation for each category like processing time for some queries executed on Pig and Hive while combining two healthcare datasets, gathered from different data sources. Comparative analysis has also been done and is presented in this paper.

Keywords Warehousing · IoT · Big data · Hadoop · MapReduce · Hive · Pig

1 Introduction

Distributed computing [1] plays vital role in management of complex data. Big data characteristics data volume, variety, veracity and velocity brings the challenge to store this huge data with all complexity of processing of data to understand the data turn it into competitive advantage. Many cloud- and IoT [2]-based applications like social media Tweeter, financial domain banking, Web search engines generate the data with increased size. Open source Hadoop [3] fills a gap by effectively storing

P. Pandey (✉) · C. S. Satsangi
Department of Computer Science and Engineering,
Medicaps University, Indore, India
e-mail: ppragya2009@gmail.com

C. S. Satsangi
e-mail: cssatsangi@yahoo.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_11

and providing platform for computation in substantial time. Example of Hadoop ecosystem software includes MapReduce [4], Spark SQL [5], Apache Hive, Apache Pig, HBase [6] and so on (Fig. 1).

Open-source system provides set of APIs and hides the complexity of big data execution by splitting it by parallel task and makes the system fault tolerant for user. Various tools such as Hive [7, 8], Pig [9, 10] are analyzing large datasets in a high-level language and computation is done on top of Hadoop.

Apache Hive is the easiest to use of the high-level MapReduce framework, it is advantageous for overcoming is steep learning curve of MapReduce. Hive's query language HiveQL supports much of the SQL specifications along with the Hive-specific specifications [11]. Hive allows us to write and execute MapReduce jobs with its SQL like syntax in the same time that it would take to write main method in JAVA (Fig. 2).

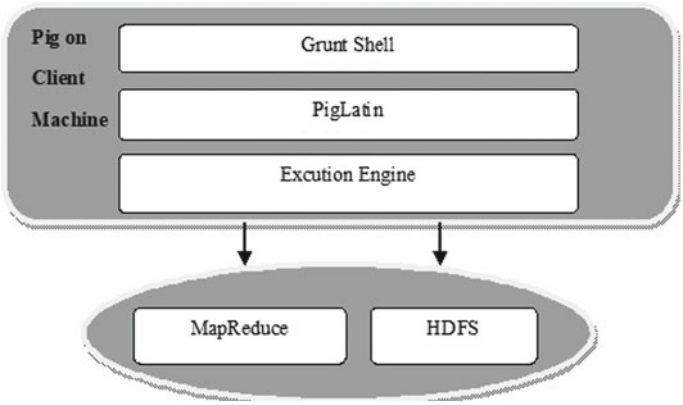


Fig. 1 Pig architecture

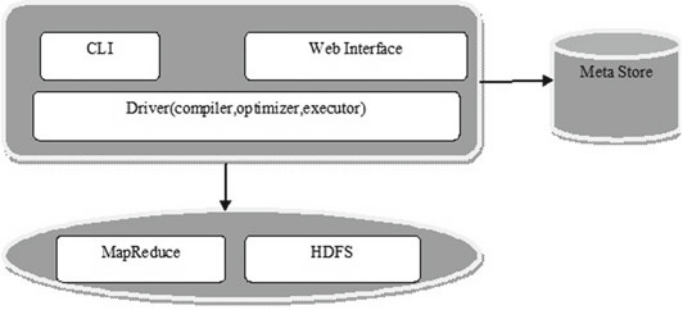


Fig. 2 Hive high-level architecture

2 Approaches and Technique

2.1 Environment Setup

Hadoop is ideal for storing, managing and processing big files on parallel-distributed environment. Our Hadoop HDFS [12], Apache Pig and Apache Hive are deployed in single-node multi-cluster manner on same hardware. Node has one core at 2 GHz and 4 GB of primary memory. Node runs CentOS-6.5 (Red Hat 64 bit). JDK1.7.0_67 is used for all tools deployed.

- Hadoop: We have used Hadoop 2.7.3 to run MapReduce with all updated features on YARN.
- Pig: Pig version 0.16.0 used is for our purpose.
- Hive: Version 2.7.3 is running on HDFS 2.7.3.

2.2 Query Statements and Dataset Details

Big data present an exciting opportunity to pursue large-scale analyses over collections of data in order to uncover valuable insights across a myriad of fields and disciplines. This work resolves the need for analyzing large amounts of information, so the healthcare providers can provide lifesaving diagnoses or treatment options efficiently, almost immediately.

Datasets are picked from the Data Science Central [13] and UCI Machine Learning Repository [14]. Some preprocessing has been performed on original data to reduce the complexity of the analysis. The dataset represents 10 years (1999–2008) of clinical care at 130 US hospitals and integrated delivery networks. There are four queries executed to Apache Pig and Hive. To estimate the average time of execution, each query is executed four times. Table 1 presents the list of query statements that were executed on both analyzing tools Pig and Hive.

Table 1 Query statements for healthcare dataset

Query	Description	Statements
Q1	Report of total number of patients registered for any disease	Join
Q2	Report of distinct diseases in male patient with average age on each disease	Join and aggregation
Q3	Report of distinct diseases in female patient with average age on each disease	Join and aggregation
Q4	Report of each patient with number of diseases he/she has and Sort the patient by many of diseases	Aggregation function and sorting

2.3 Experimental Outcomes

Apache Pig and Hadoop use the MapReduce essence to split, transform and merge to get the query execution done [15]. Table 2 represents the query results on Pig while performing on healthcare dataset.

or Complex query execution with multiple joins and filter conditions on large dataset is required, in that case Pig deals with it efficiently and it is capable of processing unstructured and semistructured data. We have taken healthcare datasets for analysis in structured formats.

This dataset is heterogeneous in nature due to data varieties. For example, patient may go under different tests and all the records and reports may not be in structured formats. Pig query loads both data files for analysis and then generates defined schema and finally calculates the different parameters. We have observed that in second and third requirement, datasets are split in homogeneous distribution that improves the performance in terms of time in which we are getting the results. In the last objective, query contains aggregation and sorting function applied on heterogeneous merged file; then, it required considerable time to get the result. Performance of Pig analysis with mean time is presented in Fig. 3.

Table 2 Pig execution time (in s) of queries on healthcare dataset

Queries	Run I	Run II	Run III	Run IV	Mean time
Q1	38	37	37	36	37
Q2	29	28	27	28	28
Q3	26	27	26	26	26.25
Q4	53	52	54	52	52.75

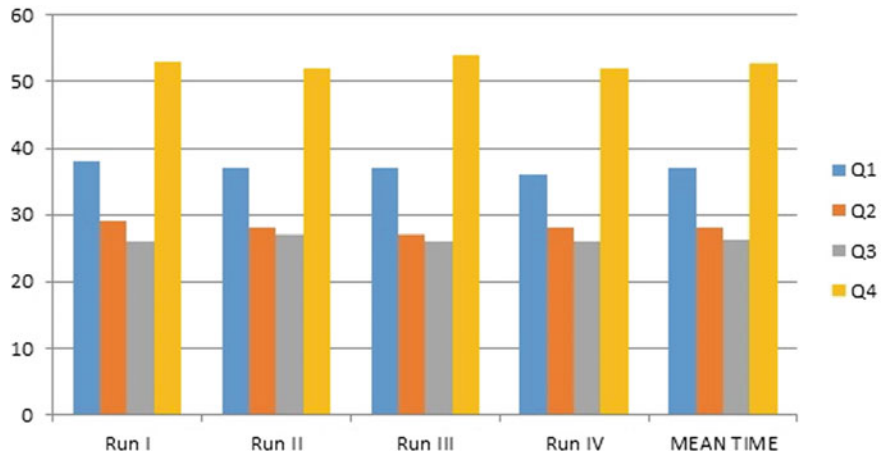


Fig. 3 Pig result on healthcare dataset with mean time (in seconds)

Table 3 Hive execution time (in s) of queries on healthcare dataset

Queries	Run I	Run II	Run III	Run IV	Mean time
Q1	23	26	25	23	24.25
Q2	22	24	23	21	22.5
Q3	24	23	22	23	23
Q4	42	44	42	43	42.75

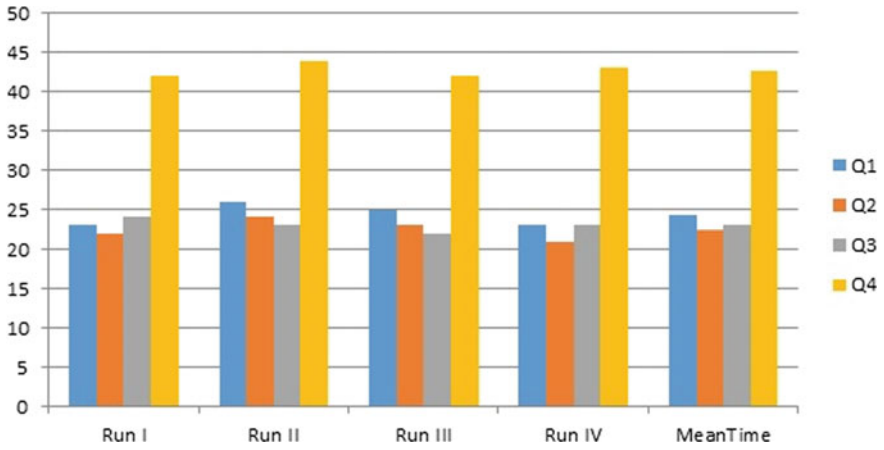


Fig. 4 Hive result on healthcare dataset with mean time (in seconds)

Hive results followed by representation of mean time are shown in Table 3 and Fig. 4, for all four queries. Hive is efficient by invoking MapReduce only when query is having join and aggregation function like sorting [16]. Hive deals with the batch tasks, for that, queries are converted into MapReduce jobs transparently.

By creating external table, large unstructured data will be provided in the schema. Our observation shows Hive's response time is fast in all the four cases as utilizes indexing in files but in both the cases homogeneous data increases the efficiency.

3 Conclusions

Analytical study shown in this work aimed to present insights of Apache Pig and Apache Hive data processing techniques. Hadoop HDFS was used to host the dataset, and same query objectives were experimented on both the tools. To optimize a plan, simple rule-based algorithm is used by both Pig and Hive but results show that Pig generates results more effectively when dataset is very large and complex queries were executed. Hive overcomes these issues and produces the results in lesser time with basic queries. This paper presents the in-depth analysis of these performance

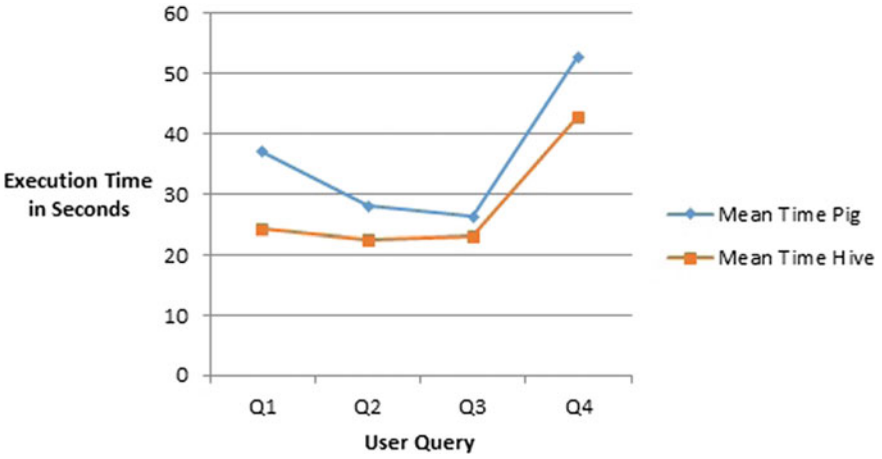


Fig. 5 Comparison of mean execution time (in seconds)

differences and system utilization differences between the two frameworks. The investigation can be further continued for larger size of data file including both structured and unstructured data with many more attributes included.

Comparative analysis for Pig and Hive is presented in graph in Fig. 5, while considering mean time for generating results. Hive can overcome MySQL Cluster and Pig performance on low-cost hardware with basic query. However, this is tested on a low-cost hardware.

References

1. A. Luntovskyy, D. Guetter, M. Klymash, Up-to-date paradigms for distributed computing, in *2nd International Conference on Advanced Information and Communication Technologies (AICT)*. IEEE Conference Publications, Ukraine, 2017, pp. 113–119
2. H. Cai, B. Xu, L. Jiang, IoT-based big data storage systems in cloud computing: perspectives and challenges. IEEE Internet Things J. **4**(1), 75–87 (2017)
3. Apache Hadoop, http://hadoop.apache.org/
4. K.H. Lee, Y.J. Lee, H. Choi, Y.D. Chung, B. Moon, Parallel data processing with MapReduce: a surver, SIGMOD Rec. **40**(4), Korea (2011)
5. S. Ryza, U. Laserson, S. Owen, J. Wills, *Advanced Analytics with Spark: Patterns for Learning from Data at Scale* (O'Reilly Media, 2015)
6. C. Cao, W. Wang, Y. Zhang, X. Ma, Leveraging column family to improve multidimensional query performance in HBase, in *IEEE 10th International Conference on Cloud Computing (CLOUD)* (IEEE Conference Publications, USA, 2017), pp. 106–113
7. A. Thusoo, J.S. Sarma, N. Jain, Z. Shao, P. Chakka, S. Anthony, H. Liu, P. Wyckoff, R. Murthy, Hive: a warehousing solution over a map-reduceframework. Proc. VLDB Endow **2**(2), 1626–1629 (2009)
8. Z. Shao, A. Thusoo, J.S. Sarma, N. Jain, Hive-a petabyte scale data warehouse using hadoop, in *Data Engineering (ICDE)* (2010)

9. A. Gates, *Programming Pig*. O'Reilly Media, 1st edn. (October 2011)
10. C. Olston, B. Reed, U. Srivastava, R. Kumar, A. Tomkins, Pig latin: a not-so-foreign language for data processing, in *Proceedings of the ACM SIGMOD International Conference on Management of Data*, ACM (2008), pp. 1099–1110
11. T. Liu, J. Liu, H. Liu, W. Li, A performance evaluation of Hive for scientific data management, in *IEEE International Conference on Big Data* (2013), pp. 39–46
12. K. Shvachko, H. Kuang, S. Radia, R. Chansler, The hadoop distributed file system, in *Proceedings of the IEEE 26th Symposium on Mass Storage Systems and Technologies (MSST '10)* (IEEE Computer Society, USA, 2010), pp. 1–10
13. Data Science Central, https://www.datasciencecentral.com
14. UCI Machine Learning Repository, http://archive.ics.uci.edu
15. A. Holmes, *Hadoop in Practice*, 2nd edn. (September 2014)
16. A. Choudhary C.S. Satsangi, *Query Execution Performance Analysis of Big Data Using Hive and Pig of Hadoop*, vol. Su-9, no. 3, Sep. (2015), pp. 91–93

Improved Ranking for Search Over Encrypted Cloud Data Using Parallel Index

Anu Khurana, C. Rama Krishna and Navdeep Kaur

Abstract With proliferation in the number of users/organizations adopting cloud computing, cloud storage in particular, the storage servers now hold large volumes of data. At such times, for enhancing user experience, a search mechanism is required for locating the desired and relevant documents quickly. As per the existing research, most of the keyword search schemes for performing search over the encrypted cloud data create an encrypted index of the document to be uploaded along with the encrypted document. The index created serves as the basis of performing the search and contains the unique keywords from the text of the document. In information retrieval systems metadata has proven to be important for improving the ranks of the relevant pages for the issued queries. On similar explanation, we proffer a mechanism in which we extend the main index with an additional parallel index containing metadata. In the experiments conducted, the search performed over the encrypted documents using this additional parallel index shows improvement in the ranking of the documents listed in the search results.

Keywords Encrypted cloud data · Index file · Metadata · Ranking
Parallel index file · Search

A. Khurana (✉)
I.K. Gujral PTU, Kapurthala, Punjab, India
e-mail: annu_khurana@yahoo.com

C. Rama Krishna
Department of Computer Science and Engineering, NITTTR, Chandigarh, India
e-mail: rkc_97@yahoo.com

N. Kaur
Department of Computer Science and Engineering,
Sri Guru Granth Sahib World University, Fatehgarh Sahib, Punjab, India
e-mail: drnavdeep.iitr@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_12

1 Introduction

Cloud computing is now widespread with a high percentage of users and organizations using cloud computing for storage. A study by Clutch over some organizations shows that 70% of these organizations used cloud services for file storage, 62% for backup and recovery and 51% for application deployment [1]. The number of cloud storage users is expected to rise even more in the near future [2] and so will the number of documents stored. It will not be wrong to say that "The cloud is now becoming a home for sensitive data" as users now even store their sensitive data on the cloud. Therefore, to secure user's data, documents are uploaded in encrypted format. However, this makes the task of searching over the encrypted data onerous.

As per the literature review, Song and Wagner [3] were the first to give practical solution to the problem of searching over the encrypted data. The threat for performing search over the encrypted data is the untrusted/partially trusted server. Therefore, the need is encrypted data, encrypted query and encrypted results. So that, in the entire process of searching, no such information is leaked to the server, from which it can decipher the data or its importance. One way to search over the encrypted data is to perform sequential scan. But it may not be efficient when the data size is large. Instead, a pre-computed index after encryption can be used to perform search more effectively without sacrificing security [3]. The precomputed index is built by selecting unique keywords from the concerned document. Researchers have also variedly included frequency statistics of the selected unique keywords along in the index. In information retrieval systems, metadata plays an important role in resource discovery [4]. On similar lines metadata can prove helpful for locating relevant documents in cloud storage. However, the nature and extraction ways in cloud environment may vary from that in web.

In this paper, we proffer a scheme Parallel Index-Multi-Keyword Ranked Search (PI-MKRS) in which we extend the precomputed main index with an additional parallel index containing metadata. Both these indices are encrypted and uploaded along with the concerned encrypted document in the cloud. Through our experiments, we show that PI-MKRS improves the rank of the documents and returns the more relevant documents with higher ranks in the search result.

2 Related Work

Many researchers worked on the keyword-based search [5–23] for performing search over the encrypted data with preserving privacy as a key concern. Of these [7–9] worked toward providing a keyword-based privacy-preserving search mechanism over the encrypted cloud data using the constructed encrypted index. Following the same track some researchers further focused on enhancement and the other aspects of the problem like fuzzy search [10–14], synonym-based keyword search [15–20], cluster-based keyword search [21–23].

From a different aspect some researchers [24–26] worked toward providing an improved search with metadata. However, these search schemes are either not keyword based or they are not privacy preserving. Of these Prabavathy and Devi in [24] proffered a technique with deduplication. They split the file into unique chunks, where each chunk is associated with a chunkID. chunkID and the chunk location are stored in chunk index for retrieving the file. Metadata of the file is stored along with the file to facilitate retrieval. The metadata used composes of properties of the file and the start address of physical block of the file.

Anjanadevi and Vijayakumar [25] propose a scheme for indexing and metadata management for accessing the distributed data. They use indexing and metadata for determining the storage. The metadata used is the information about who uses the content and the number of content pieces used. It also holds information about the number of partitions and levels of the file storage. The search process is initiated by obtaining the metadata of the requested file, with which the server starts searching for block ids, that stores the spilt form of the requested file which are then merged and returned.

Yu et al. [26] give a search solution with metadata index (using K-D-B trees). They use index partitioning, and in a distributed architecture the metadata index is among distributed multiple servers. A complete index is stored on disk for every partition with a partial copy in the memory. This helps in updating the index. Here the on-disk index stored is a complete K-D-B tree, and the in-memory index is a partial K-D-B tree without the indexed metadata. For processing the query the coordinator decides the server(s) on which the request is to be forwarded. On the server that contains the matching results, the metadata further facilitates in deciding the partition(s) to be searched.

Xu and Zhu in [27] proposed an order-preserving encryption (OPE)-based multi-keyword search scheme over the encrypted cloud data. In this they follow a two-step procedure to rank the documents in the search results. In the first step they organize the documents using coordinate matching, i.e., as per the number of query terms contained in the document. In the second step they refine and rank the document set obtained under each category, i.e., number of matched query terms, from step 1 as per their term frequencies. Their inverted index is encrypted with OPE. The inverted index constructed in their scheme emphasizes on the keywords and their frequencies. The results obtained show that our scheme is better as it includes metadata and returns more relevant documents in the first k search results.

3 Proposed Scheme

We propose a privacy-preserving multi-keyword-based search scheme named Parallel Index-Multi-Keyword Ranked Search (PI-MKRS) wherein we extend the index file with an additional parallel file containing metadata. The challenge is to preserve the privacy while performing search over the encrypted data and hence improve the ranking of the relevant documents for the issued queries.

Metadata often helps in deciding the relevance of the information. We in PI-MKRS include one parallel index for metadata. The metadata included is "filename," "title of the document" and the "file description tags" called "file descriptors." Cloud has now many users storing their personal and official documents in it. At such times, users should follow [28] and they tend to follow the relevant naming convention for the easy discovery of the documents that are stored on some other servers. The title of the document also carries high weight as a component of metadata [29]. Over a document storage many a times the search queries issued have keywords from the title itself. Therefore, we include title as a component of metadata. File descriptors are the user-defined tags normally added by the data owner while uploading the encrypted documents. File descriptors are the keywords that define the purpose of the document(s) and at times classify them. For, e.g., while uploading some files like "The Tempest," "Julius Caesar" or "As you like it." The file descriptor "Tales of Shakespeare" can help locate these documents easily.

3.1 Proposed Scheme Architecture

In PI-MKRS we consider the cloud storage system to consist of four entities, namely data owner, data user, private cloud server and the public cloud server as shown in Fig. 1.

Data owner is responsible for uploading the encrypted document to the cloud. He/she creates an index I comprising of unique words from the text of the document and a parallel index $\bar{I}$ containing the metadata of the document. He/she encrypts the index I and $\bar{I}$ at first level with key M to form $I(M)$ and $\hat{I}(M)$. These are then sent to the private cloud server for second level of encryption with key S to form $I(M, S)$ and $\bar{I}(M, S)$ [15].

PI-MKRS scheme is an enhancement in CMRS scheme [15] that was for synonym search using query click logs. Therefore, the architecture and encryption standards in PI-MKRS follow that in CMRS. First level of encryption at data user end is done

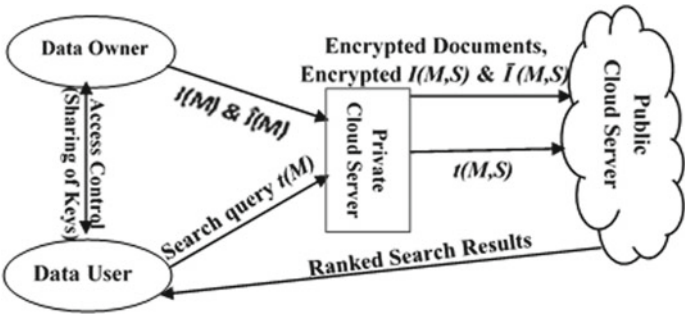


Fig. 1 PI-MKRS architecture

with key *M*. (This is a random set of English alphabets repeated many times.) The keywords are converted to bit string, by setting every found character in sequence to 1 and others to 0. The formed bit string called word vector is now shifted with a shifting factor. For, e.g., if the snippet of key *M* is a,d,s,a,p,d,r,t,p,q…., then with this the bit string/word vector for the word "add" will be 1100010000 and for the word "sad" it will be 0011010000. After shifting with a shift factor (mid-value in our case) these word vectors become 1000011000 for the word "add" and 1000000110 for the word "sad." Both indices and the search query are encrypted to form word vectors and sent to the private cloud server for second level of encryption.

Private cloud server contains a word vector table $\bar{W}V$ formed in the offline stage and contains the equivalent ciphers for every word vector. $\bar{W}V$ is formed by randomly selecting any random cipher (from a non-deterministic symmetric encryption scheme) of the word vector for recording. Second level of encryption is done by $\bar{W}V$ lookup method.

Data user is an authenticated user who can search and access the documents uploaded. Private cloud server is a trusted entity used for second level of encryption. Public cloud server is an untrusted entity used for storing documents and performing search. It has to be made sure that nothing is revealed to the public cloud server in the entire search process.

3.2 Index Structure

PI-MKRS involves parallel metadata index $\bar{I}(M, S)$ along with index $I(M, S)$. Index $I(M, S)$ consists of the unique words selected from the text of the document, their term frequency and their term frequency squares. Both term frequency and their squares are encrypted with Paillier encryption because of its additive homomorphic properties.

Metadata index $\bar{I}(M, S)$ consists of unique keywords from the title of the document, document name and file descriptor along with their term frequency and their term frequency squares. Metadata index also has the term frequency and their squares encrypted with Paillier encryption that enables us to compute similarity and yet prevent statistical leakages.

Both the indices randomly contain dummy data to prevent guessing attacks [30]. The term frequency for the dummy words is set to 0. Therefore, it does not affect the similarity scores.

3.3 Search Process

When a data user wants to locate any document, he/she issues a multi-keyword search query $t(M, S)$. The search query is encrypted with key *M* and then with key *S* in a manner similar to the indices encrypted. The search query issued contains some dummy keywords to prevent guessing attacks [30].

We use cosine similarity to compute the similarity scores as per Eq. 1. Similarity scores are computed for both indices, i.e., main index and metadata index in parallel. Since we are having the term frequency and their squares encrypted with Paillier encryption, no value is depicted in the entire search process except the final similarity scores from both indices. Paillier encryption is additively homomorphic so if we are having two values say a and b, and we encrypt them with Paillier encryption to get $e(a)$ and $e(b)$, then we can calculate $e(a) + e(b)$. Also we can use Paillier encryption to get multiplicative value of $e(a) * c$, where c is not encrypted. In order to perform $D_i \cdot Q_i$ we leave the vector values of term frequency for the issued query unencrypted.

$$similarity = \cos(\theta) = \frac{D \cdot Q}{||D||_2 \cdot ||Q||_2} = \frac{\sum_{i=1}^{n} D_i Q_i}{\sqrt{\sum_{i=1}^{n} D_i^2} \cdot \sqrt{\sum_{i=1}^{n} Q_i^2}} \tag{1}$$

We calculate the similarity scores in parallel (*sim1* for main index and *sim2* for parallel metadata index) by using Eq. 1. We calculate the final similarity score *sim12* by adding the weighted scores of *sim1* and *sim2, i.e.,* $w_1 \cdot sim1$ and $w_2 \cdot sim2$ as in Eq. 2. Weights *w1* and *w2* lie between 0 and 1 and are assigned in a manner that $w1 + w2 = 1$.

$$sim12 = w_1 \cdot sim1 + w_2 \cdot sim2 \tag{2}$$

4 Experimental Result

The experiments are done using Python on a Linux machine with Intel i7 processor with 8-GB RAM on MEDLINE IR test collection [31]. MEDLINE is an IR test collection of 1033 documents and 30 queries. We experimented by first extracting the unique keywords from the documents and adding some dummy keywords for the creation of its index *I(M).* We included the term frequencies of the keywords and the squares of the term frequencies encrypted with Paillier encryption in the index file *I(M).*This *I(M)* is sent to the private cloud server where it is encrypted to *I(M, S)*.

We queried the MEDLINE database with the 30 queries given along with the MEDLINE test collection. We extracted the unique keywords from the query to be issued, added some dummy keywords and formed an encrypted search query *t(M, S)*. The retrieval accuracy results with precision @ K and r-precision for scheme PI-MKRS, i.e., with both indices, main index *I(M, S)* and metadata index $\bar{I}(M, S)$ and for the scheme without using the meta index, i.e., with main index *I(M, S)* alone (for convenience we refer this scheme as MKRS in the table), are shown in Table 1.

The r-prec obtained with MKRS is 0.4581, whereas it is 0.3242 with TSR r-r 60 scheme in [27]. This shows that our scheme even without metadata index is able to retrieve more relevant documents until the rank that equals the number of relevant documents in our test collection. In Table 1, for MKRS the precision @ K value is good till p@5; thereafter, the number of relevant documents retrieved is

Table 1 Retrieval accuracy measures for both MKRS and PI-MKRS

Scheme	r-prec	p@1	p@5	p@10	p@20	p@50	p@100
MKRS	0.4581	0.8	0.67	0.57	0.47	0.27	0.17
PI-MKRS	0.5941	0.96	0.78	0.74	0.64	0.36	0.215

comparatively lower. However, in a search performed over the encrypted cloud data, we would like to get more relevant documents in the first K results returned. Hence, we need to improve the p@K values. Therefore, for the construction of metadata index, we gave relevant file names and titles to the documents in test collection. While outsourcing these documents we added file descriptors for these documents. We built a metadata index $\bar{I}(M, S)$ using the unique words from filenames, titles and file descriptors. We conducted the experiments on the same test collection with the same 30 queries again. The results for PI-MKRS as shown in Table 1 are obtained with weights, $w1 = 0.7$ and $w2 = 0.3$, for the calculation of similarity score $sim12$. We found that the ranking of the relevant documents improved and we got better p@K values and better r-prec values as shown in Table 1 for PI-MKRS scheme. This shows that adding a parallel metadata index helps in retrieving more relevant documents in the first K positions as it improved the ranking of the documents.

We also conducted the search experiments on a different collection of 3000 documents of different types like annual reports, emails, e-books, research papers, question papers and bills in docx, txt and pdf formats.

The idea to collect documents of different types was because of their varying information representation format. From the data set downloaded, only few filenames had relevant filenames. We deliberately renamed few files with relevant filenames following the naming convention and entered file descriptor for the files during upload. Title was extracted from the documents; however, because of the different formats title extraction process needed customization.

We uploaded two parallel indices $I(M, S)$ and $\bar{I}(M, S)$ along with the encrypted documents. A search query $t(M, S)$ is then issued to get ranked results, and the r-prec values with PI-MKRS showed 35% increase over the MKRS scheme. Hence, it improved the ranking of the relevant documents in the search results.

In Table 2, we show the ranking order of the relevant documents returned as a result of the search query "cloud storage." In this we consider 15 relevant documents returned against the issued search query "cloud storage." These documents are saved on the cloud with document ids from $d1$ to $d15$. We further consider four cases for the presence or absence of the query keywords. These cases are named c1, $c2$, $c3$ and $c4$ with meaning as follows.

Case 1 ($c1$): Both keywords "cloud" and "storage" are present;
Case 2 ($c2$): Only Keyword "cloud" is present;
Case 3 ($c3$): Only Keyword "storage" is present;
Case 4 ($c4$): Both keywords "cloud" and "storage" are absent.

Table 2 Ranking of documents with different weights to main index and parallel index having metadata for the query "cloud storage"

Doc ID	Keywords presence codes		Ranking of documents (calculated with Eq. 2, w_1 is weight for main index and w_2 is weight for metadata index) for query "cloud storage"										
	Main index	Metadata index	$w_1 = 0.1$, $w_2 = 0$	$w_1 = 0$, $w_2 = 0.1$	$w_1 = 0.01$, $w_2 = 0.09$	$w_1 = 0.02$, $w_2 = 0.08$	$w_1 = 0.03$, $w_2 = 0.07$	$w_1 = 0.04$, $w_2 = 0.06$	$w_1 = 0.05$, $w_2 = 0.05$	$w_1 = 0.06$, $w_2 = 0.04$	$w_1 = 0.07$, $w_2 = 0.03$	$w_1 = 0.08$, $w_2 = 0.02$	$w_1 = 0.09$, $w_2 = 0.01$
(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)	(l)	(m)	(n)
d1	c1	c1	d1	d1	d1	d1	d1	d1	d1	d1	d1	d1	d1
d2	c1	c2	d2	d13	d5	d5	d5	d5	d3	d3	d3	d3	d3
d3	c1	c3	d3	d5	d9	d9	d9	d3	d5	d5	d2	d2	d2
d4	c1	c4	d4	d9	d13	d13	d3	d9	d2	d2	d5	d4	d4
d5	c2	c1	d5	d11	d3	d3	d13	d7	d7	d7	d4	d5	d5
d6	c2	c2	d6	d15	d7	d7	d7	d2	d9	d4	d7	d7	d7
d7	c2	c3	d7	d3	d11	d11	d11	d13	d4	d9	d6	d6	d6
d8	c2	c4	d8	d7	d15	d15	d2	d11	d6	d6	d9	d8	d8
d9	c3	c1	d9	d10	d2	d2	d15	d6	d11	d11	d8	d9	d9
d10	c3	c2	d10	d14	d6	d6	d6	d4	d13	d8	d11	d11	d11
d11	c3	c3	d11	d2	d10	d10	d10	d15	d10	d13	d10	d10	d10
d12	c3	c4	d12	d6	d14	d14	d4	d10	d8	d10	d13	d12	d12
d13	c4	c1	–	–	d4	d4	d14	d8	d15	d15	d12	d13	d13
d14	c4	c2	–	–	d8	d8	d8	d14	d12	d12	d15	d15	d15
d15	c4	c3	–	–	d12	d12	d12	d12	d14	d14	d14	d14	d14

In Table 2, column (a) shows the relevant documents for the search query "cloud storage." Columns (b) and (c) show the presence or absence of the query keywords in the main index and metadata index, respectively. Column (d) to column (n) show the ranked search result for the query "cloud storage" computed with Eq. 2 by assigning different weights *w1* and *w2* to *sim1* and *sim2*. Column (d) shows the ranked result with main index alone. Thereafter, column (f) to column (n) show the ranked results with inclusion of metadata index with different weights. The ranking results obtained in Table 2 show that adding metadata index helps in listing the documents that are relevant but actually do not contain the query keywords in its main index. So probably these keywords when extracted either due to their presence in the filename or added as file descriptors helped in including such documents in the search results. Even the documents that have query keywords in the main index and metadata index both got higher ranks in search results. During experiments weight $w1 = 0.7$ and weight $w2 = 0.3$ gave best results in form of $p@k$ and r-prec. However, for a cloud storage system if the metadata (i.e., relevant document name, title and file descriptor) is more appropriate and relevant, these weights could be adjusted accordingly. We also experimented by including metadata keywords in the main index itself rather than constructing a separate metadata index, but here the ranking order was almost the same as the one shown with $w_1 = 1$ and $w_2 = 0$ (column (d)) in Table 2.

5 Conclusion

In this paper we proffered a scheme Parallel Index-Multi-Keyword Ranked Search (PI-MKRS) for improving the ranking order of the relevant documents in the search process. PI-MKRS is privacy preserving as it only reveals nothing except the final sums for the similarity scores of the indices during the search process. Over the documents stored in the cloud, most of the queries issued for locating a document can be fulfilled by a relevant filename, title of the document and the document tags called file descriptors. Hence, the additional parallel index for computing similarity improved the ranking of the relevant documents against a search query. This additional index can also prove helpful in locating documents that may contain scanned documents or image formats.

References

1. Retrieved from www.waterfordtechnologies.com, https://www.waterfordtechnologies.com/cloud-computing-stats-2017/ (2017)
2. Retrieved from https://www.statista.com/statistics/499558/worldwide-personal-cloud-storage-users/ (2017)
3. D.X. Song, D. Wagner, Practical techniques for searches on encrypted data, in *IEEE Symposium on Security and Privacy* (2000)
4. D. Haynes, *Metadata for Information Management and Retrieval*. Facet publishing (2004)

5. A. Khurana, C. Rama Krishna, N. Kaur, Searching over encrypted cloud data, in *International Conference on Communications & Electronics.* Ghaziabad, India (2013), pp. 223–227
6. R. Handa, C. Rama Krishna, A survey on searching techniques over outsourced encrypted cloud data, in *8th International Conference on Advanced Computing and Communications Technologies*, Panipat, India, pp. 128–137
7. N. Cao, C. Wang, Privacy-preserving multi-keyword ranked search over encrypted cloud data, in *Proceedings of IEEE INFOCOM* (2011), pp. 829–837
8. Z. Xu, W. Kang, R. Li, K. Yow, C.Z. Xu, Efficient multi-keyword ranked query on encrypted data in the cloud, in *ICPADS* (2012)
9. C. Yang, W. Zhang, J. Xu, A fast privacy-preserving multi-keyword search scheme on cloud data, in *International Conference on Computing & Processing (Hardware/Software)* (2012), pp. 104–110
10. N.S. Khan, C.R. Krishna, Secure ranked fuzzy multi-keyword search over outsourced encrypted cloud data, in *5th International Conference on Computer and Communication Technology* (IEEE, Allahabad, 2014), pp. 241–249
11. J. Li, Q. Wang, C. Wang, Fuzzy keyword search over encrypted data in cloud computing, in *INFOCOM* (IEEE, San Diego CA, USA, 2010), pp. 1–5
12. M.M. Ahsan, M. Sabilah An efficient fuzzy keyword matching technique for searching through encrypted cloud data, in *International Conference on Research and Innovation in Information Systems (ICRIIS)* (2017)
13. S. Ding, Y. Yidong, L. Chen, An efficient and privacy-preserving ranked fuzzy keywords search over encrypted cloud data, in *International Conference on Behavioral, Economic and Socio-cultural Computing* (IEEE, USA, 2016)
14. J. Chen, K. He, L. Deng, EliMFS: achieving efficient, leakage-resilient, and multi-keyword fuzzy search on encrypted cloud data. Trans. Serv. Comput. **PP**(99), 2017 (n.d.)
15. A. Khurana, C.R. Challa, Search with context sensitive synonyms: in multi keyword ranked search over the encrypted cloud data, in *International Conference on Communication, Computing and Networking ICCCN-2017*, Chandigarh, India (2017), pp. 251–258
16. R. Handa, C.R., Privacy-preserving multi-keyword search supporting synonym query over encrypted data, in *1st International Conference on Communication, Computing and Networking.* Chandigarh, India (n.d.)
17. V. Saini, R. Challa, An efficient multi- keyword synonym-based fuzzy ranked search over outsourced encrypted cloud data, in *9th Springer International Conference on Advanced Computing and Communication Technologies* (Springer, India, 2015)
18. Z. Fu, X. Sun, Achieving effective cloud search services: multi-keyword ranked search over encrypted cloud data supporting synonym query. IEEE Trans. Consum. Electron. **60**(1), 164–172 (2014)
19. T.S. Ho, Efficient semantic search over encrypted data in cloud computing, in *International Conference on High Performance Computing & Simulation (HPCS)* (Bologna, 2014), pp. 382–390
20. X. Sun, Y. Zhu, Z. Xia, Privacy-preserving keyword-based semantic search over encrypted cloud data. Int J Secur. Appl. **8**(3), 9–20 (2014)
21. R. Handa, R.K. Challa, A cluster-based multi-keyword search on outsourced encrypted cloud data, in *2nd International Conference on Computing for Sustainable Global Development*, New Delhi (2014), pp. 115–120
22. R. Handa, R.K. Challa, An efficient cluster-based multi-keyword search on encrypted cloud data. CSI Commun. **39**(3), 20–27 (2015)
23. C.R. Gagan, Dynamic cluster based privacy-preserving multikeyword search over encrypted cloud data, in *6th International Conference on Cloud System and Big Data Engineering* (IEEE, Noida, India, 2016), pp. 146–151
24. B. Prabavathy, S.M. Devi, Multi-index technique for metadata management in private cloud storage, in *International Conference on Recent Trends in Information Technology (ICRTIT)* (2013), pp. 84–89

25. S. Anjanadevi, D. Vijayakumar, An efficient dynamic indexing and metadata based storage in cloud environment, in *International Conference on Recent Trends in Information Technology* (IEEE, 2014), pp. 1–6
26. Y. Yu, Y. Zhu, An efficient multidimension metadata index and search system for cloud data, in *6th International Conference on Cloud Computing Technology and Science* (IEEE, 2014), pp. 499–504
27. J. Xu, W. Zhang, C. Yang, J. Xu, N. Yu, Two-step-ranking secure multi-keyword search over encrypted cloud data, in *International Conference on Cloud and Service Computing* (IEEE, 2012), pp. 124–130
28. Retrieved from www.huridocs.org, https://www.huridocs.org/2016/07/file-naming-conventions-why-you-want-them-and-how-to-create-them/ (2017)
29. Retrieved from https://technet.microsoft.com, https://technet.microsoft.com/en-us/library/dn169065.aspx (2017)
30. C. Liu, L. Zhu, M. Wang, Search pattern leakage in searchable encryption: attacks and new construction. J Inf Sci Int J **265**, 176–188 (2014)
31. Retrieved from http://ir.dcs.gla.ac.uk, http://ir.dcs.gla.ac.uk/resources/test_collections/medl/ (2017)

A Framework for Data Storage Security with Efficient Computing in Cloud

Manoj Tyagi, Manish Manoria and Bharat Mishra

Abstract The modern technology defines cloud computing as a digital facility which is accessed by users over the web. The proposed framework provides the protection according to the classification of data. In this framework, cloud service provider (CSP) selects the eminent server using cuckoo algorithm with Markov chain process and Levy's flight. After server selection, user encrypts their data using elliptic curve integrated encryption scheme (ECIES) at the user side and sends it to CSP for storage, where CSP stores the data after applying second encryption on it using advanced encryption standard (AES) at the cloud side. This double encryption provides the confidentiality on both the sides. This framework integrates server selection approach, authentication and encryption schemes in order to achieve the integrity, confidentiality as well as efficient computing.

Keywords Cloud computing · Cuckoo search algorithm · Cloud service provider Confidentiality · Integrity · AES · ECIES · Efficient computing

1 Introduction

Like the basic utility services of gas, telephony, water and electricity, cloud computing is a model which offers variety of services in the same manner. In this model, the services are delivered to user as per their need without showing the detail like how they delivered. Cloud computing has become popular in the recent years because of its easy access, use on demand and scalability [1]. Cloud computing provides

M. Tyagi (✉) · B. Mishra
Mahatma Gandhi Chitrakoot Gramodaya Vishwavidyalaya, Chitrakoot, India
e-mail: manojtyagi80.bhopal@gmail.com

B. Mishra
e-mail: bharat.mgcgv@gmail.com

M. Manoria
Sagar Institute of Research Technology and Science, Bhopal, India
e-mail: manishmanoria@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_13

several services such as SaaS, PaaS and IaaS to users remotely through internet. Through these services, cloud computing offers variety of resources in the form of storage, software and computing power to individuals, groups and organizational users. Cloud computing is widely accepted and successful in the market because of its well distribution of resources and on-demand availability. As the data are stored in cloud remotely, users are under the impression that they have no command on their data and are insecure about the safety of the data as claimed by cloud provider [2].

One of the main issues in cloud is to achieve efficient scheduling in cloud, considered for the research. The allotment of task among the resources in efficient way, in order to enhance the performance, is the main aim of scheduling. Scheduling is a time-taking process to search an eminent solution. Maximum algorithms take long time to generate best possible solution, but meta-heuristic approaches like league championship algorithm (LCA), genetic algorithm (GA), ant colony algorithm (ACA), BAT algorithm, cuckoo search algorithm (CSA) and particle swarm algorithm (PSA) take reasonable time to find the best solution [3]. Cloud computing offers storage services to the users through internet where they keep their data remotely in the cloud. Remote computing brings various security challenges and security issues for both the user and the CSP. The CSP offers its services across the internet and applies various network technologies that also bring security issues [4]. Cloud computing faces several number of security challenges like shared technologies vulnerabilities, data breach, account hijacking, service hijacking, denial of service (DoS) and malicious insiders [5].

2 Related Work

Makhloufi et al. [6] investigated flower pollination algorithm (FPA), CSA and firefly algorithm (FFA) to determine the optimal power flow in the Adrar power system. He proved that CSA is better than FFA and FPA to find optimum solution. Bhateja et al. [7] check the practicability of CSA as a cryptanalysis tool for Vigenere cipher. Through experimental results, they proved that CSA is better in fast convergence and accuracy over GA and PSA. Zineddine [8] proposed CSA with Levy flights to discover the best possible set of answer to known set of vulnerabilities.

Mareli and Twala [9] showed that CSA is effective and efficient in solving global optimization. Through experimental result, he showed that CSA is better than differential evolution (DE), simulated annealing (SA) and PSA. Tawalbeh et al. [10] proposed a model for secure cloud computing based on data classification. He applied the various levels of security based on types of data.

Yahya et al. [11] discussed various levels of data security based on security level it wants. Hank et al. [12] tell in his books that ECIES is secure against adaptive chosen-ciphertext attacks. Martínez and Encinas [14] describe that ECIES is the well-known cryptographic method included in various standards like SECG SEC 1, ANSI X9.63, IEEE 1363a and ISO/IEC 18033-2.

3 Problem Formulation

In today's scenario, many people use cloud for data storage. A cloud is a collection of many servers situated remotely and shared by many users, so server selection for storage of data and its security is mandatory in this environment. Here, the focus is on two issues: firstly to choose the eminent data storing server in the cloud, and secondly data security in the cloud and prevent its unauthorized access. Many scheduling algorithms are available to decide which server is suitable for storing the particular data in the cloud. In the recent years, nature-based algorithms like ACA, FPA, GA, PSA, FFA, BAT and CSA are very popular in many areas to find the optimum solution. Many approaches like authentication, confidentiality and integrity are available for security of various types of data stored in the cloud. The level of security depends on data classifications, namely very sensitive data, sensitive data and protected data [11].

4 Proposed System

An environment capable of providing reliability, customization and guarantee has been introduced by cloud computing in computational services for its customer. Here, centralized data centers for different application and databases are used. This work mainly focuses on server-side computation and data security; in this way, it fulfills the requirement of secure data storing and other security features in the cloud. Each user's data received at the server goes through security mechanism consisting of authentication and encryption. After each successful access from the user to the cloud system, dynamic server stipulation is applied. An optimized result is obtained in the server selection by utilized cuckoo algorithm with Markov chain random walk and Lévy flights. Once the efficient server is selected, this framework allows the system for data storing and access to the user. Based on data classification, user chooses the security mechanism to manage the various security levels. Single-factor authentication is applied for protected data, and for sensitive data multifactor authentication is applied [11]. Owner and user both use the ECIES at own side for confidentiality and integrity. Encryption and decryption at client side remove the dependency on cloud for confidentiality (Fig. 1).

4.1 System Model

Owner (O). An individual or an organization which shares the data among its customers through cloud. Owner encrypts the data using ECIES at its end and sends this encrypted data (C_t) to cloud through secure Internet channel. It also sends the value of N_R and S_t to its customer.

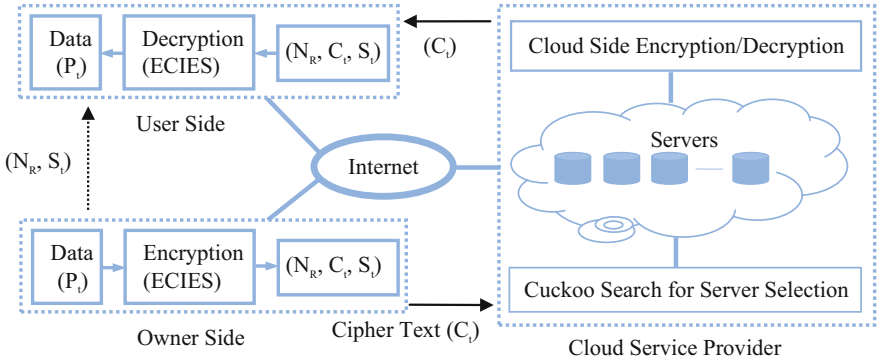


Fig. 1 Proposed framework

User (U). An individual or an organization which wants to use the data from the cloud. It receives the ciphertext from cloud and generates the tag S'_t for received ciphertext using ECIES and then compares it with tag S_t. If these tags are same, then the user decrypts the ciphertext using ECIES at its end and receives the data.

Cloud Service Provider (CSP). It is responsible authority for cloud. It receives the encrypted data from the owner and again encrypts it using symmetric cryptographic method AES-128 and AES-256 for protected data and sensitive data, respectively, and the data are secured at the cloud side. CSP also uses the cuckoo algorithm for selecting an efficient server and stores this double-encrypted data in cloud. It is also responsible to decrypt the data using respective AES method, when users request data access. So CSP converts data from double-encrypted form to single-encrypted form and sends these data to user in encrypted form, where user decrypts it.

4.2 Efficient Server Selection

Cuckoo is a brood parasite which cleverly chooses the nest of the other birds for laying their eggs where its eggs are taken care by the host bird. This behavior of cuckoo is adopted by CSA to find an optimal solution. Markov chain random walk and Lévy flights are helpful in cuckoo search to improve its performance. CSA is a naturally inspired meta-heuristic approach which is used for best possible searching [7]. The server has a set of virtual machines, and to select an efficient server to store the user's data in the cloud, here CSA is applied. CSA generates cuckoo habitat matrix assuming the population of eggs. Here, an egg is denoted as jobs, cuckoo works as scheduler, and nest represents virtual machine (VM). Now, calculate the fitness function of virtual machine for all servers and select the efficient server based on fitness function. Here, J_i represents the jobs, n represents the number of VM's, processor and memory are denoted by C_i and M_i, respectively, and mu represents the quantity of memory units. VM_{fit} represents the fitness function of virtual machine.

$$VM_{fit} = \frac{\sum_{i=1}^{n} j_i}{\left(\frac{\sum_{i=1}^{n} C_i \sum_{i=1}^{n} M_i}{mu}\right) \times t}. \quad (1)$$

4.3 Data Storage Security

When the size of security level is small, RSA-AES performs better than ECIES in execution time, whereas ECIES performs fast execution than RSA-AES on higher security level. It clearly shows that ECIES is better than RSA-AES and a legitimate option to RSA-AES that combines asymmetric and symmetric cryptography in an efficient manner. The execution result of encryption/decryption process on 1 KB text file is given in Table 1.

At client end and server end, ECIES [13] and AES are used, respectively. KDF, ENC, KA and MAC are key derivation function, encryption function for a symmetric key, key agreement and message authentication code, respectively. KDF is a method of generating the keys using some parameters. Through ENC, data are translated into encrypted form using common key which is confidential for the communicators. Under the key agreement, sender and receiver follow a mutually defined agreement. MAC verifies the data at the receiver end. Diffie–Hellman derives the symmetric keys G_1 and G_2, where G_1 and G_2 are used for encryption and certification of cipher text, respectively. ECIES protects data from adaptive chosen-ciphertext attacks [12].

Various standards use XOR function for translating the message into cipher. But in some time, if we vary the length of data, the XOR fails to provide the security to the cipher text. So AES can be preferred for encryption purpose to strengthen ECIES. A set of techniques given in Table 2 are proposed in the paper for attaining the secure ECIES version [14].

Here, we take an elliptic curve R for finite field F_f [13]. Let H be the point of prime order p in $E(F_f)$ and domain parameters are (p, F_R, R, c, d, H, m, h), where field is represented by F_R, the elliptic curve created randomly is represented by R, the coefficient c and d are used to define the equation of elliptic curve, h is the cofactor, and m is the order of p.

Table 1 Execution time for RSA-AES and ECIES [14]

Security level	RSA-AES			ECIES		
	RSA key length	Encryption time (ms)	Decryption time (ms)	Key length	Encryption time (ms)	Decryption time (ms)
80	1024	1.81	1.81	160	7.41	7.36
112	2048	6.42	6.53	224	13.73	13.31
128	3072	18.88	19.01	256	16.96	17.27
192	7680	240.55	239.76	384	40.77	39.75
256	15,360	1827.05	1834.13	512	81.52	82.84

Table 2 Set of techniques used in ECIES

KA	HASH	KDF	ENC	MAC
DH	SHA-512	KDF2	AES-128 in CBC mode with PKCS#5 padding	HMAC-SHA-512

An elliptic curve R over F_f is described by the given equation:

$$y^2 = d + cx + x^3. \tag{2}$$

Here, c, d $\in F_f$ satisfy $4c^3 + 27d^2 \neq 0 \pmod{p}$.

4.3.1 Key Pairs

To obtain secret key and public key, the process which is applied is key generation.

ECIES key generation [13]
1. Select private key $S_k \in_R [1, m - 1]$.
2. Generate public key $P_k = S_k H$.
3. Return (P_k, S_k).

4.3.2 Encryption

At owner end, data are converted into ciphertext by performing encryption and also generate a tag S_t for cipher text.

ECIES encryption [13]
1. Choose $G \in_R [1, m - 1]$.
2. Compute $N_R = GH$ and $V = hGP_k$. If $V = \infty$ then go to step 1.
3. $(G_1, G_2) \leftarrow$ KDF (xV, N_R), where xV is the x-coordinate of V.
4. Compute cipher text $C_t = ENC_{G1}(P_t)$ and tag $S_t = MAC_{G2}(C_t)$.
5. Return (N_R, C_t, S_t).

4.3.3 Decryption

Decryption is performed at the user side where the user decrypts the cipher text into data after signature or tag certification.

ECIES decryption [13]
1. Perform an embedded public key validation of N_R. If the validation fails return ("cipher text rejected").
2. Compute $V = hS_kN_R$. If $V = \infty$, return ("cipher text rejected").
3. $(G_1, G_2) \leftarrow KDF\ (xV, N_R)$, where xV is the x-coordinate of V.
4. Compute $S_t^{'} = MAC_{G2}(C_t)$. If $S_t^{'} = S_t$, return ("cipher text rejected").
5. Compute $P_t = DEC_{G1}\ (C_t)$.
6. Return (P_t).

Proof for decryption process. The proof consists that the valid user generates an encrypted data (N_R, C_t, S_t) from P_t.

$$hS_kN_R = hS_k(GH) = hG(S_kH) = hGP_k.$$

After that, similar keys (G_1, G2) are used at the receiver side to find the actual data.

Security Proofs for ECIES. The security of ECIES is considered very high as it depends on the phenomena that symmetric key encryption and MAC algorithm applied here are secure in nature. The computation and transferring of key in DH are quite complex which make it more secure as it uses KDF, in turn provides a proven security to ECIES.

5 Conclusion

This work has recognized the data security issues and has provided a framework that utilizes the recourses as well as data security. This framework has been designed in such a way that it uses security mechanism based on data classification as well as it selects eminent data storage server. CSA has been used for selecting the most suitable server to accomplish cloud computing efficiently. ECIES has been used for encryption and to assure data integrity and secrecy at user end, which is further encrypted at cloud server side before storing; in this way, it enhances user's data security in the cloud. Finally, this framework shows the confirmation of integrity, authentication and confidentiality of data as well as computing efficiency.

References

1. S.M. Habib, S. Hauke, S. Ries, M. Muhlhauser, Trust as a facilitator in cloud computing: a survey. J. Cloud Comput. Adv. Syst. Appl. **1**, 19 (2012)
2. R. Buyyaa, C.S. Yeoa, S. Venugopala, J. Broberg, I. Brandic, Cloud computing and emerging IT platforms: vision, hype, and reality for delivering computing as the 5th utility. J. Future Gener. Comput. Syst. **25**, 599–616 (2009)
3. M. Kalra, S. Singh, A review of metaheuristic scheduling techniques in cloud computing. Egypt. Inf. J **16**, 275–295 (2015)

4. A. Singh, K. Chatterjee, Cloud security issues and challenges: a survey. J. Netw. Comput. Appl. **79**, 88–115 (2017)
5. L. Coppolino, S. D'Antonio, G. Mazzeo, L. Romano, Cloud security: emerging threats and current solutions. Comput. Electr. Eng., 1–15 (2016)
6. S. Makhloufi, A. Mekhaldi, M. Teguar, Three powerful nature-inspired algorithms to optimize power flow in Algeria's Adrar power system. J. Energy **116**, 1117–1130 (2016)
7. A.K. Bhateja, A. Bhateja, S. Chaudhury, P.K. Saxenaa, Cryptanalysis of Vigenere cipher using Cuckoo Search. Appl. Soft Comput. **26**, 315–324 (2015)
8. M. Zineddine, Vulnerabilities and mitigation techniques toning in the cloud, a cost and vulnerabilities coverage optimization approach using Cuckoo search algorithm with Levy flights. J. Comput. Secur. **48**, 1–18 (2015)
9. M. Mareli, B. Twala, An adaptive Cuckoo search algorithm for optimisation. Appl. Comput. Inf. (2017)
10. L. Tawalbeh, N.S. Darwazeh, R.S. AlQassas, F. AlDosari, A secure cloud computing model based on data classification. Procedia Comput. Sci. **52**, 1153–1158 (2015)
11. F. Yahya, R.J. Walters, G.B. Wills, Protecting data in personal cloud storage with security classifications, in *Science and Information Conference*, IEEE (2015)
12. V. Tilborg, C.A. Henk, S. Jajodia, *Encyclopedia of Cryptography and Security*. Springer (2011)
13. D. Hankerson A. Menezes, S. Vanstone, *Guide to Elliptic Curve Cryptography*. Springer (2004)
14. V.G. Martínez, L.H. Encinas, A.Q. Dios, Security and practical considerations when implementing the elliptic curve integrated encryption scheme. J. Cryptologia **39**(3), 244–269 (2015)

Trust and Security to Shared Data in Cloud Computing: Open Issues

Bharti L. Dhote and G. Krishna Mohan

Abstract Cloud computing is encouraging technology to the users, but not satisfactory in trust management, which hinders market growth. While sharing data, owner expected to restrict the authorised or unauthorised users from modification. It is essential to have a robust cryptographic mechanism which can provide fine-grained data access control with confidentiality, authenticity and anonymity at the same time. In addition, framework is needed for analysing trust management systems which can help to develop solutions to challenges such as identification, privacy, integration, security. In this paper, we focus on the data security from the perspective of three stakeholders, i.e. data owner, users and cloud provider. From the surveys analyse, there is still a need for new approach and policies is to be devised for trust, secure data out sourcing, and access policies to be investigated. The first need is to improvise cloud storage service privacy, second is to improvise the cloud data accessing policies and the third need is to integrate of trust computing and access control for identifying cloud reliability.

Keywords Cloud computing · Data security · Trust · Access policies

1 Introduction

Cloud computing has emerged as a popular data storage and computation paradigm. Cloud provides three deployment models, namely private, public and hybrid. The three services provided by the cloud are software as a service (SAAS), platform as a service (PAAS) and infrastructure as a service (IAAS). It is a model of "use on Demand" and "pay as per use". It has got popularity because of its benefits to users. The user does not require to invest in or manage an extensive computing infrastructure

B. L. Dhote (✉) · G. Krishna Mohan
Department of Computer Science and Engineering, K. L. University, Vijayawada, India
e-mail: bhartildhote@gmail.com

G. Krishna Mohan
e-mail: gvlkm@kluniversity.in

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_14

but can access to the resources they need anywhere at any time. It also promises several benefits such as expense reduction, resource elasticity and simplicity. Many advantages of cloud computing motivate individuals and organisations to outsource their data from local server to public cloud servers. In addition to cloud platform and infrastructure, cloud application providers are also emerging such as Amazon, Microsoft and Google. These application providers offer user-friendly services for data storage.

It is a clear trend that cloud data outsourcing is versatile service. Hence it is concerned about data security with cloud data storage which is arising regarding reliability and privacy which raise the primary obstacles to the adoption of the cloud [1, 2]. Most of the work focuses on deploying the most vital data services, e.g. data management and utilisation while considering reliability and privacy assurance [2, 3]. Securing information systems from cyber attacks, malware and internal cyber threats is a difficult problem. Attacks on authentication and authorisation (access control) are one of the more powerful and potentially rewarding attacks on distributed architectures. A secure framework for analysing trust management systems can help researchers develop innovative solutions to authentication and authorisation challenges using identification and privacy schemes. On the contrary cloud services are highly dynamic, distributed and non-transparent nature makes establishing and managing trust among Cloud Service Providers (CSP) and consumers a significant challenge [4, 5].

This paper explores the problem of secure and reliable data outsourcing in cloud computing as a challenging issue. This study enlightens the various aspects of cloud data security in relevance to privacy and trust management.

The rest of the paper is organised as follows. Section 2 presents motivation for finding issues and challenges in cloud computing. Section 3 presents a state of the art. In Sect. 4, we talk about the trust model in the cloud computing. In Sect. 5, we present identified research gaps. Finally, the paper is concluded in Sect. 6.

2 Motivation

The local complex systems within organization for database management outsourced to the cloud due to its high flexibility and economic savings which motivate individuals and enterprises.

In the modern world more and more companies are running their services either on the private cloud belonging to the private company or public cloud owned by the external provider. The greatest advantage of the CC is the possibility of dynamically reacting to the demand of resources. However, even this solution has some limitations, and there might be a situation when it is forced to move our services to the other cloud. Information on this cloud is considered as an asset whether it be medical, financial, personal or some other form of valuable data. An example of such scenario is shown in Fig. 1.

For public or private cloud, access control is a fundamental aspect of any given information system. Several enterprise-level organisations deploy highly scalable

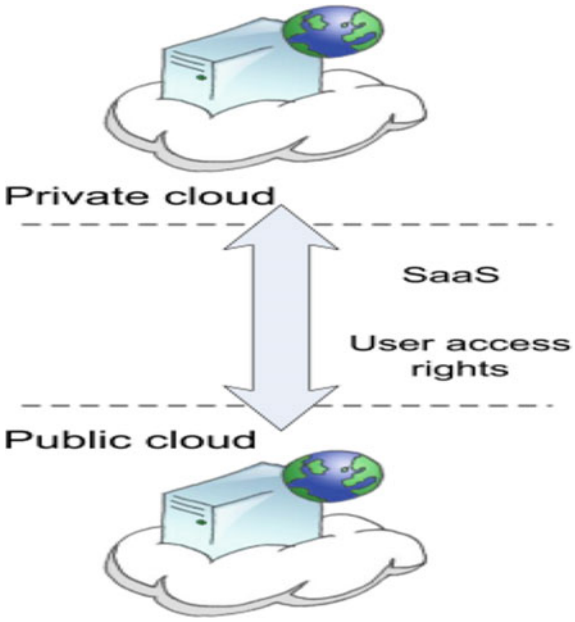


Fig. 1 Scenario for SaaS movement

cloud computing solutions for an internal data sharing and collaboration. Moreover, some companies make available their services for commercial use and act as a CSP in the market. CSPs assert to provide better security, reliability, sustainability, cost effectiveness and better support than an IT system of individual organisation. These features make it possible to move the business from individual systems to the cloud and make it accessible over the Internet. Data privacy and integrity issues are not adequately addressed at present though it is utilised by increasing number of organisations worldwide. Accessing sensitive data over the Internet increases the risk of privacy violation and data compromise without the knowledge where the data is residing which hinders a wide adoption of this technology [6, 7, 8].

To provide data security, most of the CSPs offer encryption mechanism to transfer and store the data of the consumers in the encrypted form in the cloud storage system [9, 7, 3]. However, inside threats and key management challenges still need addressed. Certification of CPSs may provide some level of assurance to the customers. However, for the customers, there are still no guarantees of full self-control over the data resided in the cloud.

3 Review of Related Work

Nowadays, open access data are a powerful method to share, stored in encrypted form to provide data security. But under which conditions the data are to be shared

is required. Access controls are the powerful authorised techniques for accessing shared data. In this section, how security is provided to share data is explained.

Threshold multi authority Security TMAC [10] performance analysis results show that TMAC is verifiably secure when less than threshold authorities are compromised. In a [11] proposed scheme based on the attribute-based encryption (ABE) is to reduplicate encrypted data stored in the cloud and support secure data access control at the same time. In dynamic certification of cloud services [4], the ever-changing cloud environment, fast technology cycles, regulatory changes, and increased adoption of business-critical applications demand highly reliable cloud services. The use of hybrid symmetric encryption [6] makes the data stronger against single encryption and makes it difficult for an attacker to get the original data. Attribute-based data sharing scheme [1] presents data sharing scheme to solve the key escrow issue and also to improve the access policy (AP) by providing the expressiveness of attribute with weight so that the resulting scheme is more friendly to cloud computing applications. Privacy-preserving ranked keyword method [2] which investigated cipher text search in the scenario of cloud storage. It explores the problem of the semantic relationship between different plain documents over the related encrypted documents and their maintenance. Trustworthy scheme [5] presents a test-based certification scheme proving non-functional properties of cloud-based services. In a secure anti-collusion data sharing scheme [12], private keys are obtained by the user through secure communication channel from the group manager Certificate Authorities. In novel two-factor data security protection mechanism [9], receiver's identity is known to data sender to encrypt the data, while the receiver is required to use both his/her secret key and a security device to gain access to the data. CCAF [13] multi-layered security can protect the data in real-time, and it has three layers of security: (1) firewall and access control; (2) identity management and intrusion prevention and (3) convergent encryption. Table 1 describes advantages and limitations of data sharing schemes.

Table 2 compares the existing data sharing schemes based on attribute-based encryption (ABE) type or certificate, policy expression, revocation method, backward and forward security.

4 Trust Model in Cloud

Understanding cloud computing privacy concerns requires knowing the key data stakeholders like Data owners, Data Consumers and Service Providers.

Data owners have their information stored in the cloud. Examples in the medical field include patients whose information is in databases that healthcare providers outsource to the cloud or physicians and hospitals whose private and sensitive professional and financial practices inferred by analysing the outsourced data. Their main concern is protecting their data and identities against unauthorised access or use. *Data consumers* query information for various reasons. For instance, physicians might consult patients' medical records before treatment, or researchers might query

Table 1 Review of related work

Paper title	Advantages	Limitations
TMACS: a robust and verifiable threshold multi-authority access control system in public cloud storage [10]	Provides an efficient method to combine the traditional multi-authority scheme with TMACS which uses CPABE. It constructs a hybrid scheme, in which attributes come from different authority sets and multiple authorities in an authority set jointly maintain a subset of the whole attribute set	It does not describe the reasons for selecting a threshold value for secret sharing; attribute set and the master key for optimised interaction design protocols which need to be addressed
Encrypted data management with deduplication in cloud computing [11]	Current industrial deduplication solutions cannot handle encrypted data. Existing solutions for deduplication are vulnerable to brute-force attacks and cannot flexibly support data access control and revocation. The proposed scheme supports to deduplicate encrypted data stored in the cloud and also secure data access control	This proposal did not evaluated the development of a flexible solution to support deduplication and data access controlled for the efficient data ownership verification and scheme optimisation
Dynamic certification of cloud services: trust, but verify [4]	It explores the dynamic certification requirement to ensure continuously secure and reliable cloud services and establish trustworthy cloud service certifications	Practical evaluation and real-time scenarios are not discussed
Cloud data security with hybrid symmetric encryption [6]	Prime requirement of security concerns needs to take care. Cloud is untrusted the third party, holding the data which may be confidential and sensitive. Security and privacy of data are a prime concern	The practical evaluation of the proposed work not evaluated for the encryption or decryption process
Attribute-based data sharing scheme revisited in cloud computing [1]	It enhances data confidentiality and privacy in cloud system against the managers of KA and CSP as well as malicious system outsiders, where KA and CSP are semi-trusted. Also, the weighted attribute was to improve the expression of the attribute, it also reduces the complexity of AP	Data confidentiality and privacy are ensured in the proposal, but the trust of the data owner and data requester not evaluated. It can use to enhance to assess the trust management models for better cloud security

(continued)

Table 1 (continued)

Paper title	Advantages	Limitations
An efficient privacy-preserving ranked keyword search method [2]	The proposed method has an advantage over the traditional method in the rank privacy and relevance of retrieved documents. It analyses the search efficiency and security under two popular threat models	The work has focused on data privacy only, but the impact of different search users and their accessing privacy not discussed
A semi-automatic and trustworthy scheme for continuous cloud service certification [5]	It increases the confidence of cloud customers that every piece of the cloud behaves as expected and according to their requirements. It defines an automatic approach to verification of consistency between requirements and models, which is by the chain of trust supported by the certification scheme	The scheme does not consider the composite service certification and cost-effective certification-based service composition
A secure anti-collusion data sharing scheme for dynamic groups in the cloud [12]	This scheme can support dynamic groups efficiently when a new user joins the group, or a user revoked from the group, the private keys of the other users do not need to be recomputed and updated	The users can revoke even though they are conspiring untrusted. User trust evaluation not considered before revoking into the group
Two-factor data security protection mechanism for cloud storage system [9]	This solution offers revocability of the device along with the confidentiality of data. The corresponding ciphertext updated automatically by the cloud server without any notice of the data owner once the device revoked	The work mostly focuses on the confidentiality of the data and device control, but it has not evaluated the untrusted user, and the verification user identifies falsification
Towards achieving data security with the cloud computing adoption framework [13]	CCAF multi-layered security can protect the data centre from the rapid data growth due to the security breach. The approach could provide real-time protection of all the data, block the majority of threats and quarantine the petabyte systems in the data center	The approach provides an integrated solution to cloud security based on a clear framework, business process modelling to study the impact on the performance of a user accessed service

Table 2 Comparing data sharing schemes

Scheme	ABE type or certificate	Policy expression (monotone/monotone)	Type of revocation	Backward security	Forward security
TMACS scheme [10]	ABE	Monotone	Multiple authorities	×	✓
Encrypted data management [11]	ABE	Monotone	–	×	✓
Hybrid symmetric encryption [6]	Certificate	Nonmonotone	Cloud service provider	×	✓
Attribute-based data sharing scheme [1]	ABE	Monotone	Trusted key authority	×	✓
A semi-automatic and trustworthy scheme [5]	Certificate	Nonmonotone	Certificate authority	×	✓
Secure anti-collusion data sharing scheme [12]	Certificate	Nonmonotone	User revocation	✓	✓
Two-factor data security protection mechanism [9]	Certificate	Nonmonotone	Online authority	×	✓
Multi-authority CPABE scheme [13]	ABE	Monotone	User revocation	✓	✓
Access control with dynamic policy update [14]	ABE	Monotone	User revocation	×	✓

data to determine a medication's side effects. Data consumers might also have privacy concerns. For example, researchers working on inventions might want their identities and a query protected so that no one uncovers what they are working on or steals their ideas. *Service providers* include all IT staff required to run and manage cloud services, including databases, servers, networks and applications.

One source of privacy concern is cloud administrators, who are not under the control of data owners or consumers. They might accidentally or deliberately disclose data, with unwelcome consequences. For example, they could reveal sensitive information such as unsuccessful medical treatments, ongoing research, or patients' illnesses to employers, insurance companies, or competitors. It could irreversibly

damage patients, physicians and scientists. It is challenging to identifying trustworthy cloud services because of their diversity and the similar functionalities they provide. Access control in cross-domain is an important research direction.

Trust is the core issue of access control in CC. Trust mechanism is incorporated in the access control model which will be redefined and calculated. A popular way to establish trust among independent entities is to control end-user authorisation through a privacy policy. Numerous researchers have explored trust management in Web services. However, most studies preceded the recent surge in cloud computing and thus do not reflect state of the art. Moreover, they do not cover many issues such as distrusted feedback, poor identification of trust feedback, trust participant privacy and lack of trust feedback integration.

Figure 2 shows trust management can be viewed from the perspective of either the service provider, who wants to assess service requesters' trustworthiness, or the service requester, who wants to evaluate service providers' trustworthiness. Trust management can be based on policies, recommendations, reputation or prediction. Here, it looks at the problem of managing trust in the cloud holistically. It describes different trust management perspectives and techniques, identifies the trust characteristics of cloud service providers and proposes a generic analytical framework to assess trust management systems in cloud computing.

The service for trust computation and management can be independent of cloud services. For the underlying cloud service models (IaaS, PaaS or Saas), different parameters like security, performance, compliance assessment function and trust techniques must be compatible. It argues that it is vital to consider the possible trust management perspectives and techniques to recognise the types of cloud services that these techniques support and to develop the most suitable trust management system per cloud service type. There is a need for the research work which allows the user to know their cloud service's privacy policy which can help to determine whether to trust that service with essential data. It provides insight on the privacy responsibility

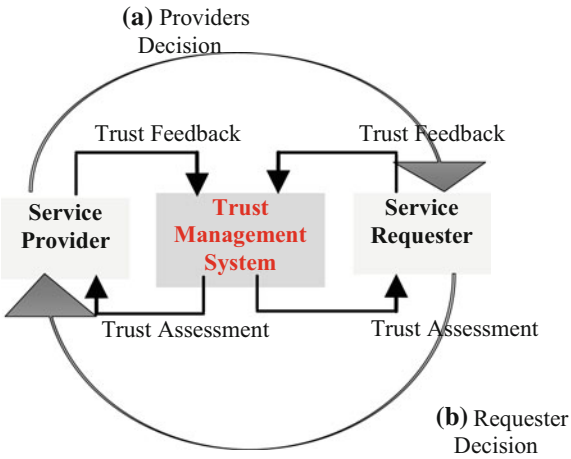


Fig. 2 Trust management from the perspectives of the **a** service provider and **b** service requester

which can split between the provider, who deploys all necessary security measures and consumers, which helps to take their steps to preserve data privacy.

5 Open Issues and Challenges

Cloud computing is a highly promising technology, but deficient trust management is hindering the market growth. There is a need to design a framework for analysing trust management systems which can help researchers to develop innovative solutions and improvements to challenges such as identification, privacy, personalisation, integration, security and scalability.

However, several recent surveys [1, 4, 6, 12] show that 88% potential cloud consumers are worried about the privacy of their data, and security is often cited as the top obstacle to cloud adoption. The risk of confidential data leakage and privacy violations in the cloud significantly hinders a wide adoption of this technology [2, 3]. Privacy and security of data is a prime concern in cloud computing data storage [10, 9]. Data stored at cloud server may be confidential also which require more security consideration. Cryptography techniques [6, 7, 3] provide a secure way for confidential data storage at a third party.

The biggest problem in the cloud computing is sharing the resources [1, 12] from the trusted authority. The second problem is to provide data access effectively when the service requested for the resources by the authority, and the third problem is providing security to the published or stored data from the third-party vendors access. Therefore, to trust on a cloud, they need to have careful system design, review of existing standards, risk management and requirements analysis before deploying services in the cloud. There is a need to address the backward security issues which are not yet handled properly by the current system.

6 Conclusion

The comparative analysis and literature review identifies the research gaps. There is a need to investigate and improvise system design and existing standards. The first need is to *improvise cloud storage service privacy*. In many existing cloud storage systems, data owners have to assume that the cloud providers are trusted to prevent unauthorised users from accessing their data. This untrusted privacy concern needs to be a target. The second need is to *improvise the cloud data accessing policies.* The current practice of privacy protection makes authorised users satisfy by the access policies. Using the private key, user can decrypt the data content, but the unauthorised user cannot. Therefore, the mess of managing access to data stored in the cloud leads to the issue of key management which in turn is determined by the access policies. The third need is to *integrate of trust computing and to access control for identifying cloud reliability*. Most of the current systems apply role-based access

control, which is suitable for homogeneous environments, where similar nature of data, roles and tasks allows doing so. However, in systems deployed in the cloud and spanning multiple countries, support for more diverse objects, users and rules must be considered.

References

1. Wang, K. Liang, J.K. Liu, J. Chen, J. Yu, W. Xie, Attribute-based data sharing scheme revisited in cloud computing. IEEE Trans. Inf. Forensics Sec. **11** (2016)
2. C. Chen, X. Zhu, P. Shen, J. Hu, S. Guo, Z. Tari, A.Y. Zomaya, An efficient privacy-preserving ranked keyword search method. IEEE Trans. Parallel Distrib. Syst. **27**(4) (2016)
3. M. Marwahe, R. Bedi, Applying encryption algorithm for data security and privacy in cloud computing. Int. J. Comput. Sci. **10**(1), 367–370 (2013)
4. S. Lins, P. Grochol, S. Schneider, A. Sunyaev, Dynamic certification of cloud services: trust, but verify. IEEE Comput. Reliab. Soc. 16, 1540–7993 (2016)
5. M. Anisetti, C. Ardagna, E. Damiani, F. Gaudenzi, A semi-automatic and trustworthy scheme for continuous cloud service certification. IEEE Trans. Serv. Comput. (2016)
6. Kaushik, C. Gandhi, Cloud data security with hybrid symmetric encryption, in *IEEE International Conference on Computational Technology in Infomatics and Communication Technologies* (2016)
7. S. Wang, J. Zhou, J.K. Liu, J. Yu, J. Chen, W. Xie, An efficient file hierarchy attribute-based encryption scheme in cloud computing. IEEE Trans. Inf. Forensics Sec. **11**(6) (2016)
8. C. Yanli, S. Lingling, Y. Geng, Attribute-based access control for multi-authority systems with constant size cipertext in cloud computing computing. IEEE China Commun. **13**, 146–162 (2016). https://doi.org/10.1109/cc.2016.7405733
9. J.K. Liu, K. Liang, W. Susilo, J. Liu, Y. Xiang, Two-factor data security protection mechanism for cloud storage system. IEEE Trans. Comput. (2015)
10. W. Li, K. Xue, Y. Xue, J. Hong, TMACS: a robust and verifiable threshold multi-authority access control system in public cloud storage. IEEE Trans. Parallel Distrib. Syst. (2016)
11. Z. Yan, M. Wang, Y. Li A.V. Vasilakos, Encrypted data management with deduplication in cloud computing. IEEE Cloud Comput. Comput. Soc. 16, 2325–6095 (2016)
12. Z. Zhu, R. Jiang, A secure anti-collusion data sharing scheme for dynamic groups in the cloud. IEEE Trans. Parallel Distrib. Syst. **27**(1) (2016)
13. V. Chang, M. Ramachandran, Towards achieving data security with the cloud computing adoption framework. IEEE Trans. Serv. Comput. (2015)
14. K. Yang, X. Jia, K. Ren, Secure and verifiable policy update outsourcing for big data access control in the cloud. IEEE Trans. Parallel Distrib. Syst. **26**(12) (2015)

Cloud Scheduling Using Improved Hyper Heuristic Framework

Akhilesh Jain and Arvind Upadhyay

Abstract Effective scheduling is a main anxiety for the execution of performance motivated applications. Cloud Computing has to work with the large number of tasks. The question arises, How to make appropriate decisions, while allocating hardware resources to the tasks and dispatching the computing tasks to resource pool that has become the challenging problem on cloud. In cloud environment task scheduling refers to an allocation of best suitable resources for the task which are executing with the consideration of different characteristics like makespan, time, cost, scalability, reliability, availability, resource utilization and other factors. We had tried to find the right method or sequence of heuristic in a given situation rather than trying to solve the problem directly. To check the importance of proposed algorithm we had compared it with the existing algorithms which had provided the far better results. We have introduced the improved hyper heuristic scheduling algorithm with the help of some efficient meta-heuristic algorithms, to find out the better task scheduling solutions for cloud computing systems and reduced the makespan time, and enhanced the utilization of cloud resources.

Keywords Cloud computing · Task scheduling · Meta-heuristic · Makespan Hybrid heuristic · Hyper heuristic · Min-min · Max-min · ACO · PSO

1 Introduction

Cloud computing paradigm is an abstract framework for giving effective and low cost computing as a service to the client. Resource organization and task scheduling are the key technology in cloud computing environment. It is the process of resources distribution and utilization in a limited environment.

A. Jain (✉) · A. Upadhyay
Department of Computer Science and Engineering, IES-IPS Academy, Indore, India
e-mail: jainakhil615@gmail.com

A. Upadhyay
e-mail: upadhyayarvind10@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_15

To search an optimal solution to schedule a given set of tasks T = {T1, T2, T3, …, T_n} to a given set of machines M = {M1, M2, M3, …, M_m} subject, which is a predefined set of checks and dimensions. For an example, makespan is the commonly used for scheduling problems measurement $C_{max}(S)$, which is well-defined as the finishing time of the last task. Arithmetically [1], the problem is-

$$minimize\ f(s) = C_{max}(S)$$

1.1 Scheduling Methods for Cloud Computing

In the cloud environment most traditional scheduling algorithms are wide used on today's cloud computing systems as a result of they are simple and easy to implement. But not appropriate for large-scale or complex scheduling problems.

After that Meta-heuristics algorithms which are very popular on scheduling and successfully work as compared to traditional scheduling algorithm. But they suffers from the following problems-Slow response to changing conditions, Algorithms may trap in local optimum.

The performance can increases by the new algorithm that is Hybrid meta-heuristic algorithm. But they introduces their own difficulties, Because of different convergence rate the convergence of the whole algorithm is unpredictable.

2 Problem Identification

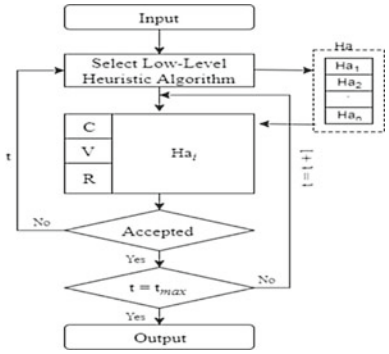
In the study of literature hyper heuristic methodology is best for optimization of task scheduling as compare to other scheduling algorithms. We focus on minimizing scheduling computational time by proposing algorithm Improved Hyper Heuristic Scheduling Algorithm (IHHSA) to achieve following objectives.

- To improve task secluding on cloud.
- To reduce the makespan.
- To implement propose hyper heuristic mathematical model.

3 Proposed Methodology

The methodology we are offering here follows the following steps written below:

First of all randomly choice the low level heuristic (LLH) algorithm from the algorithm pool which is available to us. In heuristic pool we have Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), Min-Min, Max-Min, FCFS algorithms.



1. Parameters setting
2. Input the problem
3. Initialize population Z
4. Randomly select LLH from heuristic pool Ha_i.
5. While termination condition is not met.
  a. Update the population Z by selected heuristic
  b. X1: Intensification Detection operator on Z
  c. X2: Diversification Detection operator on Z
  d. If f(Ha_i, X1, X2)
    Randomly select new Ha_i

Fig. 1 Proposed architecture and algorithm

Intensification Detection Operator (IDO):
In this operator a random technique is used to choice the low-level heuristic Ha_i from the heuristic pool *Hai*. These function only checks to see whether the solutions found by Ha_i, i.e., (Best so far makespan), are being improved or not.

Diversification Detection Operator (DDO):
After the IDO, the DDO is used by proposed algorithm to decide "when" to change the low-level heuristic algorithm Ha_i.

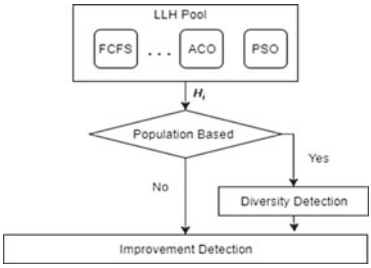


Fig. 2 Operator selection architecture

Develop the operators for intensification and diversification. Perform validation on selected LLH by proposed operators.

We examine the performance of propose algorithm by calculating the makespan. Heuristics used three main operators are Conversion (C), Valuation (V) and Resolve (R). Proposed Architecture and algorithm given in Fig. 1.

Function f (Ha_i, X1, X2) is determine whether selected algorithm continue or to change new heuristic as defined below.

- If $Ha_i \in$ S and X2 is true, return false, If $Ha_i \in$ P and X1 is true, X2 is true, return false, Otherwise return true.

Where S is SSBHA and P is PBHA, if function return true means Select new LLH (Fig. 2).

4 Result

4.1 Parameter Settings

Emulating a cloud computing environment we have used the CloudSim tool. It is used as an emulator to determine whether cloud task scheduling problem is bean solved or not. CloudSim can be used to construct a data center with a set of virtual machines as the resource (Table 1).

4.2 Simulation Results

To evaluate the performance of proposed algorithm for the cloud task scheduling problem, we had compared it with three traditional algorithms and two meta-heuristic algorithms on j30, j60 and j90 datasets. The comparison result had been shown in Table 2, and we found that IHHSA provided the better results than that of the other algorithms.

Results had shown that the implemented algorithm is better than all of the other scheduling algorithms. It also showed that the traditional algorithms produce the similar makespan for all the iterations because it is single solution based heuristic algorithm (SSBHA) (Fig. 3).

All though it has been shown that in Fig. 4 as the majority of task increases with the convergence speeds of proposed algorithm it gets much better, in terms of results, Propose algorithm has a higher chance to search out a better result than that the other heuristic scheduling algorithms. We have also done the comparison between proposed algorithm and existing algorithms as shown in Fig. 3.

Table 1 Cloudsim parameters configuration

Cloudlets (task)		VMs		Data centers	
Length	4000	No of VMs	5	No of data centers	2
File size	300	RAM	512 MB	RAM	3 GB
Output size	300	PES	2	Storages	1,000,000 MB
PES	2	Bandwidth	1000	Bandwidth	10,000
		MIPS	100	MIPS	1000

Table 2 Makespan on 1000 iterations

Datasets	Iterations	FCFS	MIN-MIN	MAX-MIN	PSO	ACO	IHHSA
j30	100	22.26	17.33	16.13	19.08	9.70	8.49
	200	22.26	17.33	16.13	19.08	9.60	8.33
	400	22.26	17.33	16.13	18.84	8.74	7.95
	600	22.26	17.33	16.13	17.88	8.54	7.94
	800	22.26	17.33	16.13	18.67	8.54	7.80
	1000	22.26	17.33	16.13	16.37	8.00	7.81
j60	100	91.86	60.26	58.13	43.76	43.60	44.69
	200	91.86	60.26	58.13	45.68	44.40	45.12
	400	91.86	60.26	58.13	43.71	44.50	42.92
	600	91.86	60.26	58.13	43.70	44.80	43.62
	800	91.86	60.26	58.13	46.76	44.80	45.76
	1000	91.86	60.26	58.13	45.17	44.00	43.88
j90	100	209.46	141.33	138.13	90.28	93.80	89.50
	200	209.46	141.33	138.13	89.01	92.60	87.88
	400	209.46	141.33	138.13	92.70	90.25	87.04
	600	209.46	141.33	138.13	91.38	88.20	84.87
	800	209.46	141.33	138.13	90.16	86.26	82.89
	1000	209.46	141.33	138.13	90.78	83.00	80.27

5 Conclusion

In the Previous research most of the researchers had focused on reduction of makespan time, execution cost and average resource utilization. The proposed algorithm not only provide better results than the traditional rule-based scheduling algorithms, it also performs much better than that of the other heuristic scheduling algorithms. The main aim of the implemented algorithm is to control over the strengths of all the low-level heuristic algorithms by not increasing the computation time, also by running one and only one LLH algorithms at each iteration. The implemented algorithm may be applied to different cloud computing environment to improve their performance of scheduling problems. It also significantly decrease the makespan of task scheduling.

In the future, the focus will be done upon the finding more effective detection operators and apprehension method for the best performance of the proposed algorithm.

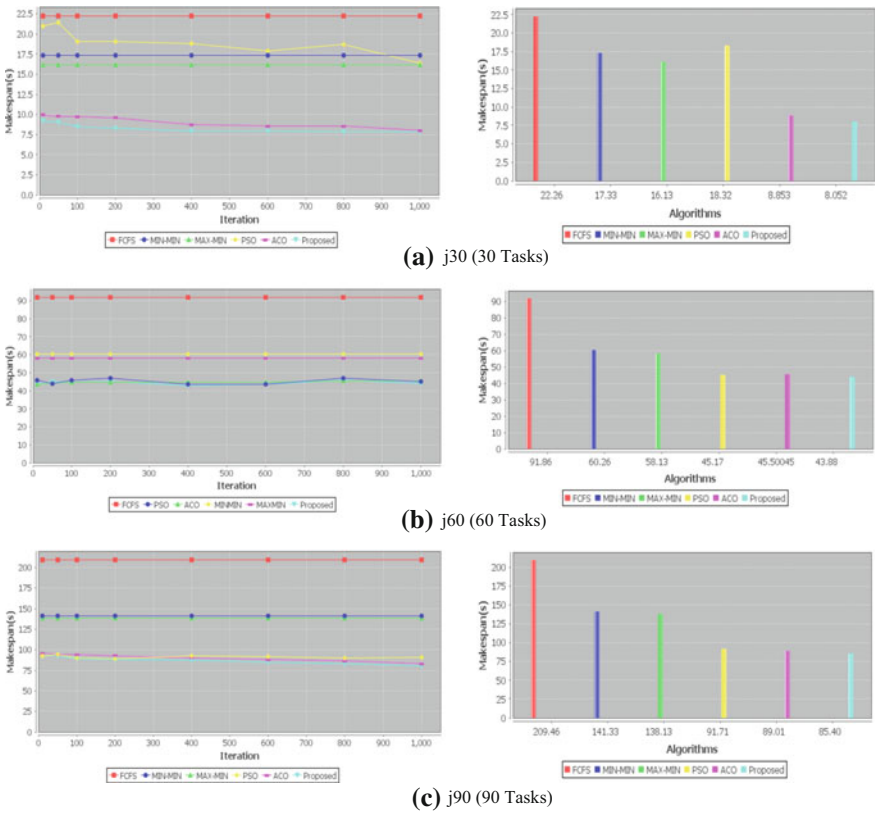


Fig. 3 Analysis of best Makespan on 1000 iteration

References

1. M. Kalra, S. Singh, A review of metaheuristic scheduling techniques in cloud computing. Egypt. Inf. J. **16**(3), 275–295 (2015)
2. S. Kumar, R. H. Goudar, Cloud computing—research issues, challenges, architecture, platforms and applications: a survey. Int. J. Future Comput. Commun. **1**(4) (2012)
3. A. Battou R. Bohn, J. Messina, M. Iorga, M. Hogan, A. Sokol, NIST senior advisor for cloud computing, https://www.nist.gov/programs-projects/cloud-computing
4. S. Devipriya, C. Ramesh, Improved Max-Min heuristic model for task scheduling in cloud, in *Green Computing, Communication and Conservation of Energy (ICGCE), International Conference*, Dec 2013
5. J. Gu, J. Hu, T. Zhao, G. Sun, A new resource scheduling strategy based on genetic algorithm in cloud computing environment. J. Comput. **7** (2012)
6. K. Zhu, H. Song, L. Liu, J. Gao, G. Cheng, Hybrid genetic algorithm for cloud computing applications, in *Services Computing Conference (APSCC)* (IEEE Asia-Pacific, 2011)
7. K. Li, G. Xu, G. Zhao, Y. Dong, D. Wang, Cloud task scheduling based on load balancing ant colony optimization, in *IEEE Sixth Annual China Grid Conference*, Aug 2011
8. Z. Pooranian, M. Shojafar, J.H. Abawajy, A. Abraham, An efficient meta-heuristic algorithm for grid computing. J. Comb. Opt. **30**(3), 413–434 (2015)

9. X. Wen, M. Huang, J. Shi, Study on resources scheduling based on ACO algorithm and PSO algorithm in cloud computing, in *11th International Symposium on Distributed Computing and Applications to Business, Engineering & Science*, Oct 2012
10. S. George, Hybrid PSO-MOBA for profit maximization in cloud computing. Int. J. Adv. Comput. Sci. Appl. (IJACSA) **6**(2) (2015)

An Approach for Grammatical Inference Based on Alignment of Slot

Manish Pundlik, Kavita Choudhary and G. L. Prajapati

Abstract In this paper, we consider the unsupervised grammatical inference problem. To determine the grammatical structures consistent with the input examples, a scheme slot alignment score which constituted from statistical information suggested by examples is illustrated. The system finds the slot alignment profile and formulates fuzzy similarity to quantify the slot alignment profile for a given collection of sentences. We give a methodology of slot alignment score that acquires more correct context-free grammar without any domain knowledge and useful for learning language structure.

Keywords Grammatical inference · Slot alignment · Fuzzy similarity
Context-free grammar · Language structure

1 Introduction

We consider the learning of context-free grammars from input examples. Identifying a grammar from the sample texts is an important topic in grammatical inference. Grammatical inference entails inferring a grammar or inducing production rules from a set of observation in a language by using machine learning techniques. Machine learning is a scientific field addressing the question how can we program system to automatically learn and to improve with experience. The problem of inferring grammars has been studied during the past few decades either as a mathematical problem where the goal is to find some hidden function or a practical problem of endeavoring to represent some knowledge about language structure. Building algorithms that learn context-free grammar or more restricted grammar is one of the open, crucial, and computationally hard problems in grammatical inference.

M. Pundlik · K. Choudhary (✉) · G. L. Prajapati
School of Computers, IPS Academy, Indore, India
e-mail: kavitachoudhary@ipsacademy.org

M. Pundlik
e-mail: manishpundlik@ipsacademy.org

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_16

In recent years, there have been significant advances in the field of unsupervised grammar induction for natural languages. In this respect, several systems are emerging, which are built using complex models and are capable of deriving language grammatical phenomena. These systems can be grouped into the following categories: models based on categorical grammar (GraSp, CLL, EMILE) [1–3]; memory-based learning models (FAMBL, RISE) [4, 5]; evolutionary computing models (ILM, LAgts) [6, 7]; and string pattern searches (ABL, GB) [8, 9], CCM model [10].

Alignment-based learning (ABL) [9] is based on the alignment of sentences and substitutability. In the alignment phase, the matched parts of sentences are considered as possible constituents. Equal parts and unequal parts are used for generating context-free grammar production rules. Nonterminals are assigned as the left-hand sides of the production rules with the unequal parts as the right-hand sides are shown by the following example

[Jon] sees the [large, green] apple.
[Jim] sees the [red] apple.

The underlined parts located above each other in the alignments are called equal parts, and unequal parts considered as potential constituents are called substitutions. If there is a substitution, then the two substitutions are said to be aligned in the same slot, where slot denotes location of subsentence in the alignment. Jon and Jim are considered in the same slot and large, green and red in same slot shown in brackets. The created grammar rules are:

S → NT_A sees the NT_B apple
NT_A → Jon
NT_A → Jim
NT_B → large, green
NT_B → red

where S is starting symbol in the CFG and NT_A and NT_B are the nonterminals assigned. However, two problems remain open in the ABL framework according to Zaanen [11]. We need words in the two sentences having exact match. For example, we have the following sentences in these sentences. “Book” and “Show” are of the same type, but standard ABL cannot learn this concept and unable to identify the following alignment

Book [a trip to Goa beach].
Show [me Big Bird’s house].

Another problem is that words of different types, if found in the same context, have recognized as of the same type. Thus, the words “apple” and “well” have determined as of the same type in the following:

Ram eats [apple].
Ram eats [well].

We address these problems in our study and improve alignment by using slot alignment profile similarity based on statistical information [12].

Preliminaries

A grammar is a quadruple $G = (N, \Sigma, P, S)$, where N and Σ are the alphabets of nonterminals and terminals, respectively, such that $N \cap \Sigma = \emptyset$. P is a finite set of productions, and SN is a special nonterminal called the start symbol. A grammar $G = (N, \Sigma, P, S)$ is called context-free if all production rules are of the form $A \rightarrow \alpha$, where $A \in N$ and $\alpha \in (N \cup \Sigma)^*$. In the production rule $A \rightarrow \alpha$, A is called the left-hand side (LHS), and α is the right-hand side (RHS). We define A-production to be a production with LHS as the nonterminal A. A language L is a context-free language if there exists a context-free grammar (CFG) G such that $L = L(G)$. Stochastic context-free grammar (SCFG) is a context-free grammar G that includes an assignment of weights, with each weight being between zero and one (inclusive), to the productions; such that, for every nonterminal A, the sum of weights of all the A-productions in G is 1. The assigned weights are also called probabilities, and it is denoted as $A \rightarrow \alpha$ (pr), where pr is the assigned weight.

2 Proposed Scheme

2.1 Preprocessing

We consider the sample texts of the target language as input. For a given sample string α, a production has added to the production set as $S \rightarrow \alpha$, where S is starting symbol. The input is preprocessed for finding prelude grammar. To begin with, a simple CFG has built by creating a production rule with the LHS as the grammar start symbol and the RHS as each sentence. Therefore, the built grammar exactly generates the given sample set. For example, we create initial production rules like:

$S \rightarrow$ [She][loves][her][job]
$S \rightarrow$ [You][like][reena][shoes].

After preliminary grammar, we prepare a word set which contains occurrences of each word, called word structure set (WSS). This concept borrows from Nevil—Manning and Witten's approach to make use of statistical information [12], for slot alignment score calculation. This improves the accuracy of identifying grammar.

2.2 Alignment Process

Starting from the simple CFG, for each pair of RHS on the production rule we do alignment for finding best alignment. For pairwise alignment, we use the dynamic programming approach [7] similar to ABL framework. One simple strategy is to find the alignment which is the longest common subsequence (LCS) in which we find the longest matching sequence in two sentences and generate a matrix that store alignment scores of each paired sentences. It is a simple arrangement of paired sequence (example given in introduction).

For finding alignment, we use the recurrence relation and create a matrix that stores score of pairing sequences called substitution matrix.

$$\text{opt}[i][j] = \begin{cases} 0 & \text{if } i = M \text{ or } j = N \\ \text{opt}[i+1][j+1] + 1 & \text{if } x_i = y_j \\ \max\left(\text{opt}[i][j+1], \text{opt}[i+1][j]\right) & \text{otherwise} \end{cases} \quad (1)$$

The goal of this step is finding a maximum match of two aligned sentences. Therefore, alignment scores are the number of matches of the sentence pair. Induction of production rules starts with alignment and alignment scores. From improving alignment results, we use dynamic sentence similarity threshold given by Chaudhari and Wang [13].

2.3 Dynamic Sentence Similarity (DSS)

Sentence pairs have different-different sentence similarity some; the sentence has maximum sentence similarity that is exceeding from one of the sentence lengths. For example, the following are nonmatching sentences with high similarity measure.

They all [have] to stay in order.
This time they are all [going] to stay in order.

Some have minimum similarity, which produce incorrect result because of misleading alignment. For example, the following are "misleading alignments."

[Thank] you for showing me the games.
[How are] you?

Due to this reason which pairs proceeds, it decided at runtime called dynamic sentence similarity (DSS). For this, we use an algorithm designed by Wang and Chaudhari [5]. Algorithmic description is given below.

Procedure DSS.

```
var found: boolean;
     R_score,step_value,min_value,threshold: Real;
 begin
   found:= false;
      readln(simple  CFG);
      for each pair of CFG do following
      begin
          R_score=a_score/len1+len2;
      repeat
           threshold:=1;
      while(threshold >= min_value) do
          begin
            if(r_score>threshold) then
            begin
               GenerateNewRule(ref CFG);
               Break;
            End;
           Threshold:=threshold-step_value;
         End;
         until found new rule
   end;
end.
```

In algorithm, a_score is the alignment score of sentence pair and len1 and len2 are lengths of the sentences. The threshold first set of high value does the alignment. If no rule is generated, then threshold decreases until a new rule is generated. Whenever a new rule is generated, threshold is again set to high value. To prevent misleading result, a bottom threshold is defined. Our experiment results report for varying min_value and step_value. But we set min_value = 0.2 and step_value = 0.01 because it offers better results with minimum misleading alignment and precision 91.48%.

2.4 Slot Alignment Calculation

In ABL [9], only exact match infers rules, but in real-application language sample, it is quite common for the two sentences to have a similar grammatical structure but contain no words in common. Use of lexical or tagging information can help, like in the earlier case “Book” or “Show” are labeled that they are matched. To tackle this problem, we use statistical information about sentence pair, which represents possibilities and frequencies of two subsentences which are aligned as a match. Substances that are found in the same alignment slot multiple times are considered grammatically similar. Therefore, if component pairs in the same alignment slot are

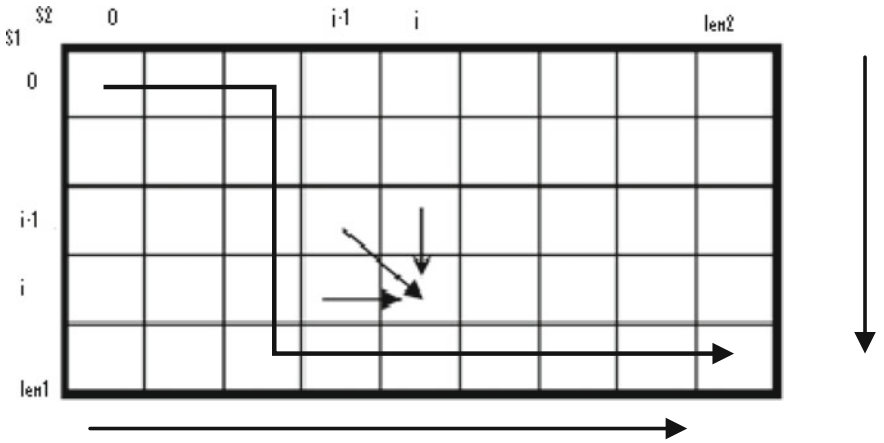


Fig. 1 Substitution matrix

marked, then they can help with further alignment. In this approach, we consider slot that follows one-to-one match. For example, we have two sentences in the sample set: {Ram loves apple, Shyam like mango}. Suppose further similarity of the words "loves" and "likes" are qualified, then they are considered as a match. Paths of the substitution matrix are used for calculation (Fig. 1).

$$N = (len1 + len2)!/(len1! * len2!); \tag{2}$$

$$Np = (i + j - 2)!/((i - 1)! * (j - 1)!); \tag{3}$$

$$Ns = (len1 - i + len2 - j)!/((len1 - i)! * (len2 - j)!); \tag{4}$$

len1 and len2 are sentence lengths of paired sentences. Therefore, the probability of aligning *i*th constituent of S1 and *j*th constituent of S2 using Eqs. (3) and (4) is given by:

$$Pij = 2 * Np * Ns/N; \tag{5}$$

Pij is the slot alignment score (SAS) of two constituents in the sentence pair which are one-to-one match.

Refinement Process

In alignment learning, words of different types are considered as the same type; for more accuracy, we refine the rule using a slot alignment score. The rules generated in alignment process are input; tokens that replaced with same nonterminals are possible constituents. For rectifying rules, we calculate profile similarity using statistical data. We generate a frequency matrix based on paired words which is found in the same alignment slot. This matrix has tokens with their slot score with another, which occur more than one time in sample or maybe only one time in the sample. Normalize this

matrix by dividing each score with the sum of all scores in each row; hence, the sum of all scores is normalized by 1. Such a matrix represents the alignment behavior of each token with another and hence called as alignment profile. Alignment profile is a fuzzy set; this concept is borrowed from Wang and Chaudhari [5].

Definition 1 Based on an SCFG G(N, Σ, P, S), an alignment profile is a fuzzy set AP defined on the set of symbols N ∪ Σ, denoted by AP = {<v, μAP(v)> | v ∈ N ∪ Σ}, where μAP(v): N ∪ Σ → [0, 1] is the membership grade of the fuzzy set AP.

After normalized profiles, we calculate similarity of the profile of each token with another using the formula used by [4]. Based on maxima (union of profile) and minima (intersection of profile) values of fuzzy set.

Definition 2 The profile similarity of alignment profiles AP_1 and AP_2, denoted as Ps (AP_1, AP_2), is formulated as:

$$Ps(AP_1, AP_2) = \frac{|AP_1 \cap AP_2|}{|AP_1 \cup AP_2|}$$

Using this matrix, we set the similarity threshold value according to profile similarity values. Each token may have 0 similarities with others and only one similarity with a token pair or multiple values; due to this reason, we use the following algorithm for similarity threshold value.

Procedure Similarity Threshold (ST) Based on Profiles

```
   var similiarity,min_th,threshold: Real;
       begin
           readln(matrix of profile);
             for each token value other than zero
              begin
                if(there is only one value other than
equal symbol  ) then
                          threshold := value of token;
                  else
                    begin
                         compute the minimum threshold as
min_th
                        if min_th> 0.80 then
                          threshold:=min_th;
               else
                begin
                 if(similiarity>=min_th) then
                   threshold:=similiarity;
                end;
            end;
          end;
end.
```

All the tokens in the matrix that have similar value with other token and have value greater than their threshold are considered as equal symbols and replaced with same conterminal.

2.5 Experimental Setup

The data set we used is from Child Language Data Exchange System database, which is a collection of sample sentences of child-related speech. We used the English-UK CHILDES data corpora. The number of sentences is over 500, and the average sentence length is 4. Since the system identifies subsentences through alignment which indicated interchangeability between aligned pair. Identifying pair is evaluated by slot. Multiple constituents aligned in same slot are considered, and corresponding slot that is paired multiple times is used for calculating slot score and stored in matrix for alignment profile. Sentence similarity changes according to generated rule; if no rule is generated, it decreases called dynamic sentence similarity threshold (DSST). More DSST is tested from 0.1 to 1. Our experiment results report for varying min_value and step_value. But we set min_value = 0.2 and step_value = 0.01 because it offers better precision 91.48%. Relative alignment score (r_score) method is better than alignment score method (a_score); it identifies more rule with higher accuracy. Figure 2 shows two curves of threshold in iterations. Threshold up and grow according to iterations; it usually originates from high value and drops to a lower value when rule formulation takes place. In our experiments, it takes seven iterations for grammar formulation. In refinement process, we are using ST based on the profile. In experiment, we consider that threshold not less than 0.6 means higher percentage of profile similarity; if the profile matches are less than 60%, then it is not considered as match and not replaced with same nonterminal. Most of the time if the constituent is matched in sample text and match percentage is 80, then they are replaced with same nonterminal.

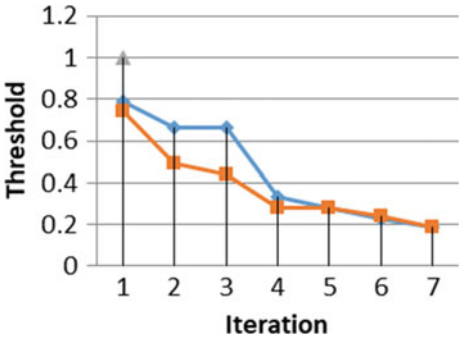


Fig. 2 Threshold wave in DSST

2.6 Conclusion

Inducing grammar rules from a given sample of target language based on alignment of sentences has been studied. We generate alignment profile using slot alignment (statistical information) of sample texts for optimizing alignment results. Here, we use one-to-one match slot alignment demonstrated by suitable example. Results can be improved further using one-to-many slot alignment.

References

1. G.L. Prajapati, N.S. Chaudhari, Learning alignment profiles for structural similarity measure, in *2012 7th IEEE Conference on Industrial Electronics and Applications (ICIEA)*
2. P. Adriaans, M. Vervoort, The EMILE 4.1 grammar induction toolbox, in *Proceedings ICGI (Lecture Notes in Computer Science)*, vol. 2484, pp. 293–295 (2002)
3. S. Kirby, Natural language from artificial life. Artif. Life **8**(2), 185–215 (2002)
4. P. Domingos, The RISE system: a case study in multistrategy learning. Technical Report 95-2, Department of Information and Computer Science, University of California (1995)
5. X. Wang, N.S. Chaudhari, Alignment based similarity measure for grammar learning, in *Proceedings IEEE International Conference Fuzzy Systems*, pp. 9034–9041 (2006)
6. E.J. Briscoe, Grammatical acquisition: "inductive bias and coevolution of language and the language acquisition" device. Language **76**(2), 245–296 (2000)
7. S.B. Needleman, C.D. Wunsch, A general method applicable to the search for similarities in the amino acid sequences of two proteins. J. Mol. Biol. **48**, 443–453 (1970)
8. M.A. Paskin, Grammatical bigrams, in *Advance in Neural information processing System*, ed. by T. Dietterich, S. Becker, Z. Gharahmani, vol. 14 (MIT Press, Cambridge, MA, 2001)
9. M.V. Zaanen, Implementing alignment-based learning, in *Proceedings ICGI (Lecture Notes in Computer Science)*, vol. 2484, pp. 312–314 (2002)
10. D. Klein, C.D. Manning, Natural language grammar induction with a generative constituent-context model. Pattern Recogn. **38** 1407–1419 (2005)
11. M.V. Zaanen, Theoretical and practical experiences with alignment based learning, presented at the Australas. Language Technology Workshop (Melbourne, Australia, 2003)
12. C. Nevil-Manning, I. Witten, Identifying hierarchical structure in sequences: a linear-time algorithm. J. Artif. Intell. Res. **7**, 67–82 (1997)
13. N.S. Chaudhari, X. Wang, Language structure using fuzzy similarity. IEEE Trans. Fuzzy Syst. **17**, 1011–1024 (2009)
14. P.J. Henrichsen, GraSp: grammar learning from unlabelled speech corpora, in *Proceedings of CoNLL-2002*, ed. by D. Roth, A. van den Bosch (Taipei, Taiwan, 2002), pp. 22–28
15. A.W. Biermann, Feldman, On the synthesis of finite state machines from samples of their behavior. IEEE Trans. Comput. C-21, 592–597 (1972)
16. C. de la Higuera, A Bibliographical study of grammatical inference. Pattern Recogn. **38**, 1332–1348 (2005)
17. M. Gold, Language identification in the limit. Inf. Control. **10**, 447–474 (1967)
18. M.V. Zaanen, Bootstrapping structure into language: alignment-based learning (Leeds, U.K., Univ. Leeds, 2002)
19. S. Watkinson, Manandhar 2001a (CLL) A psychologically plausible and computationally effective approach to learning syntax, in *CoNLL '01: The Workshop on Computational Natural Language Learning, ACL/EACL*

Online Model for Suspension Faults Diagnostics Using IoT and Analytics

Pravin Kokane and P. Bagavathi Sivakumar

Abstract Automobile world and technologies are advancing with rapid pace. A lot of resources are pouring into evolution of technologies considering safety, ride, and comfort of passengers. Suspension systems have also changed from just a mechanical assembly to active suspensions with multi-sensors for enhancing the actuations. Detecting faults in the suspension system early and categorizing them not only reduce the maintenance cost but add comfort and safety. In this paper, suspension faults have been studied and investigated. Approaches toward detecting suspension faults have been discussed. Internet of things and analytics-based online model for suspension fault detection are proposed.

Keywords Faults · Fault prediction · Machine learning · Suspension Fault detection · Internet of things

1 Introduction

Last decade brought surge in automotive technology improvements. These high-end vehicular technologies are changing the dimensions of markets. Vehicles with enhanced safety, ride, and comfort aspect are gaining large market share. In all these aspects, suspension system plays a vital role. The ride quality and effective isolation from uneven roads are greatly influenced by the suspension systems. Diagnosing the faults early not only avoids major fault but also improves comfortable ride. This also reduces maintenance cost. Drivers are able to diagnose the faults only if audible noise is associated with it or by drastic change in comfort levels. As per ISO 2631-1 (1997), vibration significantly influences human perception and ride comfort

P. Kokane
Department of Electronics and Communication Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore, India
e-mail: cb.en.p2ael16016@cb.students.amrita.edu

P. Bagavathi Sivakumar (✉)
Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore, India
e-mail: pbsk@cb.amrita.edu

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_17

[1]. Evidences from the analysis clearly demonstrate that resonant frequencies for suspension arm are different from upper mounting of shock absorber. It also indicates that suspension systems produce high amplitude vibrations between 0 and 20 Hz [2]. These low-frequency vibrations largely affect the human body [3]. Resonant frequency for different parts of various body sections lies between 2 and 90 Hz [4].

To reduce these vibrations, a lot of attention is already paid, and still improvements are going on. Present-day suspension systems are the evolved versions with improved safety, ride, and comfort aspects. Presently, suspension systems are electromechanical with various sensors and actuators to enhance the response time and performance. This complexity will give rise to parts failure, but inherent diagnostics methodology is possible by adding small infrastructure with it. The sensor data can be used in various ways to detect faults.

With the advances in Internet of things, the cost of gathering data is reduced, millions of nodes can be added to networks, and results can be enhanced. The enhancements brought in vehicular communications can connect vehicles to contribute the data for analysis. Vehicular data analytics is emerging as a field of its own. This helps in the fault identification and isolation, fault prediction, and preventive maintenance. Various kinds of models can be built from understanding the data. There are various approaches in machine fault diagnostics, in which presently fuzzy logic, machine learning, and data analytics are playing major role. For all results to arrive at certain conclusion, the feature engineering needs to be done. Feature engineering helps to find out conclusive feature, and it reduces computational cost and reduces dimensions of datasets.

Internet of things not only improves the data availability but also helps to achieve real-time diagnostics. With intelligent transport systems and cloud computing, the gathered data from each vehicle unit can be processed instantly and fault code messages can be relayed back to user. The research in vehicular to infrastructure (V2X) has provided the possibility of cloud-based diagnostics tools, to off-load work of onboard ECUs. Onboard equipment has limited computation cost as well as storage capacity; hence, it is good to depend on online expert systems. This can be useful in case of suspension faults because large amount of data can be processed and location-specific details can be considered (e.g., road profiles). Moreover, one can also use GPS and other technologies to connect with nearest service center in case of high fault severity. Amarasinghe et al. [5] have conceptualized the online expert system that can connect with service providers and workshops for faulty suspensions.

In this paper, system is proposed to detect suspension faults by determining the vertical accelerations of suspension system and other sensor inputs. The data is sent to cloud for further analytics, and expert online model will relay back with comparing database. Paper is organized as follows. Suspension systems and various suspension faults associated with it are discussed in Sect. 2. The literature review has been presented in Sect. 3 giving an overview of previous work in fault diagnostics of suspension and related systems. Section 4 deals with feature engineering. In this section, various methods of feature extraction and feature selection are discussed. Section 5 shows proposed online suspension for fault diagnostics with help of Internet of things. Section 6 gives conclusion and future scope.

2 Suspension System and Suspension Faults

It is system of various parts such as shock absorbers, springs, and dampers which connects chassis (or body) to the wheel. It should regulate the body movement by providing isolation of body from road disturbances, vibrations, and shocks. It allows rapid cornering without body roll. Suspension system is constituted by various parts such as control arm, steering knuckle, ball joints, springs, shock absorbers (dampers), control arm, and bushing. These parts are used in varied ways to form different types of suspensions.

On the basis of actuators and sensors used, suspension system can be classified as active, semi-active, and passive system. The suspension system is started with passive suspension and gone through series of advancements. Presently, most of the high-end vehicles are with active or semi-active suspensions. The active suspension provides more ride and comfort by varying stiffness. The vehicles having tight stiffness or soft stiffness will not be able to provide isolation from road. Stiffness also has its own impact on camber angle. This mainly affects the vehicle while cornering. The contact patch of tires degrades more in case of wrong settings of stiffness and camber angle. The trade-off between comfortable ride and low deformation of tires can be achieved with the runtime variation of stiffness. Hence, modern vehicles are majorly equipped with active/semi-active suspensions using electronics components.

2.1 Suspension Faults

Faulty suspension system cannot provide isolation from ground irregularities and will produce vibrations. As referred in previous section about human body resonance frequencies, these frequencies affect the human behavior. Only if major fault occurs in suspensions, user will notice abnormalities in ride, comfort, and ease. The abnormalities get unnoticed with novice driver and with smaller issues. Effective steering will also get affected with worse suspension. As every part of suspension creates different issues in case of failure, some faults may become severe during inattention period or multiple parts failure occurs. Table 1 shows a few types of suspension failure.

Actuator Faults—Moradi et al. [6] have proposed a fault-tolerant approach for actuator faults. With the seven degrees of freedom full-scale car model, even with 20% uncertainties, controller provides good performance. The adaptive approach helps to tune itself during fault. Gaspar et al. [7] used fault detection and identification filters to isolate the faults in actuators and in active brake arrangements to enhance rollover prevention. These faults can be communicated to the user to enhance operations.

Spring Faults—Yin and Huang [8] have worked on attenuation constant variations that occurs in springs; over the time, it reduces to certain extent and the spring fault occurs.

Table 1 Primary suspension faults and reasons associated with them

Part of suspension	Failure symptoms	Primary reason
Damper (shock absorbers)	Bounce and shakes in bumps	Shock fluid started to leak Worn-out shocks
Springs	One or more corner sits low Clunking noise over bumps	Spring wear, sag, or break
Ball joints	Squeaking and creaking noise while cornering	Ball joint break, wear
Control arms	Clunk and rattles	Bend of control arm
Rubber bushings	Ride and handling problems Imprecise steering	Wear
Strut	Bounce, sway, and shakes Camber/caster misalignment	Worn-out strut Bent strut
Nuts and bolts	Excess vibration and noise Vibrations at steering	Loose mounting of nuts and bolts
Tie rods	Vehicle pulling to one side, uneven tire wear, vibrations, steering issue, and handling problem	Tie rod wear

3 Literature Survey

This section discusses the various methods of analysis used for prediction and detection of suspension fault in various scenarios. There are two methods in fault diagnostics, namely data-driven and studying models. Study of suspension of vehicle is usually carried out with quarter car model [9]. In case of data-driven approach, machine learning plays an important role. Machine learning algorithms are broadly classified as supervised, unsupervised, semi-supervised, reinforcement, and transudation. Isermann and Wesemeier [10] have suggested clustering (unsupervised)-based fault diagnostics. Regression, i.e., supervised learning, has been widely applied [5, 8].

Data acquisition for suspension fault diagnosis is done by vibration's measurement. These vibration details can be captured by accelerometers. Accelerometers with two-axis measurement are usual, but three-axis accelerometers are increasingly common and inexpensive. There is evidential use of them in indirect tire pressure monitoring [10, 11] and in automobile gearbox fault diagnostics [12]. Kiencke et al. [13] have done spectral analysis of wheel speed along with neural network to monitor tire pressure. Even with varied road profiles, the obtained results were satisfactory.

4 Feature Engineering

In case of failure identification and predictive maintenance, acquired data needs to be preprocessed and then it is transformed to new set of variables to apply machine learning algorithms. Hence, feature engineering is an important aspect.

4.1 Signal Processing

Interpretation, generation, and transformation of raw and unprocessed data are known as signal processing. Frequencies associated with suspension faults are more important; hence, most of the scholars used fast Fourier transform, i.e., frequency domain analysis [9], or short-time Fourier transform, viz., time–frequency domain analysis. Conversion of vibration signals to frequency domain filters noise. Research has been done in bearing fault, railway suspension, actuators and springs of vehicle suspension fault detection using vertical accelerations, vibration measurements, and machine learning. These approaches can be used for suspension faults in vehicles.

4.2 Feature Selection and Feature Extraction

Feature selection selects the features which are most representative, whereas feature extraction transforms existing features into new set of features by combining them. Both methods reduce dimensions and computation cost associated with it. Some of the frequently employed methods are dynamic principal component analysis [8] and linear discriminant analysis [14, 15].

5 Online Suspension Fault Diagnostics

The section describes the online model proposed for fault identification and predictive maintenance for suspensions in automotive vehicles. It consists of vehicle unit, cloud server, data analytics, OEM research center, and service center.

5.1 Vehicle Unit

Figure 1 shows the online suspension fault diagnostics block diagram. The vehicle unit consists of accelerometers mounted on suspension systems, controller with global positioning system and communication unit. The existing sensors along with display will be available without any additional cost.

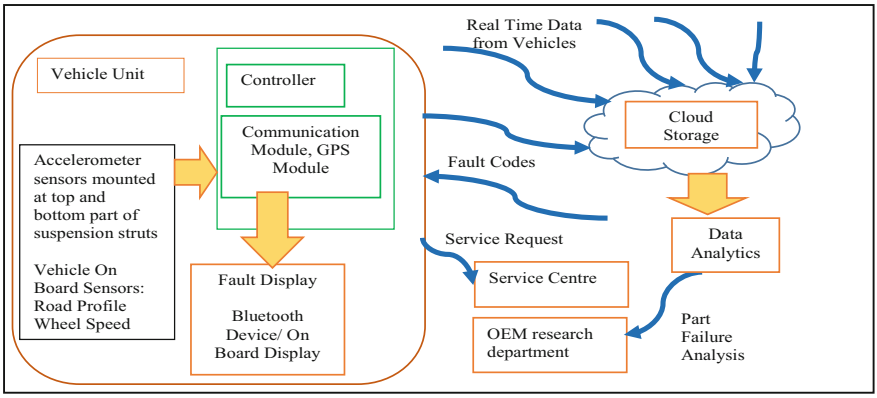


Fig. 1 Online suspension fault diagnostics

5.1.1 Accelerometers

Variable capacitance accelerometers have bandwidth of 0–1 kHz; lesser the bandwidth more the sensitive response. While mounting the accelerometers, precaution is always needed as it affects high frequency response and usable frequency range. In case of suspension faults, main concern is low frequencies where it will have little effect. For test setup, adhesive or magnetic mounting along with smooth surface can be used. Accelerometers mounted on rubber will have filtering effect; hence, it should be avoided. Mounting positions can be varied, at top of suspension, or at the base of the suspension strut. In data-driven approach, it is good to have more data which increases accuracy; hence, mounting of accelerometer at both positions is proposed.

In experimentation, we used triaxial accelerometer from PCB piezoelectronics. Nylon cube is useful for mounting accelerometer as it does not affect frequency vibrations.

Sensor Details: Triaxial ICP® Accelerometer

- Sensitivity: (±20%) 5 mV/g (0.51 mV/(m/s^2))
- Measurement Range: ±1000 g pk (±9810 m/s^2 pk)
- Broadband Resolution: 0.003 g rms (0.03 m/s^2 rms)
- Frequency Range: (±5%) 2–8000 Hz.

5.1.2 Data Acquisition

For low-frequency vibration data, the sampling frequency can be kept low. Craighead has considered 0–20 Hz frequency band, where suspension system has highest amplitude [16]. We require only 0–100 Hz band to cover edge frequencies and to increase accuracy. Hence, 200 samples/s/suspension is enough. Two accelerometers at each suspension (total 4), sending vertical vibrations with 5 bytes per sensor, will

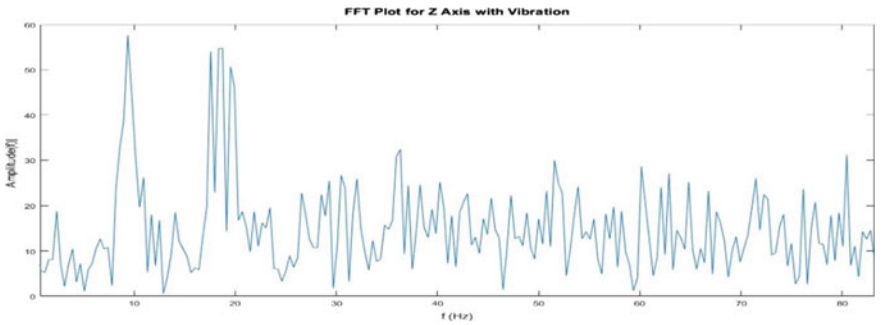


Fig. 2 FFT plot of Z-axis vibrations recorded using ADXL345

generate (4 * 2 * 5 * 8) about 320 bits/sample. Hence, 320 * 200, viz., 64 Kbps data transfer rate will be required. The low-cost ADXL345, three-axis sensor is sufficient and compatible with requirement. It has user selectable resolution of 10 bit–13 bit with 3.9 mg/LSB; hence capable of measuring inclination changes of less than 1.0° and frequencies up to 1000 Hz which is enough spectrum for condition monitoring of suspensions. Figure 2 is FFT plot of Z-axis for normal surface vibrations recorded using ADXL345.

5.1.3 Controller, Communication Module, and GPS Module

Controller constitutes the important part of the system. As the system will relay data collected, to cloud server, it will require minimal storage capacity, speed, and least code memory for trans-receiving data and display purpose. As previously explained, 64 Kbps requirement can be easily off-loaded with existing controllers or separate low-level controller can be used. With IoT and millions of node, the low unit cost is required and feasible. Data can be acquired and processed from accelerometers mounted at four wheels in round-robin fashion and other sensors data. Single controller will be sufficient without compromising accuracy, speed, and safety of passengers. As shown in Fig. 1, communication module sends out the data samples collected from accelerometers and other sensors and receives fault codes for display. The communication module can book service request in case fault severity is higher.

5.2 Cloud Storage and Data Analytics

Machine learning is widely used in various fault diagnoses and monitoring. K-means, KNN, and ANN have been applied in fault diagnosis. Machine learning models are formulated with data analytics tools using R or MATLAB. The predictive algorithms are applied on acquired data. Supervised (SVM, Random Forest, Naïve Bayes, etc.)

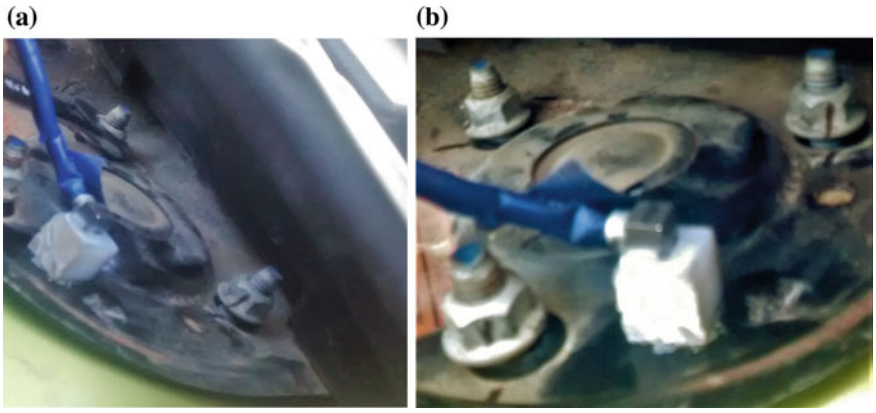


Fig. 3 **a** Unscrewing of suspension mounting and **b** unscrewed suspension mounting

algorithms with continuously acquired data along with labels train the model. Trained models classify faults with sampled data. Each vehicle unit will send data to common cloud storage/server. This data will be preprocessed, and signal processing can be applied. After feature engineering is applied to data, we can extract/select features. Machine learning techniques are applied on these short-listed features, to identify faults associated with it. If any of the faults are diagnosed, the fault code will be sent to that particular vehicle. This approach was proposed for water distribution systems, in smart cities [17]. This data can be sent to manufacturers to find frequent faults.

5.3 Experimental Conditions

Vehicle Model—Ford Figo, Left Hand, Front Wheel Drive
Sensor—ICP® Accelerometer—356A01, PCB Piezotronics, Inc.
DAQ—ArtemiS SUITE for sound and vibration analysis, HEAD acoustics GmbH
Sampling Frequency—8 kHz
Mounting of Accelerometer—Left and right top of suspensions

Figure 3 shows the unscrewing of suspension mounting.

6 Results and Discussions

After collecting data, we have applied FFT on each data samples. As we are mainly concerned with vertical vibration, Z-axis data is thoroughly analyzed. Figure 3 shows the unscrewing of suspensions, and Fig. 4a–d shows the FFT obtained.

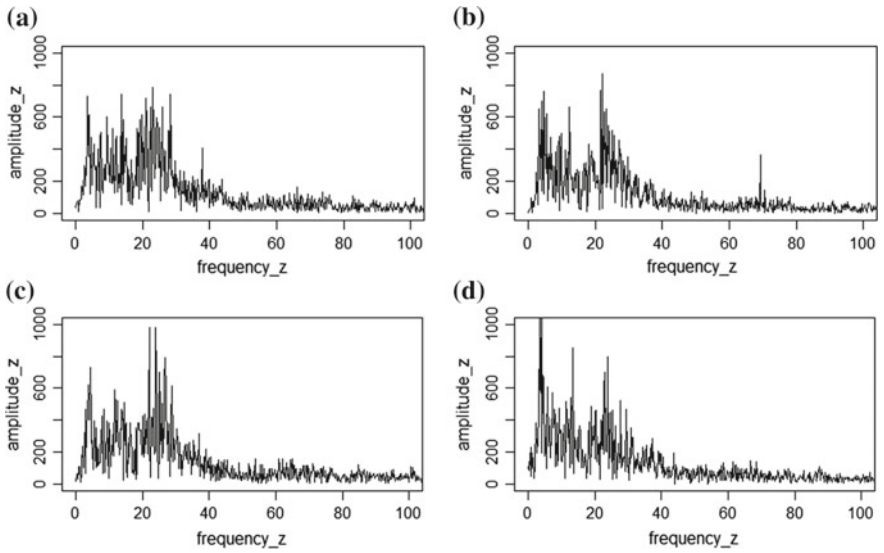


Fig. 4 **a** No fault introduced, **b** when single thread unscrewed, **c** when two thread unscrewed, **d** when three thread unscrewed

6.1 Observation

1. In case of no fault, the FFT amplitude is less than 800, whereas with unscrewed suspensions peak reaches to 1000 as it produces greater vibrations.
2. Major variations were seen in between 0 and 40 Hz; as discussed previously, human body resonance frequencies also lie in 0–100 Hz; hence, the vibrations if occurred due to faults need to be dampened with servicing and repair.

7 Conclusion and Future Scope

All the existing OBD tools and codes associated with it do not specify where the fault lies. They need extra device with expert to handle it. This existing limitation is addressed in this paper. As verified from results, the low-cost ADXL345 accelerometer can sense these vibrations with accuracy and can be useful in condition monitoring. This paper proposes use of ECUs, accelerometers, GPS, and other modules and requires a minimal cost and variation in system design. ECUs can communicate with cloud with V2X. Best results can be obtained with direct coupling with studs, and OEMs can quote this requirement with suspension systems vendors. Cloud knowledge database can be created for each user and vehicle model, and similar cases can be further analyzed to improve design of suspension systems. Each fault can be analyzed with different sets of machine learning algorithms and techniques.

OEMs can generate periodic reports to find out issues and improvements. Service centers can update user for requirement of maintenance and repair cost. Work can be extended further for tire faults, bust avoidance mechanism as they are also dependent on vertical accelerations. System can be incorporated in OBD.

References

1. International Standards, ISO 2631-1 Mechanical vibration and shock—evaluation of human exposure to whole-body vibration (1997)
2. R. Burdzik, L. Konieczny, Vibration issues in passenger car. Transp. Prob. **9**, 83–90 (2014)
3. Short Guide Human Vibration (Bruel & Kjaer, Denmark, 1999)
4. K. Ormuz, O. Muftic, Main ambient factors influencing passenger vehicle comfort, in *Proceedings of the 2nd International Ergonomics Conference* (Zagreb Croatia, Oct 2004), pp. 77–82
5. M. Amarasinghe, S. Kottegoda, A.L. Arachchi, S. Muramudalige, H.M.N. Dilum Bandara, A. Azeez, Cloud-based driver monitoring and vehicle diagnostic with OBD2 telematics, in *2015 IEEE International Conference on Electro/Information Technology (EIT)* (Dekalb, IL, 2015), pp. 505–510
6. M. Moradi, A. Fekih, Adaptive PID-sliding-mode fault-tolerant control approach for vehicle suspension systems subject to actuator faults. IEEE Trans. Veh. Technol. **63**(3), 1041–1054 (2014)
7. P. Gaspar, Z. Szabo, J. Bokor, Actuator fault detection for suspension systems, in *Proceedings of the 7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes* (Barcelona, Spain, June 30–July 3, 2009), pp. 1426–1431
8. S. Yin, Z. Huang, Performance monitoring for vehicle suspension system via fuzzy positivistic C-means clustering based on accelerometer measurements. IEEE/ASME Trans. Mechatron. **20**(5), 2613–2620 (2015)
9. M. Börner, H. Straky, T. Weispfenning, R. Isermann, Model based fault detection of vehicle suspension and hydraulic brake systems. IFAC Proc. **33**(26), 1073–1078 (2000). ISSN 1474-6670
10. R. Isermann, D. Wesemeier, Indirect vehicle tire pressure monitoring with wheel and suspension sensors, in *Proceedings of the 7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes* (Barcelona, Spain, June 30–July 3, 2009)
11. C. Halfmann, M. Ayoubi, H. Holzmann, Supervision of vehicles tyre pressures by measurement of body accelerations. Control Eng. Pract. **5**(8), 1151–1159 (1997)
12. T. Praveen Kumar, A. Jasti, M. Saimurugan, K.I. Ramachandran, Vibration based fault diagnosis of automobile gearbox using soft computing techniques, in *Proceedings of the 2014 International Conference on Interdisciplinary Advances in Applied Computing (ICONIAAC '14)*. (ACM, New York, NY, USA, 2014), Article 13, 7 pages
13. U. Kiencke, R. Eger, H. Mayer, Model based tire pressure diagnosis. IFAC Proc. **30**(18), 795–800 (1997). ISSN 1474-6670
14. X. Wei, S. Wu, J. Ding, L. Jia, Q. Sun, M. Yuan, Fault diagnosis for rail vehicle suspension systems based on fisher discriminant analysis, in *Proceedings of the 2013 International Conference on Electrical and Information Technologies for Rail Transportation (EITRT2013)*, vol. II, pp. 321–331
15. G. Wang, S. Yinn, Data-driven fault diagnosis for an automobile suspension system by using a clustering based method **351**(6), 3231–3244 (2014)
16. I.A. Craighead, Sensing tire pressure, damper condition and wheel balance from vibration measurements. Proc. Inst. Mech. Eng. Part D J. Autom. Eng. **211**(4), 257–265
17. P. Vijai, P. Bagavathi Sivakumar, Design of IoT systems and analytics in the context of smart city initiatives in India. Proc. Comput. Sci. **92**, 583–588 (2016)

Data Logging and Visualization Using Bolt IoT

Gaurav Prachchhak, Chintan Bhatt and Jaydeep Thik

Abstract The interconnection of ubiquitous and pervasive devices with Internet for data storing, manipulation, and analysis purposes captured from sensors which measures physical quantities is termed as Internet of things. Nowadays due to efficient use of technologies for welfare of people, has started increasing. Also, the demand for insights collected from the data required for knowledge extraction has increased. This paper proposes a low-cost cloud data logger which is able to capture temperature and humidity data from various locations. The proposed system uses DHT 11 sensor to capture temperature and humidity data and send it to Bolt Cloud with the help of Arduino. The main objective of this paper is to log weather data to cloud so that we could later analyze it and hopefully find some interesting patterns. Some applications of the data include probability of rainfall, census of wildlife, maintain temperature in data centers.

Keywords IoT · Bolt · Arduino · Data logger · Cloud

1 Introduction

When we talk about technologies of the future, we talk about IoT because we want our surroundings not only smart but also sustainable. Due to increasing pollution, we are facing climate change right now so it is the right time we looked forward to resolve this issue for us and our future generations. For that reason, we are making IoT-enabled environments where humans as well as flora and fauna can prosper. Due to advancement of technologies, we can now develop same topologies using different

G. Prachchhak (✉) · C. Bhatt · J. Thik
CSPIT, CHARUSAT, Changa, India
e-mail: gaurav.prachchhak@gmail.com

C. Bhatt
e-mail: chintanbhatt.ce@charusat.ac.in

J. Thik
e-mail: thik.jaydeep99@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_18

types of hardware and software configurations. For instance, there are many applications of temperature and humidity data like prediction of rain, daily minimum and maximum temperature. Weather forecasting applications have seen many advances after introduction of Internet of things. The main application of temperature and humidity data other than prediction is to monitor and control various appliances. Other applications of the same include controlling and monitoring the air conditioner inside a data center. To make important decisions, not only raw temperature and humidity data but also valuable insights are required. For similar purposes, we are using Bolt IoT and Arduino in this proposed system.

2 Background Details and Related Work Introduction

There are many papers on various topics on IoT and its applications in various sectors like IoT in health care [1, 2], IoT in industries [3, 4], data mining for IoT [5], IoT for smart cities [6, 7] as well as weather-oriented papers like impact on regional climate change on human health [8, 9].

Due to advancements of technologies, it is now possible to monitor a patient in real time by any physician so that immediate response can be given. Nowadays, there are many IoT-enabled devices which can be used to monitor temperature and sleeping pattern of kids. Advanced pacemakers are being created in which all the data about the functioning of heart is transmitted to storage devices as well as doctor. One particular application is of medicine dispenser which uses M2M communication. When a doctor prescribes medicine to a patient, the dispenser fetches the time and day of when the medicine is to be taken from doctor's database and then alerts and dispenses the medicine.

Evolution of technologies enabling IoT started off with radio frequency identification (RFID) which was used in the 1980s for identification and tracking of material in industries. It was largely used in logistics, supply chain management, retailing, and pharmaceutical production. Then came wireless sensor networks (WSNs) which mainly use interconnected wireless sensor and actuators for sensing and monitoring. Its main application includes monitoring of environment, traffic, industries, and health care. Industries cover maximum market share in terms of IoT after healthcare sector.

We now have a big amount of data generated by sensors, and the numbers are supposed to be increased in 5–10 years. So, one question that comes in our mind is what information or hidden patterns can we find from that data. Data mining solves this issue of finding patterns with the help of complex algorithms.

The main goal of IoT is to help people in their day-to-day life by being part of their life. And one such application of IoT is smart cities, in which an architecture is created to connect devices that one city might need for health and well-being of the people to one another via Internet. So that, people will have to spend less time solving issues and more time on their work. To develop a smart city, there are a lot of hurdles that one might face as till date there is a lack of regulations and standards which one

might adapt. But still, people are trying to choose the best and most efficient way out of so many available ways to reach the common goal which is to help people.

Weather affects us all not just standing as a hindrance to our productivity but also increases the chances of certain diseases caused due to certain organisms which thrive on that specific weather. One such research was carried out in the high land of Central Ethiopia where increase in temperature caused increase in malaria a vector-borne disease which is caused by mosquitoes.

3 Proposed Approach

3.1 Architecture

Here when we open our Web dashboard using our credential and open the product, then it requests Bolt to serially read temperature and humidity data that is being generated by Arduino continuously and then we send acknowledgment to Arduino which basically is used for debugging purpose. The whole communication is done securely using API which is unique for every Bolt user and with the name of the device which is again novel. And we store the data in cloud which we can download to perform analysis on it (Fig. 1).

3.2 Bolt

The Bolt is a mini Wi-Fi Internet of things (IoT) module based on Tensilica Xtensa lx106 microcontroller and provides 4 MB flash. Its five GPIO pins make this board suitable for medium IoT target audience. It can only be programmed using OTA. Due to its API, it can be programmed using any language and can be triggered using URI only. It is compatible with any present mobile charger for power supply. It has the ability to communicate to other embedded system also using UART. It also has an android and iOS mobile application to connect it to Internet as well as check the device status.

The technical specification of this module is shown in below table. In this proposed system, the Wi-Fi module of Bolt board is connected to the station having a SSID and password which will access the cloud services of Bolt and push the data that it retrieved using Arduino. This system is cost-effective as it is using commodity hardware. Length and width of this Bolt module are 35 and 35 mm (Fig. 2).

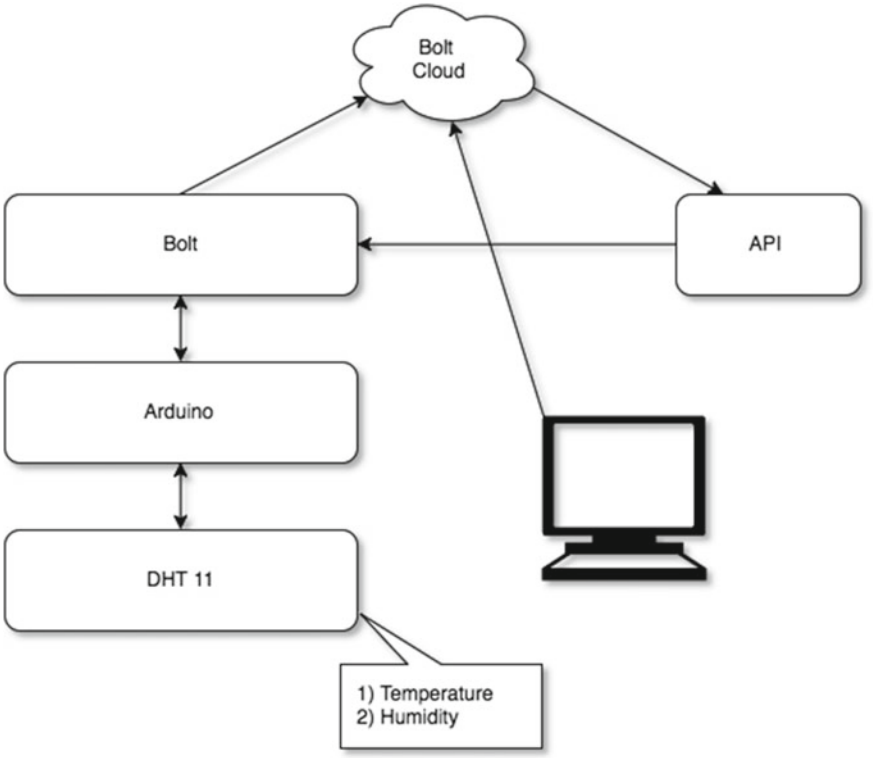
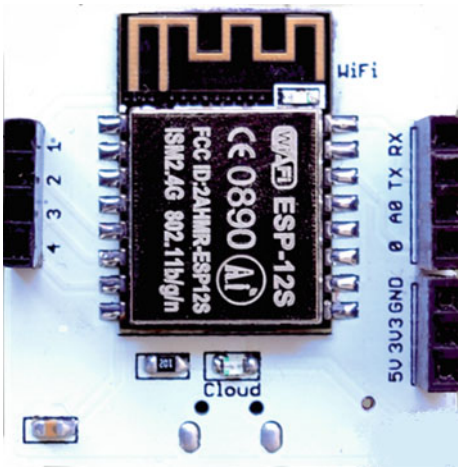


Fig. 1 Basic architecture

Fig. 2 Bolt



Technical Specifications of Bolt IoT:

Microcontroller	Mic32-bit RISC CPU: Tensilica Xtensa lx106
Operating voltage	3.3 V
Digital I/O pins	5
Analog input pins	4 [10 Bit ADC]
Clock speed	80 MHz
Flash	4 MB

3.3 Arduino

Arduino Uno is an ATmega328-based microcontroller board. It has total of 14 input and output pins out of which 6 pins can be used as PWM outputs, 6 pins as analog inputs, a crystal oscillator working on 16 MHz, a standard USB connection, a power jack for input, an ICSP header for expansion purpose, and a reset button. Its basic task is to support the microcontroller. To make it work, connect it to computer via a USB cable or you can power it using AC-to-DC adapter or a battery pack. It defers with other boards in terms of USB-to-serial conversion. Other boards generally use FTDI-based conversions, while Arduino uses ATmega8U2 programmed conversion. It includes a serial monitor which allows us to see the data communicating with other microcontrollers as well as sensors. It uses RX and TX LEDs on the board which will flash when it sends or receives data.

It also contains a bootloader which can hold one program and some data in it. Once you dump your program in Arduino, then there is no need to connect it again with the computer. The maximum length and width of the Uno PCB are 2.7 and 2.1 in., respectively, with the USB connector and power jack extending beyond the former dimension (Fig. 3).

Technical Specifications of Arduino UNO:

Microcontroller	ATmega328
Operating voltage	5 V
Digital I/O pins	14
Analog input pins	6
Clock speed	16 MHz
Flash	32 KB

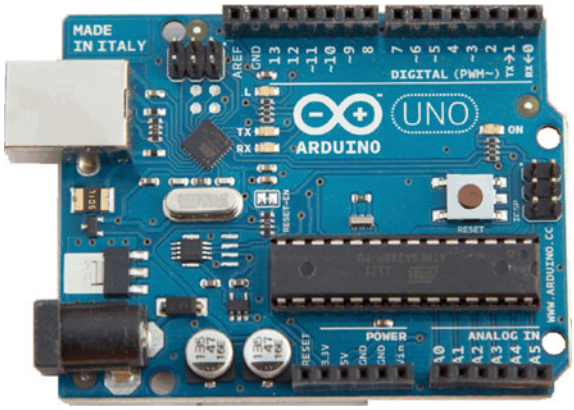


Fig. 3 Arduino

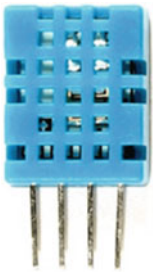


Fig. 4 DHT 11

3.4 DHT 11

It is used for sensing temperature and humidity. It is made up of two parts a capacitive humidity sensor and a thermistor. There is also a very basic chip inside that does some analog-to-digital conversion and spits out a digital signal with the temperature and humidity. The digital signal is fairly easy to read using any microcontroller. This sensor is manufactured by many companies now including local and international companies like Adafruit which by the way also provides its library that we use in Arduino.

Feature of DHT 11:

- Ultra-low cost
- 3–5 V power and I/O
- 2.5 mA max current use during conversion (while requesting data)
- Good for 20–80% humidity readings with 5% accuracy
- Good for 0–50 °C temperature readings ±2 °C accuracy
- No more than 1 Hz sampling rate (once every second)
- Body size 15.5 mm × 12 mm × 5.5 mm
- Four pins with 0.1” spacing (Fig. 4).

3.5 Cloud Services

In reality, it is very hard for people to find some knowledge from the data generated by IoT devices as well as to effectively create, manage, and monitor the products created for those devices. The most effective innovation in IoT will come when it is combined with cloud computing. This combination will enable new monitoring services as well as powerful processing of sensory data streams. Ultimately, the main goal of using cloud is to transform the raw data into insights that drive productive and cost-effective decisions. But, when we integrate IoT with cloud, we are faced with new challenges.

Due to increasing number of IoT devices and cloud platforms, there is an urgent need for new unique architectures that can seamlessly integrate with them. The main concerns that one might face at the time of integration are quality of service (QoS) and quality of experience (QoE) in addition to data security, privacy, and reliability.

For example, data captured from sensors can be uploaded and stored in cloud which can be used for intelligent tasks like smart monitoring and actuation later.

Bolt has its own cloud platform in which it is capable of processing JavaScript and HTML as primary programming languages. Also, people can use various other languages, but then Bolt Cloud will not support as of this date though they are planning to add more support. Right now, it is able to perform tasks like storing of data, displaying it with visualizations and data prediction. It provides Google chart library for visualization, but people can use any library of their choice also.

Bolt Cloud service is based on simple HTTP GET protocol where it sends the request to the stations, user, and sensor information. It has its API; to get our own API key, an account has to be created for which we have to buy Bolt which costs as low as 9$. In the same way, people can use Thinkspeak which is same as Bolt Cloud via its API and use HTTP request to push data from microcontroller to its cloud. In Thinkspeak, you have to create a channel that can be either public or private. Hence, data remains secured in the cloud.

Feature of Bolt Cloud:

- Secure cloud
- Developer console
- Data storage
- More data visualization features
- Device status
- APIs.

Bolt can be configured to use many other cloud services also as generally all the services can be accessed via URI and an API key. Thinkspeak provides various services which include integration of analysis tools like MATLAB, visualizations, apps, and plugins. Another service called WSO2 IoT which is an open platform for devices which are part of IoT is highly scalable to manage the needs of several IoT devices. Axeda is also one such cloud-based service provider which is best in remotely servicing the machines on behalf of client.

4 Implementation

4.1 Hardware Implementation

In this data logger, the temperature and humidity sensor (DHT 11) are connected to Arduino. The makers of DHT have created a library that can be used with Arduino IDE so that it can read the digital output from the sensor accordingly. We have to connect a 10 K resistor between VCC and SIGNAL pins, then connect with 3.3 V power output and any data input pin of Arduino, respectively, and then connect GND of DHT to GND of Arduino. Here after connecting DHT with Arduino, we connect Arduino with Bolt using TX of Arduino to RX of Bolt and vice versa and GND of Bolt to GND of Arduino. A micro-USB cable is used to supply power to Bolt, while a standard USB A-B is used to supply power to Arduino (Figs. 5 and 6).

4.2 Software Realization

Now after all the connections have been done, we have to push our sensor data to cloud. For that, we have to fist write Arduino code that can detect the present temperature and humidity and for debugging purpose we print output of our sensor to serial output which we can use for debugging purpose. After successful results of that module, we have to now write JavaScript which can read the serial data from Arduino, then store it after every 3 s, and plot the graph of that reading simultaneously so that we can see results are visualized better. After writing JavaScript, we now have to create a new product in Bolt's Web interface and setup accordingly and then we

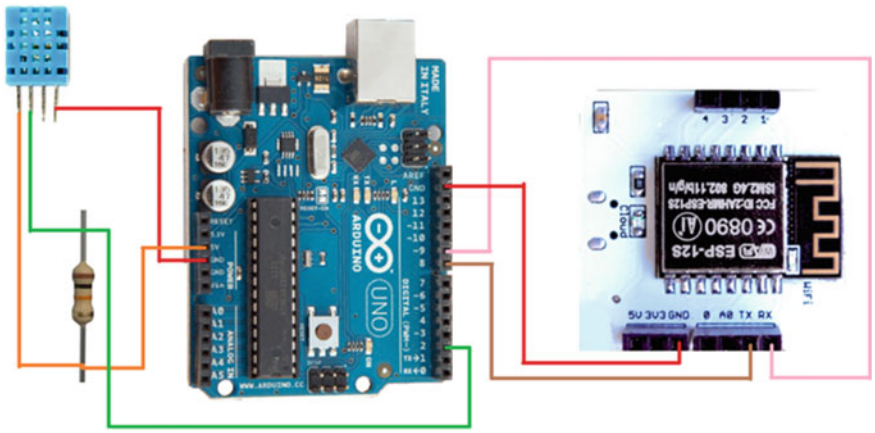


Fig. 5 Circuit connection

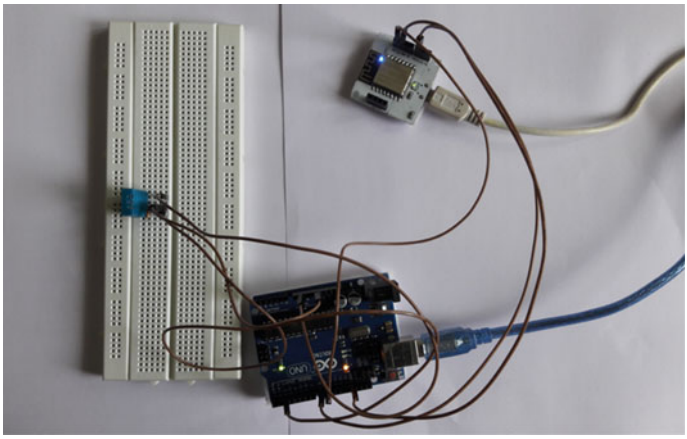


Fig. 6 Implementation

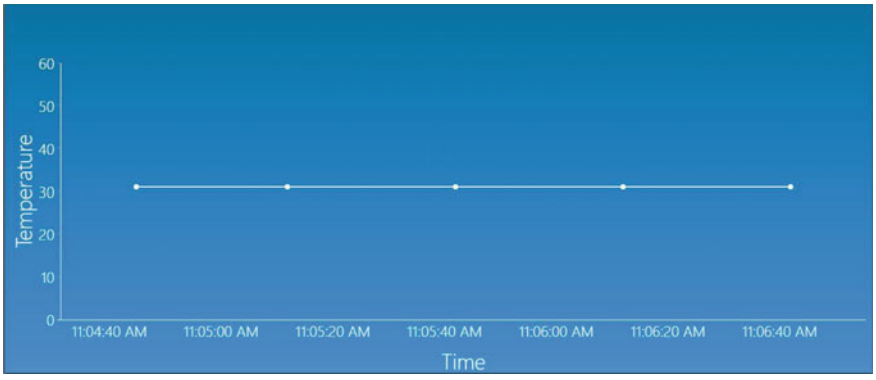


Fig. 7 Real-time temperature reading

have to upload the JavaScript that we have written and assigned that to our product. And finally, deploy the product on Internet.

5 Testing and Results

After configuring and deploying our product online, we can now power on both Arduino and Bolt and then open the Web application where we can see the temperature and humidity data plotted on graph in real time. We also created a csv module which stores all the data and binds them into a single csv object which gets pushed in a csv file when we click the download button from the application. After collecting data from various locations, we can analyze that data (Figs. 7 and 8).

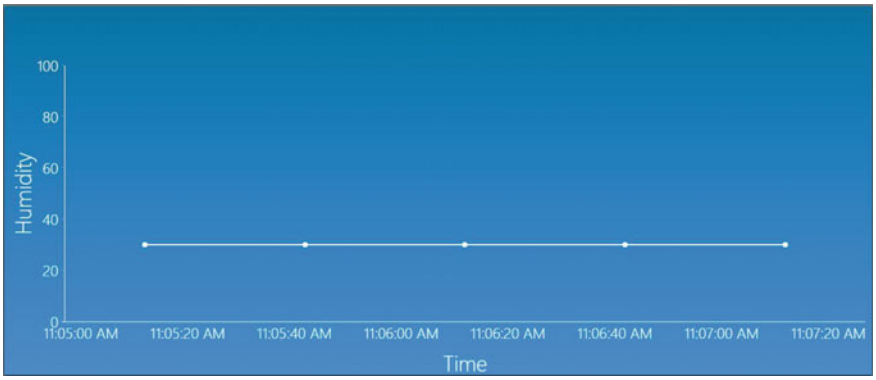


Fig. 8 Real-time humidity reading

6 Conclusions

As the number of IoT-enabled devices is increasing day by day, we can now use the data collected from them to make some important devices which result in efficient use of resources and overall sustainable development. The main aim of this paper is to implement an architecture which is capable of logging data in cloud which can be used for analysis purpose afterward. More research and work need to be carried out in field of IoT so that we can come to a point where we can have standardized architectures. Thus, using modern technologies like IoT we can create a world which is smart and nature-friendly.

References

1. A.S. Yeole, D.R. Kalbande, Use of internet of things (IoT) in healthcare: a survey (2016), http://dl.acm.org/citation.cfm?id=2909079
2. C. Bhatt, N. Dey, A.S. Ashour, Internet of Things and Big Data Technologies for Next Generation Healthcare, vol. 23 (Springer, n.d.). https://doi.org/10.1007/978-3-319-49736-5
3. T. Shah, C.M. Bhatt, The internet of things: technologies, communications and computing. CSI Commun. (2014), http://csi-india.org/communications/CSIC-April-2014.pdf#page=7
4. L.D. Xu, W. He, S. Li, Internet of things in industries: a survey, 16 Jan 2014, http://ieeexplore.ieee.org/abstract/document/6714496/, https://doi.org/10.1109/tii.2014.2300753
5. C. Tsai, C. Lai, M. Chiang, L.T. Yang, Data mining for internet of things: a survey. IEEE Commun. Surv. Tutor. **16**(1), 77–97 (2014). https://doi.org/10.1109/SURV.2013.103013.00206
6. M. Bhayani, M. Patel, C. Bhatt, Internet of Things (IoT): In a Way of Smart World, vol. 438 (Springer, 2016), pp. 343–350. https://doi.org/10.1007/978-981-10-0767-5_37
7. A. Zanella, N. Bui, A. Castellani, L. Vangelista, M. Zorzi, Internet of things for smart cities. IEEE Internet Things J. **1**(1), 22–32 (2014). https://doi.org/10.1109/JIOT.2014.2306328
8. J.A. Patz, D. Campbell-Lendrum, T. Holloway, J.A. Foley, Impact of regional climate change on human health. Nature **438**, 310–317 (2015). https://doi.org/10.1038/nature04188
9. R.K. Kodali, A. Sahu, An IoT based weather information prototype using WeMos. Contemp. Comput. Inf. (IC3I), 612–616 (2017). https://doi.org/10.1109/ic3i.2016.7918036

Efficient Real-Time Decision Making Using Streaming Data Analytics in IoT Environment

S. Valliappan, P. Bagavathi Sivakumar and V. Ananthanarayanan

Abstract Setting up and managing Internet of things environment possess challenges like capturing, storing, mining, and processing of massive high-speed data streams generated by various sensors and computing devices. In this paper, the focus is on decision support systems incorporating cloud infrastructure and streaming data analytics to provide efficient decision making at the point of care in smarter environments. It is best explained with the help of performing streaming data analytics over real-time smart environment setup to determine the factors for efficient water usage in the near real time.

Keywords Data streams · Streaming data analytics · Internet of things (IoT) Smart water

1 Introduction

In the modern times, the Internet of things (IoT) has tremendously increased the number of intelligent devices connected to the Internet. These intelligent devices help us capture more data from various places, ensuring more ways of increasing efficiency and improving safety and security. Internet of things is the ecosystem of embedded technology and connected physical devices to provide smart environments. In general, smart environments are responsible for massive high-speed data streams. Providing better storage and processing of these high-speed data streams to make efficient decisions are an active research area.

S. Valliappan (✉) · P. Bagavathi Sivakumar · V. Ananthanarayanan
Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore, India
e-mail: valliappan.ss@gmail.com; cb.en.p2cse16019@cb.students.amrita.edu

P. Bagavathi Sivakumar
e-mail: pbsk@cb.amrita.edu

V. Ananthanarayanan
e-mail: v_ananthanarayanan@cb.amrita.edu

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_19

Batch data processing helps in processing the huge volume of data collected over a period of time efficiently. In contrast, real-time data processing includes streaming data and it is to be processed in a minimal time period. Streaming data analytics plays a vital role in efficient decision making in the real-time environment. These features help to achieve efficient decisions in the near real time.

The growth of cloud computing is tremendous in the recent years. The fast-paced growth of IoT leads to the production of massive data, which needs to be stored, processed, and accessed. Cloud computing is an information technology paradigm that enables higher-level services such as big data storage and analytics over the Internet. Combining these two trending technologies provides the real innovation and increases the productivity of the business.

This paper focuses on using the cloud infrastructure that incorporates the use of batch data processing, real-time data processing, and event data processing. The main contribution of this paper is to provide efficient decisions in the near real time from the captured high-speed data streams which are generated from the IoT environment.

The rest of the paper is organized as follows. Section 2 describes the previous works carried out in this area. The design plan, implementation, and analysis of the proposed system are discussed in Sect. 3. Lastly, conclusions and future work are discussed in Sect. 4.

2 Related Work

In the recent years, we are in need of analyzing high-speed data streams and detecting the complex data patterns in the near real time led to an aspect of complex event processing in conjunction with machine learning for prediction. One such work proposed is adaptive moving window regression for dynamic IoT data [1]. The intelligent devices in the IoT environment are the source of generating high-speed data streams, and data obtained will be in a very large amount ranging from terabytes to petabytes. The challenges in mining data from real-time massive data include memory lag, faster computing environment. In the recent times, in-stream data analysis plays a vital role. A cloud-based data stream processing platform is discussed to address the challenges in handling dynamic IoT data streams [2]. Stream data mining issues, challenges, algorithms, and performance of some of the benchmark datasets are compared and analyzed based on the streaming data mining parameters which are discussed in [3]. The key challenges in mining data stream in real time includes mining large-scale data streams with lesser memory requirements, reducing the computation cost, performing classification and clustering. Standard streaming classification and clustering algorithms with the period of evolution are also portrayed, and few are taken to carry out the experiments to assess its performance. The parameters used for performance evaluation are classification accuracy, kappa statistics, temporal kappa statistics, and elapsed time.

The connected and smart cities concept is discussed in [4, 5]. The two main IoT technologies, cyber-physical cloud computing and mobile crowd sensing, are used

to maintain the networking of the connected sensors. The goal of the work is to preserve the history, survive in the present, and prepare for the future. Furthermore, it presents the IoT and data analytics results on the TreSight dataset for the smart heritage of the culture in Trento city. The amount of data that is being generated is huge and progressive in nature. Therefore, IoT needs to be integrated with the cloud infrastructure to have sufficient storage capacity and optimal resource utilization. It is named as a cloud of things and is used to build the smart applications [6]. However, it faces issues such as protocol support, energy efficiency, resource allocation, identity management, IPv6 deployment, service discovery, quality of service provisioning, the location of data storage, security, privacy, and unnecessary communication of data.

The integration of the IoT with the cloud platform is useful in various intelligent applications and efficient sharing of the knowledge across the network. The architecture of the manufacturing service system that integrates the IoT and cloud computing is ascertained to provide an on-demand service, resource utilization, and free sharing of the information [7].

3 Proposed System

3.1 IoT Environment Setup Design

Setting up a real-time IOT environment has its own challenges while collecting the intended data. The plan is designed with a point of care to collect the streaming data which determines the factors of water stored in the overflow tank and sump. Figure 1 provides the design plan for the IoT environment. This is toward designing a smart water environment.

3.2 IoT Environment Setup Challenges

The major challenges faced during the implementation of design plan are described below.

Table 1 provides the key challenges and its description. Next section gives a detailed explanation of how each challenge is addressed.

3.2.1 Challenge 1: Sensor Placement

The challenges faced during sensor placement are using a minimum number of sensors in stable position inside water to sense the smart water data. It is addressed by measuring the height, width, and capacity of the tank. Based on the measurements,

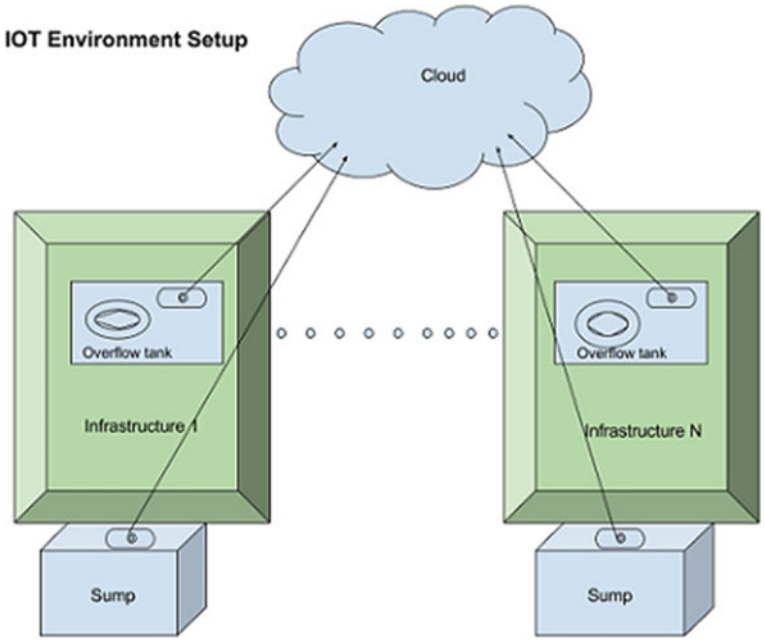


Fig. 1 Environment setup design plan

Table 1 Key challenges during environment setup

S. no.	Major challenges	Description
1.	Sensor placement	Placing a minimum number of sensors to monitor the water level in the tank
2.	Node placement	Deciding on the node to send sensor data to the cloud
3.	Cloud platform	Deciding on the cloud platform best suits the goals

every float sensors are placed equidistant from the other in an iron rod as shown in Fig. 2. The iron rod used here matches the height of the tank, and the tight coupling between each sensor and rod helps to hold the sensors in a stable position.

3.2.2 Challenge 2: Node Placement

Deciding on the node to send the smart water data to the cloud as intended is itself a big challenge. Issues faced on each node while carrying out the experimentation and how it is resolved are discussed below.

1. Arduino with radio frequency (RF) is used to send the sensor data to the cloud. In field testing, the node fails to receive data due to distance problem.

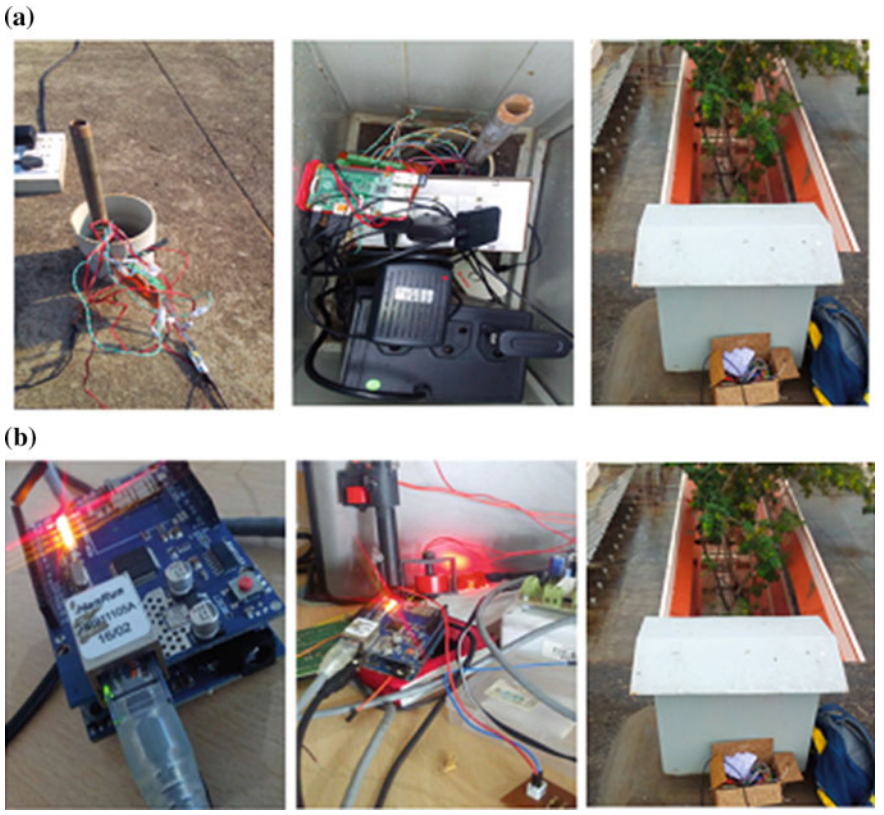


Fig. 2 **a** Raspberry Pi implementation in the field and **b** Arduino Ethernet shield setup

2. The ESP8266, a Wi-Fi chip holding TCP/IP stack and microcontroller unit, is attached to it. When ESP8266 is used in the field, it fails to connect to the enterprise network because of PEAP protection. So it requires a separate access point to send the data to the cloud.
3. Raspberry Pi is used to avoid the necessity of separate access point. It helps to connect to the enterprise network. Ubuntu MATE operating system is used in the Raspberry Pi to connect Amrita campus Wi-Fi. It is used to send the smart water data to the cloud. In the field when the raspberry pi is placed above the overhead tank, Wi-Fi connectivity started toggling due to weak signal strength. The problem here is Wi-Fi signal is not stable in the field. Figure 2a gives the raspberry pi field implementation to send the smart water data to the cloud.
4. Arduino + Raspberry Pi + near-field radio frequency (NRF)—To overcome the Wi-Fi toggling problem in addition to the Raspberry Pi, Arduino and NRF are added. In the field, near-field radio frequency (NRF) waves start scattering in a water medium. So node fails to receive the data.

5. Tx Aurdino—RS485 Communication—Rx Aurdino + Ethernet Shield—This combination is used to address the challenge. In this setup, transmitter Arduino sends the smart water data to receiver Arduino via an RS485 communication channel. The receiver Arduino sends the smart water data to the cloud via MQTT protocol [8]. This node implementation resolves the issues faced previously on various nodes. Arduino Ethernet shield is successfully implemented in the field as shown in Fig. 2b. This node is used to send the smart water data to the cloud.

3.2.3 Challenge 3: Cloud Platform

The aim is to select the cloud infrastructure which is capable of processing massive high-speed data received from IoT environment and incorporates the use of batch data processing, real-time data processing, complex event processing, and many more scalable applications programming interface to deal with various scenarios in IoT environment. Microsoft Azure cloud platform offers a wide range of capabilities. It includes collecting, storing, and analyzing data streams in motion, which is generated from various devices. It supports visualizing both real-time and historical data and facilitates integrating with back-office systems. Azure Stream Analytics (ASA) provides in-motion data analysis, which is used by the IoT suite to process incoming telemetry, aggregation, and in the detection of events. Three ASA jobs which are used to filter out the telemetry stream from devices are device info job, telemetry job, and rule's job [9]. Hence, Microsoft Azure cloud platform best suits to achieve the intended goals.

3.3 IoT Environment System Implementation

The design plan was successfully implemented in the hostels at Amrita School of Engineering, Coimbatore. Implementation challenges include sensor placement in the overflow tank and sump, node placement, data acquisition, and data feed into the cloud. Some of the major implementation challenges and how to address those challenges are discussed in the previous section. The system implementation of the design plan is portrayed in Fig. 3. Sensors are placed inside the overflow tank and sump to sense the factors of smart water. Sensor data are read with the help of node. Arduino with Ethernet shield serves as a node to collect the sensor data, thereby starting the data acquisition. Ethernet shield helps in transmitting the data to the Microsoft Azure cloud. IoT suite solution is created in the Azure cloud in order to channelize the transmitted streaming data from the environment.

The channelized streaming data went through the Azure Stream Analytics to provide Microsoft Power BI data visualization and real-time alerts.

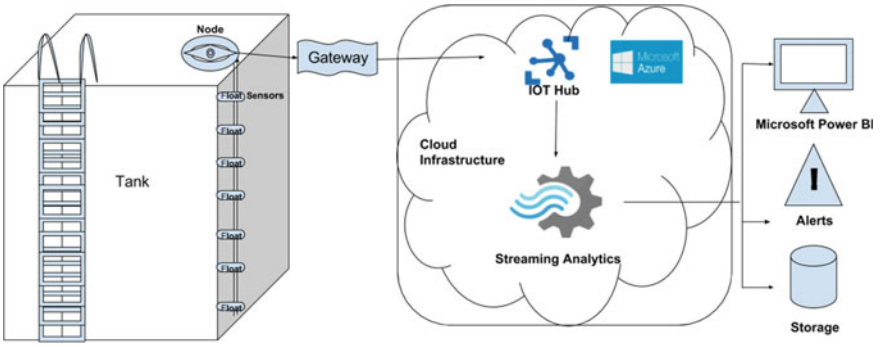


Fig. 3 System implementation

3.4 Setting up Solution in IOT Suite

The Azure IoT Suite is a set of Azure services that enable to capture and analyze the data generated. Azure IoT Hub is a fully managed service that enables reliable and secure bidirectional communication between millions of IoT devices and a solution backend [10]. Azure IoT Hub is the core of IoT suite. SmartW is the newly created remote monitoring preconfigured solution in Azure IoT suite. The solution is launched, and the devices are added for remote monitoring. It receives the data from the devices which are placed in the tanks and sump.

3.5 Batch Processing on Dataset

Data obtained from the created IoT environment for a period of time are taken as the dataset. Azure Stream Analytics job is created to perform batch processing. The dataset is given as input source to the created job. The query interface is used to run complex queries that retrieve the insights from the dataset. Figure 4a gives the results obtained from the query framed with an intention to retrieve the overhead tank water data exceeding the threshold with other insights such as indicating which event is the first within 10 min intervals per location id, lag for water data within a period of 1 h.

3.6 Stream Processing of Real-Time Data

Data can be ingested into the Azure cloud from the devices through Azure IoT Hub. Azure Stream Analytics (ASA) job is created to perform stream processing on real-time data. Here, SmartW IoT Hub is given as input source to the created job.

(a)

output

Generated the Following:

- output with 5501 rows.

Download results

DATE	TIME	OVERHEADTANKLEVEL	MINOVERHEADTANKLEVEL	MAXOVERHEADTANKLEVEL	FIRST	LAG
"2017-07-26"	"10:15:00"	"80"	0	100	1	null
"2017-07-27"	"10:15:10"	"90"	0	100	0	"80"
"2017-07-27"	"10:15:10"	"78"	0	100	0	"90"

(b)

www.paiot.net/asesm/

ASE Smart Water IoT Sensor Data

(ASE-IoT Lab)

Smart Water Sensor readings

Date	Time	Overhead Tank	Sump	Location
2017-11-10	16:41:51	50	100	GowthamaBhavanam
2017-11-10	16:41:49	50	100	GowthamaBhavanam
2017-11-10	16:41:47	50	100	GowthamaBhavanam
2017-11-10	16:41:46	50	100	GowthamaBhavanam
2017-11-10	16:41:43	50	100	GowthamaBhavanam

Fig. 4 **a** Batch processing result and **b** live data feed

Further, stream processing logic is applied to retrieve the insights from the real-time data with the intent to make effective proactive decisions. The Web application is designed in the cloud to show the live feed of smart water data based on the search string. Figure 4b gives the live feed of overhead tank and sump water data in a hostel. From the live feed data, stream processing logic is applied to retrieve the insights such as hourly, daily, weekly, monthly, and yearly water needs in each location.

4 Conclusion and Future Work

This paper discusses setting up real-time IoT environment and their integration with cloud computing for useful service provisioning for a smart water application. The useful end-user decision-making services discussed in the paper are batch processing, streaming data analytics, and live data feed. The key challenges of IoT environment setup are also discussed as well. Future work includes making the cloud solution to deal with extended IoT scenarios to resolve the decision-making challenges of streaming data.

Acknowledgements The experimentation was carried out with the support from the IOT laboratory, Amrita School of Engineering, Coimbatore, and as part of an internally funded research project —Data Collection Framework for SMART Water Management and Analytics (AMRITA/IFRP-07/2016–17).

References

1. A. Akbar et al., Predictive analytics for complex IoT data streams, in *IEEE Internet of Things* (2017)
2. D. Kalashnikov et al., Cerrera: in-stream data analytics cloud platform, in *2015 Third International Conference on Digital Information, Networking, and Wireless Communications (DINWC)* (IEEE, 2015)
3. B.R. Prasad, S. Agarwal, Stream data mining: platforms, algorithms, performance evaluators and research trends. Int. J. Datab. Theory Appl. **9**(9), 201–218 (2016)
4. Y. Sun et al., Internet of things and big data analytics for smart and connected communities. IEEE Access **4**, 766–773 (2016)
5. P. Vijai, P. Bagavathi Sivakumar, Design of IoT systems and analytics in the context of smart city initiatives in India. Proc. Comput. Sci. **92**, 583–588 (2016)
6. M. Aazam et al., Cloud of things: integrating internet of things and cloud computing and the issues involved, in *2014 11th International Bhurban Conference on Applied Sciences and Technology (IBCAST)* (IEEE, 2014)
7. F. Tao et al., CCIoT-CMfg: cloud computing and internet of things-based cloud manufacturing service system. IEEE Trans. Industr. Inf. **10**(2), 1435–1442 (2014)
8. S. Sharad, P. Bagavathi Sivakumar, V. Anantha Narayanan, A novel IoT-based energy management system for large scale data centers, in *Proceedings of the 2015 ACM Sixth International Conference on Future Energy Systems* (ACM, New York, July 2015), pp. 313–318
9. https://docs.microsoft.com/en-us/azure/stream-analytics/stream-analytics-introduction
10. https://docs.microsoft.com/en-us/azure/iot-hub/

IoT-Based Smart Doorbell Using Raspberry Pi

Abhishek Jain, Surendra Lalwani, Suyash Jain and Varun Karandikar

Abstract Security in home is now being directed from traditional methodologies to automation with the help of Internet. The Internet of things is the network of interconnecting devices (may be physical devices, vehicles, home appliances, etc.) embedded with electronics, sensors, etc., to exchange data. Nowadays, home systems are equipped with computing and information technology which provides them with smartness and intellect. Since doors are the gateway to our homes, therefore it is necessary to make them more secure. Currently available mechanism of providing secure access to doors includes bare-metal locks and some smart locking systems. The smart locking system's performance can be evaluated on the basis of identification accuracy, intrusiveness, and cost. In this paper, we introduce the idea to provide secure access to home. It will be achieved through smart doorbell which is a cost-effective alternative to currently its counterparts. Our system connects WiFi-enabled Android devices with firebase server using Raspberry Pi and enables user to answer the door when the doorbell is pressed. It learns to identify new user by using face recognition as a unique identity to authenticate the individual.

Keywords IoT · Secure home · Door locking · Face recognition · Google firebase

1 Introduction

Security has been a major concern from ages. Earlier kingdoms were secured by building castles on hilltops. In the recent past, people used to secure their places with bare-metal locks but such systems are easily compromised by the attacks of malicious attacker/intruders thus increasing the risk of burglary. But with the advent of technology, the vulnerabilities are exposed easily. Now, the requirement to build robust system has become prominent.

A. Jain (✉) · S. Lalwani · S. Jain · V. Karandikar
Department of Computer Science and Engineering, Medi-Caps University, A.B. Road, Pigdamber, Rau, Indore 453331, Madhya Pradesh, India
e-mail: abhi63269@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_20

Smart homes have been gaining popularity nowadays [1]. These systems aim to improve the user interaction with the home appliances in a smart and efficient way. Following this trend, we have smart locks. Smart lock communicates with the WiFi-enabled mobile devices to provide secure access. These locks not only include mechanical parts but also include electronic components having computer processing [5]. To accelerate the development of smart locking mechanisms, many companies such as Nest Hello (Google) [2], August [3], Danalock [4] have proposed such smart lock devices. There are several types of smart locking devices that may be classified on the basis of the technologies used such as Bluetooth Low Energy (BLE), Radio Frequency Identification (RFID), Near-Field Communication (NFC), but they are prone to one or more threats [6]. Therefore, there is a need of a system which can overcome the vulnerabilities of the previous systems.

Recently, smart lock system using the Internet of things (IoT) has been widely featured. We have proposed the idea of smart door lock which employs the state-of-the-art IoT technology to enhance the security by providing authentication in two steps. This two-factor authentication involves granting/revoking access and identifying the visitor who presses the doorbell.

Our proposal is based on the state-of-the-art IoT technology and uses Android as the base platform for the development. The idea of our system is to provide secure access to home. It will be achieved through smart doorbell; once the visitor pressed the doorbell, it will publish an alert and a trigger will be sent to all the system users using Google Firebase Platform service by instant notification on Android phones, so the users will know someone is knocking at the door no matter where the person is. Visitors no longer need to call the person and simply let the smart doorbell to do the job; deaf people also benefit using it so they can alert from vibration of their phone. After receiving the image of the person who pressed the doorbell on family members' phone, one can designate the user as an authorized or unauthorized depending upon the perspective and the same along with the image will be stored in the database which will be used in the future. If the person was categorized as authorized and whenever the same person arrives again, then the same procedure will not be repeated and just a message will come along with their name, and hence, in this way security is achieved. One can also interact with the owner via our product and can leave a message in case owner is unavailable. If in case intruder forcefully breaks in the system, alert would be generated which sounds a buzzer and notifies all system users, thus strengthening security.

2 Related Work

Security in homes is one of the major concerns these days. Many firms have started to deliver smart locking mechanisms that are based on BLE, NFC, WiFi enabled, sensor based, biometric checks [6]. The devices that are taken into account are Bluetooth enabled and also are limited to a certain distance. They must be paired with the mobile handset in order to enable the smart lock features. Smart locks that use

Table 1 This is a table describing different technologies used for providing security [6]

Technologies	Applications/usage	Drawbacks	Examples
Bluetooth	Automatically unlocks the door when an authorized device is in BLE range	Prone to unintentional auto-unlocking	August, Danalock, Kevo, Okidokeys
RFID	It can detect visitor from several feet without being in direct line of sight	It is exposed to eavesdropping attack	MFRC522 door lock
NFC	Better version of RFID, data transfer is fast	Hacking prone system, expensive	Samsung Pay, Apple Pay
Sensor based	Faster detection	If temperature difference is minute, inaccuracy arises in the system	PIR-based door lock
Biometric	Unique way of authentication, difficult to override	Expensive, false positives are common	Yale Digital Door Lock, Ultraloq, ADEL, Nucli SmartLock

Bluetooth connectivity are August [3], Danalock [4], Kevo [7], and OkiDokeys [8], while Lockitron [9] uses WiFi technology to connect the mobile device and maintain the consistency across the lock and device which is an advantage over other locks.

Bluetooth Devices. Bluetooth is a wireless technology that aims to connect close devices over short distances from a fixed device or mobile devices and building personal area networks. The range of Bluetooth is 10 m [10].

RFID-Based Devices. RFID is an automatic identification technology used to record the presence of an object using radio signals. A RFID tag is a small object which stores information and can be read from several feet away [11].

Near-Field Communication (NFC)-Based Devices. It is a method of wireless data transfer to enable communication when two devices are in close contact with each other [12]. NFC allows the user to transfer data up to 10 cm [6].

Sensor-Based Devices. Sensor detects any changes in environment, and whenever it detects something, its output pin becomes high. Human body emits infrared because of body heat; so when an object passes through sensor range, they get caught by the sensors, and hence, the main purpose of security is achieved [13].

Biometric Devices. Biometric access is one of the electronic locking systems which is operated by using unique characteristic of individual. This authentication process is based on unique physical traits, such as a fingerprint, a palm print, face recognition, iris scanner [6] (Table 1).

Face Recognition. Face detection is to identify an object as a human face and locate this face in input image regardless of color, size, position, and features. Face recognition system is a biometric solution that measures unique characteristics about one's face and will identify the human if the image of the same person is stored in the database by extracting useful information from the image. Face recognition

Table 2 This is a table describing face recognition algorithms and their accuracy rates [15]

Algorithm	Accuracy rate (in %)
Principal component analysis	93.7
Linear decrement analysis	95.3
Support vector machine	95.3
Independent component analysis	80.0
Singular value decomposition	97.5

system uses the spatial relationship among the facial features such as eyes, nose, lips, chin, and global appearance of the face [14]. We compared few of the face recognition algorithms based on their accuracy rates. The following results were observed (Table 2).

SVD has the highest accuracy rate, while ICA has the lowest accuracy rate.

3 Proposed Idea

By taking into consideration previously existing systems, we try to overcome their limitations by proposing a unique solution. Our proposed system tries to achieve this using IoT. For this, we use Raspberry Pi as the base device that incorporates several components of our system.

Raspberry Pi is a general purpose computing device that can be interfaced with lot of hardware and electric components [16]. The electronic circuit constituting the doorbell push button is attached with the Pi using a breadboard. It interfaces with a camera module which will be used for capturing images and has sensors attached like motion detectors to detect intruder activity. If such thing happens, system will trigger an alert which ultimately sounds the buzzer. It also houses the code for interacting with the firebase storage and communicates with the firebase servers using secure WiFi connection over SSL thus protecting against relay attacks.

The firebase servers provide excellent availability and provide easy storage and retrieval of data. Firebase also provides real-time database service which stores data in key–value format. It offers authentication in simple and secure way and also stores and shares user-generated content like images, audio, and video [17]. The users will interact with the system using Android application which will be accessible via Google Play Store, and updates can be pushed without much user intervention.

Android currently owns a large share of global mobile market thus being our preferred choice [18]. Android application presents user with notification upon doorbell press. The application also allows users to grant/revoke access, view records, and save details of people. Even user can get access to application using Google credentials. Users can also convey specified message using Google Assistant API. The smartness of the doorbell relies on recognizing person at the door and taking appro-

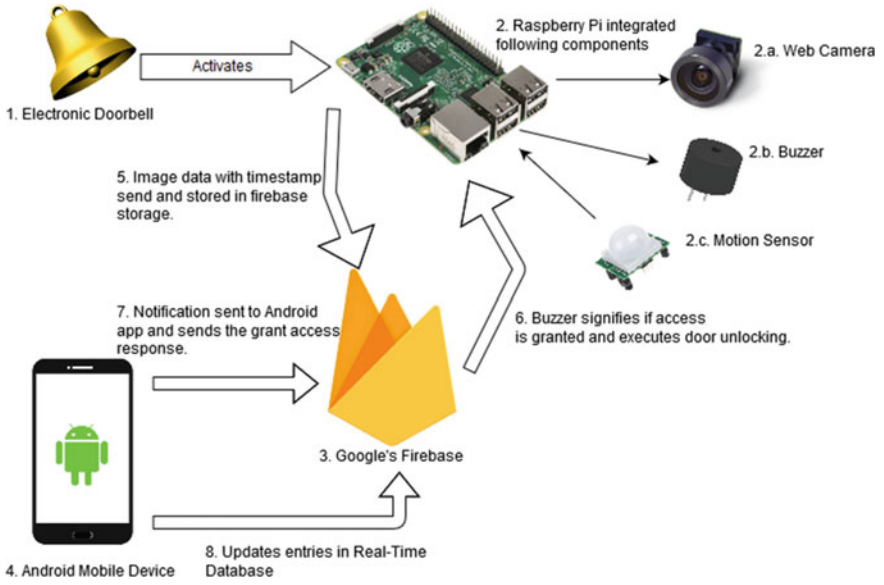


Fig. 1 Flowchart of the proposed smart locking system

priate action by itself. Implementing face recognition using TensorFlow can provide such functionality.

TensorFlow is an open-source library for expressing machine learning algorithms and implementation for using such algorithms. It is a second-generation system for the implementation and deployment of large-scale machine learning models. It takes computations described using a dataflow-like model and maps them onto a wide variety of heterogeneous systems, ranging from mobile devices such as phones and tablets up to large-scale distributed systems of hundreds of machines and thousands of computational devices such as GPU cards [19]. The system is flexible and can be used to express a wide variety of algorithms.

In 2015, researchers from Google released a paper, FaceNet [20], which uses a convolutional neural network relying on the image pixels as the features, rather than extracting them manually. It achieved a new record accuracy of 99.63% on the LFW dataset [21]. This will allow face of a person, an entity that human beings use naturally to identify any individual, to be used as a biometric. The face recognition requires training a neural network that may take a long time if the image set increases (Fig. 1).

4 Discussion

The idea of our proposed system differentiates from existing smart locking mechanism in the following aspects. Firstly, it is WiFi-based locking hence not prone to auto-unlocking unlike BLE-enabled devices. Secondly, integrating Firebase and TensorFlow let us achieved security (in terms of data encryption and users personal data), reliability, and performance in our system in a more robust way. Having several effective features, there are still some security issues to deal with. Since using WiFi, it can lead to network compromise. Face recognition accuracy decreases when disguised faces are treated or noise gets increased in image due to environmental factors. Overall system is robust and thus provides authentication and identification in an accurate and efficient fashion.

5 Conclusion

In this paper, we studied the security of the smart homes using IoT. We have proposed a system that could overcome the limitations of the existing smart door lock systems. These systems were limited to the applications of the technology used and posed many threats and vulnerabilities. With our proposed system, we eliminated these threats by making a more robust infrastructure.

Using Google's Firebase and TensorFlow Library, we integrated our system in an efficient way. Our findings had the importance and significance in terms of their implications and cost-effectiveness. We also used TensorFlow library to implement facial recognition so that the authenticated members could be recognized easily.

In future work, we plan to implement two-way smart audio communication through Google Voice Assistant API. We also plan to incorporate continuous video surveillance. We propose to conduct survey of the system regarding the usability and reliability and improve the performance based on the user feedback.

References

1. Where the smart is, https://www.economist.com/news/business/21700380-connected-homes-will-take-longer-materialise-expected-where-smart
2. Nest, https://nest.com
3. August, https://august.com
4. Danalock, https://danalock.com
5. G. Ho, D. Leung, P. Mishra, A. Hosseini, D. Song, D. Wagner, smart locks: lessons for securing commodity internet of things devices (University of California, Berkeley, USA) (2016)
6. D.B.M. Yin, M.I. Kamal, N.S. Azmanuddin, S.H.S. Ali, A.T. Othman, R.Z. Wan-Chik, Electronic door access control using MyAccess two-factor authentication scheme featuring near-field communication and eigenface-based face recognition using principal component analysis (Malaysian Institute of Information Technology, Kuala Lumpur, 2016)

7. Kevo, http://www.kwikset.com/kevo/default.aspx
8. Okidokeys, https://www.okidokeys.com/
9. Lockitron, https://lockitron.com/
10. E. Ferro, F. Potorti, Bluetooth and Wi-Fi wireless protocols: a survey and a comparison. Wirel. Commun. IEEE
11. EPC-RFID INFO, https://www.epc-rfid.info/rfid
12. NFC, http://searchmobilecomputing.techtarget.com/definition/Near-Field-Communication
13. PIR Sensor, https://circuitdigest.com/microcontroller-projects/automatic-door-opener-project-using-arduino
14. A. Juels, Minimalist cryptography for low-cost RFID tags, in *Security and Cryptography for Networks*, pp. 149–164
15. R. Ravat, N. Dhanda. Lucknow, India. Performance comparison of face recognition algorithm based on accuracy rate (2015)
16. Raspberry Pi, https://www.raspberrypi.org/
17. Firebase, https://firebase.google.com/
18. Global mobile OS market share, https://www.statista.com/statistics/266136/global-market-share-held-by-smartphone-operating-systems/
19. M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G.S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Man´e, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Vi´egas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, X. Zheng. TensorFlow: large-scale machine learning on heterogeneous systems (2015)
20. FaceNet: A Unified Embedding for Face Recognition and Clustering, https://arxiv.org/abs/1503.03832
21. Building a Facial Recognition Pipeline with Deep Learning in Tensorflow, https://hackernoon.com/building-a-facial-recognition-pipeline-with-deep-learning-in-tensorflow-66e7645015b8

Detection of Black Hole Attack in GPCR VANET on Road Network

Naziya Hussain, Priti Maheshwary, Piyush Kumar Shukla and Anoop Singh

Abstract VANET nodes can be directly communicated with each other if this VANET node comes in transmission range; the sender nodes are forwarding nodes which send packets to the receivers. In this research paper, concentrate on dismembering and enhancing the most commonly used VANET protocol Greedy Perimeter Coordinator Routing (GPCR). Our concentration particularly is on enhancing the black hole attacks security in the GPCR. The solutions of black hole attack are verified with the help of implementation and simulation using network simulator (NS-2.35). Our investigation demonstrates the comparison between GPCR and black hole-GPCR in the network. This comparison is carried between the simulation time and a number of nodes based on the QoS parameter performance of packet delivery ratio (PDR), average end-to-end delay, average throughput, and energy consumption. The packet delivery ratio performances are increased in GPCR with respect to Black Hole GPCR, its increase the performance of the network. The simulation results show that GPCR is better than black hole-GPCR.

Keywords GPCR · Black hole attack · VANET · Security

N. Hussain (✉)
School of Computers, IPS Academy, Indore, India
e-mail: naziyahussain@gmail.com

P. Maheshwary
Aisect University, Bhopal, India
e-mail: pritimaheshwary@gmail.com

P. K. Shukla
Department of CSE, UIT, Bhopal, India
e-mail: pphdwss@gmail.com

A. Singh
CMCC Mhow, Indore, India
e-mail: cmccmhow@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_21

1 Introduction

Vehicular ad hoc network (VANET) is a type of portable specially appointed system, to give interchanges between adjacent vehicles and among vehicles and closest fixed equipment, normally depicted as roadside equipment.

The vehicular ad hoc network is used to give security and comfort to a traveler. The little electronic devices will give ad hoc organize availability to the travelers inside the vehicles. By these devices working, this system does not require convoluted association and server correspondence. Every vehicle furnished with vehicular ad hoc organize devices will be a hub in the ad hoc arrange and can get and transfer others messages through the wireless system.

In vehicular ad hoc arrange, we are utilizing ad hoc specially appointed systems administration advances, for example, Wi-Fi IEEE 802.11 b/g, WiMAX IEEE 802.16, Bluetooth, IRA, and ZigBee for simple, exact, powerful, and straightforward correspondence between vehicles on unique versatility.

Vehicular ad hoc network incorporates networks on wheels, vehicle to vehicle, and safety communication consortium [1]. Although all together for these advances to make them to the arrangement organize, potential security and protection issues [2, 3] must be addressed. Since protection is a twofold edge sword because of its contention with other security necessities, a restrictive and tradeoff arrangement should be at the place keeping in mind the end goal to adjust the impact of contention. For instance, if there should be an occurrence of Sybil attack and security preservation, just an exchange off arrangement is possible to deter the impact of the Sybil attack and moderate the contingent protection of the clients in the meantime [4]. Without tending to these issues, consumer loyalty will be tested, which will straightforwardly influence the possibility of these advances.

2 Methodology

There is sort of attacks like black hole attack and denial-of-service attack. Black hole attack is an error node procedure, and it is defeating protocol to promote itself taking the shortest path toward the destination node. At point route is set up, then error node forwards it to the malicious attacks wants address [5, 6].

Nodes decline to appreciate the system or when a setup node drops out. All system traffics are occupied to a specific node, which does not exist at all that, makes those data be lost.

There are two opportunities for black hole attack in VANET [7].

(1) Correspondence of new client with another client and other client declines to communicate with messages of too bad. At that point, the new client tries with next; however, the position is still same.
(2) When the correspondence begins with other client and all of a sudden dropout the communication interface with the neighbor vehicle and the aggravation will

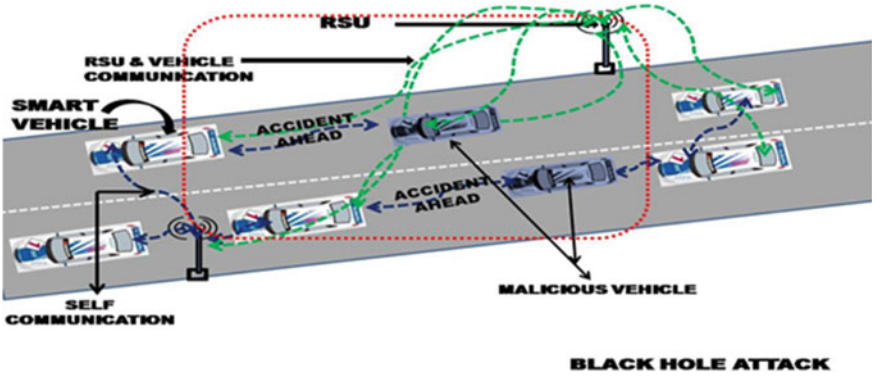


Fig. 1 Black hole attack [7]

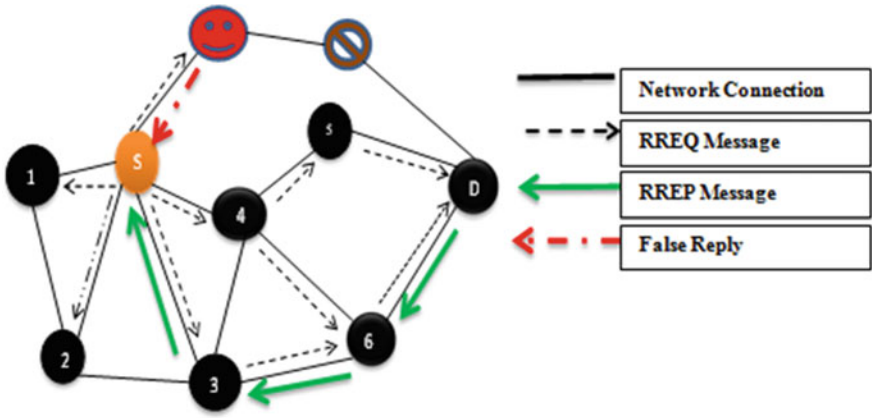


Fig. 2 Black hole attack topology

happen. Since this vehicle playing out the routing task, numerous vehicles were associated with it as the switching customer in (Fig. 1).

The black hole attack must make RREP with destination arrangement more noteworthy than the destination arrangement of the receiver node, and sender node trusts that black hole node and additional interconnects with black hole node in its place of the real destination node. This mischievous frequently harm nodes interface and thus waning all asset usage in accumulation to losing packets [8, 9].

In this scenario, black hole attacks are represented which act as a malicious node within the network topology as shown in (Fig. 2).

How to apply black hole attack in GPCR, check the below correction.

- Need to change GPCR protocol source code and add **Addnsaddr_t malicious**; inside GPCR routing protocol file. When we applied this malicious on the vehicles, then that vehicle behaves as a black hole attacker.

- When these malicious vehicles are performed inside the network, then we need to classify the attacker species.
- A black hole has two probabilities which are mentioned below:
- First one is the vehicle exploits the GPCR protocol, such as GPCR protocol, to advertise itself as having a valid route from source to destination vehicle. Node which exploits GPCR protocol route is spurious, which has the intention to make intercepting packets.
- Second one is the vehicle which exploits GPCR protocol route that consumes the intercepted packets.

3 Results and Analysis

After the mathematics, integration, and algorithms simulated the performance of Black Hole-GPCR and GPCR with the help of network simulator 2 (NS-2.35). Here used a real road network topology. The scenario consists of 100, 200, 300, 400, and 500 s simulation time and 20, 40, 60, 80, 100 numbers of VANET nodes which are shown in the figure. The movement of presented road network nodes was generated with VANET network simulator [7]. For the evaluation considered, two protocols of the VANET networks—GPCR, BH-GPCR protocol with black hole attack—are developed and designed for a comparative study on the fundamental of QoS performance parameter [10] (Table 1).

A. **Performance Metrics**

(a) **Average end-to-end delay**: Average end-to-end delay expressed the average time which data packets passed to transmission from source nodes to destination however since all delays initiated by buffering, queuing and propagation delays. Thus, average end-to-end delay somewhat depends on packet delivery ratio. When distance increased between source and destination, the probability of the packet drop is also increased. The mathematical

Table 1 Simulation parameter

Parameters	Values
Operating system	Linux (Ubuntu 12.04)
NS-2 version	NS-2.35
No. of node	20, 40, 60, 80, 100
Packet size	512
Traffic type	UDP/CBR
Simulation time	100, 200, 300, 400, 500 s
Antenna type	Omni-antenna
Transmission range	1000 * 1000 m
Routing protocol	GPCR, BH-GPCR

formula of average end-to-end delay (D) and the total number of packet delivery successfully (n) in this scenario is shown in Eq. (1).

$$Average\ end2end\ delay = \frac{\sum_{i=1}^{n}(\text{Received Packet Time} - \text{Send Packet Time}) * 1000(ms)}{Total\ Number\ of\ Packets\ Delivery\ Successfully} \quad (1)$$

(b) **Average network throughput**: The average network throughput expressed the total amount of data packets which successfully arrived at final destination as per given simulation time. The mathematical calculation of throughput shows here PacketSize is the size of a packet of ith packet reaching to the destination, PacketArrival is the time when the last packet arrived, and packet start is the time when the first packet arrived at a destination.

$$Throughput = \frac{PacketSize}{(PacketArrival - PacketStart)} \quad (2)$$

(c) **Packet Delivery Ratio (PDR)**: Packet delivery ratio expressed the ratio of total packets positively reached at the destination nodes source nodes. The network performance is high when packet delivery ratio is high in the network. The mathematical calculation of packet delivery ratio is shown in Eq. (3)

$$\text{Packet Delivery Ratio} = \frac{\sum \text{Total packets received by all destination node}}{\sum \text{Total packets send by all source node}} \quad (3)$$

(d) **Average Energy Consumption**: The average spent energy is calculated by total number of energy consumed for transmitted and received packets during the simulation in the networks. The total energy consumption is the summation of spending energy of overall nodes in the network, where the spent energy of a node is the summation of energy spend for communication, packet transmit (Pt), received packet (Pr), and idle packet (Pi). Assuming every transmission consumed energy unit, total energy consumption is equal to the total number of packets sent in the network.

B. Simulation Results

Several simulation scenarios on the different approaches were done. Here, we represent two different comparison scenarios of the present work (Table 2, 3, 4 and 5).

The simulation results shows the following: X-axis denotes the simulation time and node variation, and y-axis shows the performance metrics parameter.

Average End-to-End Delay: The average delay of GPCR is increased with a number of nodes but after a 60-node delay is increased smoothly. The overall performance of average delay for GPCR black hole attack with respect to a number of node variation or simulation time is higher as compared to the GPCR (Fig. 3).

Table 2 Delay comparison table for black hole attacks using GPCR (BH-GPCR)

Black hole-GPCR delay on simulation time			Black hole-GPCR delay on node variation		
Simulation time (s)	GPCR	BH-GPCR	No. of node	GPCR	BH-GPCR
100	110	130.17	Nodes-20	100	130
200	117	130.17	Nodes-40	120	130.17
300	120	130.17	Nodes-60	123	131.09
400	100	130.17	Nodes-80	116	120.2
500	123	130.17	Nodes-100	119	125.64

Table 3 PDR comparison table for black hole attacks using GPCR (BH-GPCR)

Black hole-GPCR PDR on simulation time			Black hole-GPCR PDR on node variation		
Simulation time (s)	GPCR	BH-GPCR	No. of node	GPCR	BH-GPCR
100	83	88.95	Nodes-20	90	88.96
200	82	82.08	Nodes-40	95	94.77
300	84	82.65	Nodes-60	96	95.42
400	86	84.52	Nodes-80	94	94.21
500	88	87.73	Nodes-100	97	96.4

Table 4 Throughput comparison table for black hole attacks using GPCR (BH-GPCR)

Black hole-GPCR throughput on simulation time			Black hole-GPCR throughput on node variation		
Simulation time (s)	GPCR	BH-GPCR	No. of node	GPCR	BH-GPCR
100	39.1	54.18	Nodes-20	169.16	54.18
200	58.65	31.52	Nodes-40	59.32	57.72
300	34.13	33.18	Nodes-60	64.31	61.27
400	45.99	42.03	Nodes-80	80.97	62.94
500	41.81	39.4	Nodes-100	61.76	46.4

Table 5 Energy comparison table for black hole attacks using GPCR (BH-GPCR)

Black hole-GPCR energy on simulation time			Black hole-GPCR energy on node variation		
Simulation time (s)	GPCR	BH-GPCR	No. of node	GPCR	BH-GPCR
100	4.5	36	Nodes-20	4.5	36
200	4.5	36	Nodes-40	2.25	18
300	4.5	36	Nodes-60	1.5	12
400	4.5	36	Nodes-80	1.12	9
500	4.5	36	Nodes-100	0.9	7.2

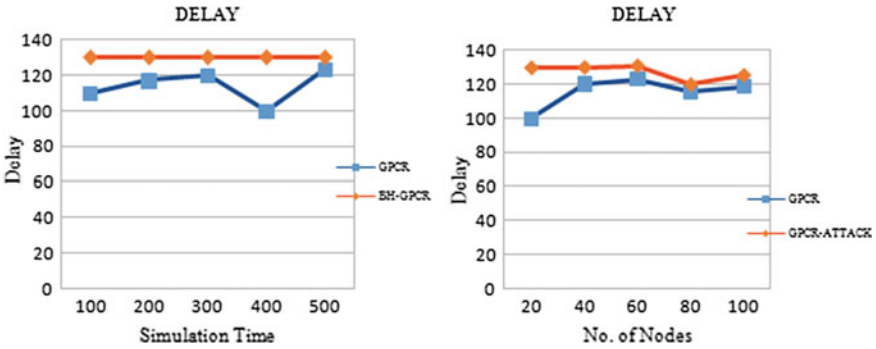


Fig. 3 Delay comparison for single and cooperative black hole attacks with respect to a number of node variation using GPCR

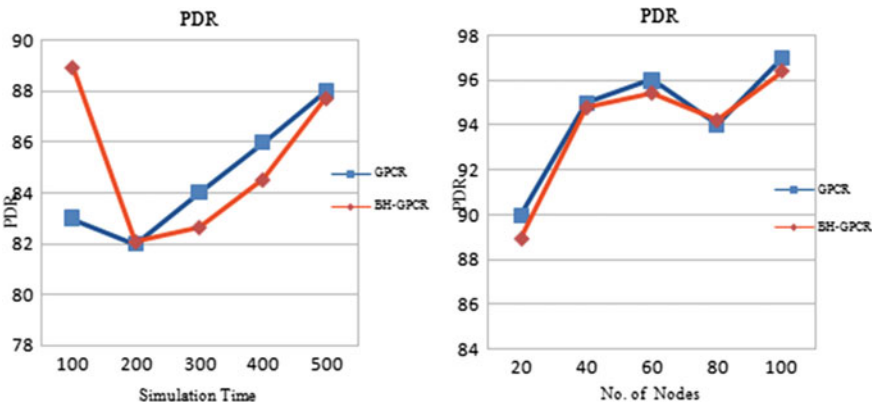


Fig. 4 Packet delivery ratio comparison for single and cooperative black hole attacks with respect to a number of node variation using GPCR

Packet Delivery Ratio: The performance of packet delivery ratio of black hole-GPCR is increased with 200 s simulation time and 20 nodes. With the variation of a number of nodes, GPCR routing protocol packet delivery ratio is same as black hole-GPCR, but with respect to simulation time, GPCR is better than BH-GPCR (Fig. 4).

Throughput: The performance of throughput for black hole-GPCR is almost same for nodes 40, 60 and 300, 400 simulation time, but throughput at 20 nodes is showing different performance as GPCR decreased (Fig. 5).

Energy Consumption: The performance of energy consumption of black hole in GPCR network continuously increased as compared to GPCR (Fig. 6).

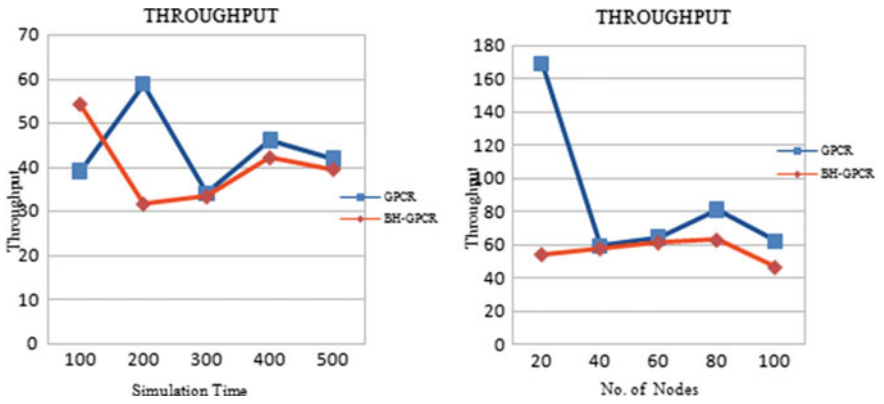


Fig. 5 Throughput comparison for single and cooperative black hole attacks with respect to the number of node variation using GPCR

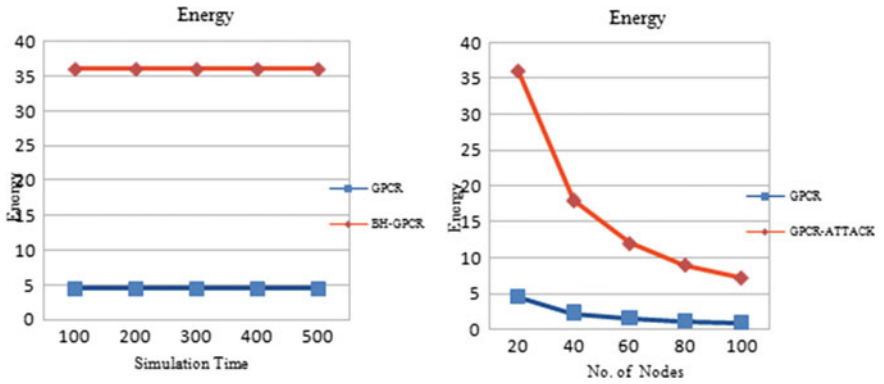


Fig. 6 Energy consumption comparison for single and cooperative black hole attacks with respect to the number of node variation using secure-GPCR

4 Conclusion

A black hole attack is one of the genuine security issues in any VANET network. It is an attack where a vindictive hub imitates a goal hub by sending fashioned RREP to a sourcing hub that starts course disclosure and therefore denies information movement from the source hub. In this paper, a review of various existing strategies for identification of dark opening attacks in VANET with their deformities is displayed. The discovery methods which make utilization of responsive directing conventions have low overheads and yet have high parcel misfortune issue. In light of the above execution correlations, it can be presumed that black hole attacks influence organize adversely. The recognition of black holes in impromptu systems is as yet considered to be a testing errand. Future work is expected to a productive Black Hole attacks

discovery and disposal calculation with least postponement and overheads that can be adjusted for impromptu systems helpless to Black Hole attacks. The overall performance of average end-to-end delay, packet delivery ratio, and throughput for black hole attack with respect to the number of nodes variation are GPCR performance better to BH-GPCR protocols.

References

1. Y. Gongjun, N. Mitton, X. Li, Reliable routing in vehicular ad hoc networks, in *2010 IEEE 30th International Conference on Distributed Computing Systems Workshops (ICDCSW)* (IEEE 2010)
2. H. Rasheed, et al., Privacy-aware route tracing and revocation games in VANET-based clouds, in *2013 IEEE 9th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)* (IEEE 2013)
3. M. Rostami, F. Koushanfar, R. Karri, A primer on hardware security: models, methods, and metrics. Proc. IEEE **102**(8), 1283–1295 (2014)
4. D.P.I.I. Ismail, M.H.F. Ja'afar, Mobile ad hoc network overview, in *2007 IEEE Asia-Pacific Conference on Applied Electromagnetics, APACE* (IEEE 2007)
5. Z. Sherali, et al., Vehicular ad hoc networks (VANETS): status, results, and challenges. Telecommun. Syst. **50**(4), 217–241 (2012)
6. B. Vimal, et al., Performance analysis of black hole attack in VANET. Int. J. Comput. Netw. Inf. Secur. **4**(11), 47 (2012)
7. H. Naziya, A. Singh, P.K. Shukla, In depth analysis of attacks & countermeasures in vehicular ad hoc network. Int. J. Softw. Eng. Appl. **10**(12) 329–368 (2016)
8. A. Rawat, S. Sharma, R. Sushil, VANET: security attacks and its possible solutions. J. Inf. Oper. Manag. **3**(1), 301 (2012)
9. B. Sourav Kumar, P.M. Khilar, A secure routing protocol for vehicular ad hoc network to provide ITS services, in *2013 International Conference on Communications and Signal Processing (ICCSP)* (IEEE 2013)
10. H. Jorge, J.C. Ruiz, P. Manzoni, Evaluating the usefulness of watchdogs for intrusion detection in VANETs, in *2010 IEEE International Conference on Communications Workshops (ICC)* (IEEE 2010)

An Approximation Algorithm to Find Optimal Rendezvous Points in Wireless Sensor Networks

Raj Anwit and Prasanta K. Jana

Abstract In recent times, mobile sink (MS) technology is one of the most popular approaches to collect data from sensor nodes. It has several advantages over a static sink network. To make data collection by MS more efficient, the path of the MS must be well designed. Several factors contribute in an optimal path of MS such as length of the path, number of halting points, and capacity of the MS. Location of the halting points is a major issue which affects the path length of the MS. Finding best position and number of halting points is very difficult and is considered as an NP-hard problem. Genetic algorithm (GA) is a popular and effective approximation approach for solving many NP-hard problems. In this paper, we present a novel approach based on GA to find the location of the rendezvous points (RPs) such that a good path for a MS can be obtained. We performed extensive simulation of our work and compared with other approach of tour planning. We also applied T test to judge statistical significance of our algorithm. The results demonstrate better performance of our algorithm in terms of the path length.

Keywords Approximation algorithm · Genetic algorithm · Path design · Mobility Data collection · Wireless sensor network

1 Introduction

A wireless sensor network (WSN) with static sink suffers from several problems such as unbalanced energy consumption, sink hole problem, and buffer overflow of sensor nodes (SNs) [1]. A mobile sink (MS) plays a vital role in alleviating these shortcomings effectively [2]. So, it has gathered attention of researchers in past few

R. Anwit (✉) · P. K. Jana
Department of Computer Science and Engineering, Indian Institute of Technology (ISM), Dhanbad, Dhanbad, India
e-mail: anwit_raj@yahoo.com

P. K. Jana
e-mail: prasantajana@yahoo.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_22

years [3]. A sink is made mobile by mounting it on humans, animals, vehicle, robots, etc. [4, 5, 6]. It moves throughout the network area to collect data from SNs. While using MS, it is not feasible to visit each and every node in a sensor network with large number of sensor nodes. It will increase the path length of the MS causing high consumption of energy of MS and buffer overflow of SNs. To handle these problems, a sojourn point-based approach is used where a MS visits a subset of SN called rendezvous point (RP). The SNs communicate data to these RPs which in turn sends the data to the MS directly. However, finding suitable location of RPs is very difficult and approximation algorithms can be used to find out the same.

In order to improve the efficiency of the RP-based approaches, better organization of the network is necessary. This includes finding the optimal location of RPs and aggregation of nodes so that the SNs do not overburden an RP. In [7], though a reduction in the number of halting points is achieved, it could have been improved by better organization of network into optimally placed RPs along with the MS. In [8], the MS visits each and every SN, and then, collection sites are combined. This will cause MS to spend high amount of energy, and also the data collection latency is high. In [9], the scheme is based on COM algorithm [8] in which some nodes can be skipped by MS. Both these approaches do not consider the optimal placement and number of halting points of MS.

Based on the above research works, we attempt to fill in the gap and propose a GA-based [10] method which finds out the locations of RPs which are optimal for the MS. The reason behind using GA is easy representation of the problem and simplicity to obtain a solution. Also, according to the no-free-lunch theorem [11], one cannot say that one metaheuristic technique always performs better than the other. The theorem says that the performance of all metaheuristic techniques when averaged over a large number of cases is same. We investigate the performance of proposed algorithm in the considered problem scenario. The proposed algorithm reduces the path length of the MS. Data collection latency gets reduced, and energy consumption of the network also becomes balanced using our approach. Our contributions can be summarized as follows.

- A novel approach to determine optimal location of the RPs
- Comparison of simulation results with an existing approach

The organization of the paper is as follows. Section 2 presents the literature review of some existing works. Section 3 presents basic overview of genetic algorithm. Section 4 presents the system model and terminologies used. Section 5 details the proposed algorithm. Sections 6 and 7 present the simulation work and conclusion, respectively.

2 Related Works

Many research works have been conducted in recent years for path or tour planning of the MS. The tour planning of MS is considered to be a TSP problem [12]. In [13], the authors use evolutionary approach to solve TSP problem. Although the

complexity of solving TSP is reduced, the problem of finding optimal placement of halting points (used as RPs) is not addressed. In [14], authors propose a weighted rendezvous point (WRP) approach where RPs are selected based on weights; instead, if better location of RPs was explored, then their performance improvement could have been achieved. In [15], the authors propose a binary search approach to find out better number of rendezvous points, but they have not considered their optimal location. Also, this algorithm fails to allow MS to pass through dense areas of the network. The COM algorithm [8] and SAS algorithm [9] also suffer from the same issues. In [16], authors have addressed clustering as a means to find RP of MS, but again finding good enough locations of cluster centers is not addressed by them so that the path length of MS becomes optimal.

3 Overview of Genetic Algorithm

Genetic algorithm (GA) was proposed by the researcher in [10]. It is an evolutionary optimization approach which mimics the human evolution process to reach a solution of the NP-hard problems. An NP-hard problem cannot be solved in polynomial time, so approximation algorithms are employed which give a good enough approximate solution to the problem. A group of solutions obtained from the diversified search space are called population. Each individual of the population is called as a chromosome. A chromosome is further composed of a set of individual genes. The other elements of this algorithm are fitness function, crossover, and mutation operation.

At first, a set of chromosomes is generated with randomly assigned values to the genes. This process is called initialization of chromosomes. The population is then subjected to crossover and mutation operation. Two randomly selected chromosomes (parents) undergo crossover to produce two child chromosomes. They further undergo mutation operation, with a hope to enhance their superiority. The fitness of the children is evaluated using a fitness function. If they are better than the parents, then they are included in the population; otherwise, they are discarded. The basic working principle of a GA is shown in Fig. 1.

4 System Model and Problem Formulation

We assume a WSN in which SNs are randomly deployed in the region. The region is assumed to be a square area of fixed dimension. There are some predetermined numbers of rendezvous points (RPs) to which SN sends data. The location of the RPs is to be searched. Each SN is assigned to an RP based on communication range and distance from the it. A MS collects data from SN by visiting RPs. The location of RPs is random. The assignment of SNs to RPs can be done using any clustering method. The mobile sink has enough storage and energy to collect data from SNs.

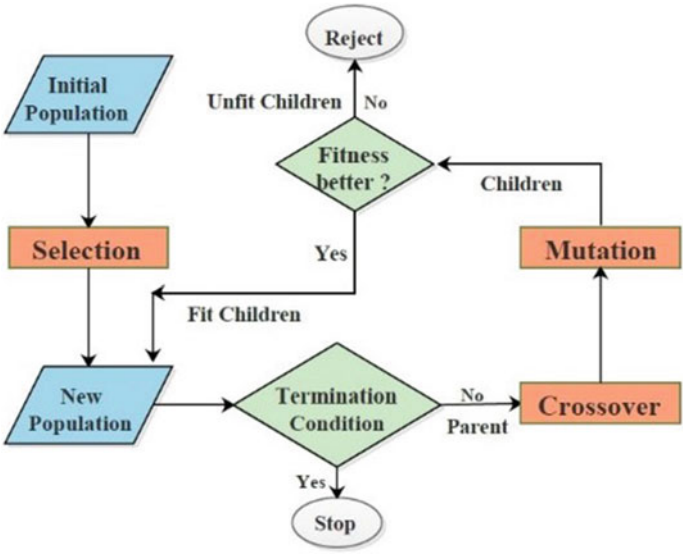


Fig. 1 Basic procedure of genetic algorithm

It starts from the base station (BS), visits the RPs, and returns back to the BS after data collection. The following terminologies are used in our proposed algorithm:

1. The set of sensor nodes is denoted by the symbol $S = \{s_1, s_2, \ldots, s_n\}$
2. The set of rendezvous points is denoted by $R_p = \{r_1, r_2, \ldots, r_m\}$
3. r_0 is the location of the base station.
4. $d'_{i,j}$ is the Euclidean distance between the ith sensor node and jth rendezvous point $1 \leq i \leq n$ and $1 \leq j \leq m$.
5. $d_{i,j}$ is the Euclidean distance between the ith rendezvous point and jth rendezvous point, $1 \leq i, j \leq m$.
6. d_{i,r_0} is the Euclidean distance between a rendezvous point and the base station, $1 \leq i \leq m$
7. P is the total Euclidean distance of the path travelled by the MS to collect data from sensor nodes, $P = \sum(d_{i,j} + d_{1,r_0} + d_{m,r_0})$
8. $|SN(r_i)|$ is the total number of sensor nodes covered by a rendezvous point r_i, $1 \leq i \leq m$

Now, we address the problem of reduction of path length of the MS. The problem depends on mainly two factors, location of the RP and combination of the RPs to form a path for the MS. Finding optimal combination of these factors will lead to an efficient path for the MS. Using the notations above, we state our objective as follows:

$$Minimize(P) \tag{1}$$

where

$$P = \sum (d_{i,j} + d_{1,r_0} + d_{k,r_0}),\ 1 \leq i, j \leq k, \quad k = no.\ of\ RPs\ in\ a\ chromosome \tag{2}$$

subject to

$$|SN(r_1)| + |SN(r_2) + \cdots + |SN(r_k)| = n \tag{3}$$

and

$$d_{i,j} \gg 0,\ 1 \leq i, j \leq k \tag{4}$$

The constraint in Eq. 3 ensures that all sensor nodes are covered by the set of RPs in a chromosome. This is the first condition for validity of a chromosome. The constraint in Eq. 4 ensures that the RPs are not infinitesimally close to each other such that the solution obtained collapses in the scenario considered in this paper. The second condition for validity of a chromosome is imposed by Eq. 4.

5 Proposed Work

We first present the representation of chromosome, initialization of population, derivation of fitness function in the earlier subsections, and then selection, crossover, and mutation operations in the remaining subsections.

5.1 Chromosome Representation

A chromosome in our research proposal is represented as a sequence of genes, where each gene represents a rendezvous point. The number of RPs is a predetermined number, so the size of the chromosome is also fixed.

Example: Let us consider a WSN of area size $100 \times 100\,m^2$ having 100 sensor nodes and 10 rendezvous points. So, the length of the chromosome for this network is 10. Figures 1 and 2shows a sample chromosome representation.

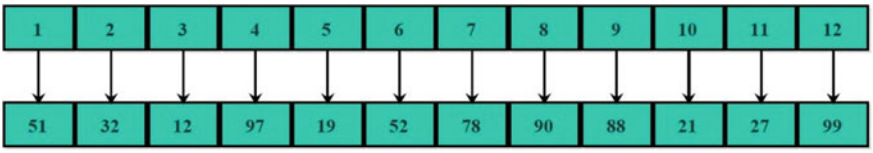


Fig. 2 Chromosome representation

5.2 Population Initialization

A group of randomly generated chromosome is the initial population. Each chromosome is a string of RPs. The chromosomes generated are valid, and the value at ith gene position is the ID of a randomly generated RP. It is noteworthy that this process is independent of any specific algorithm. All chromosomes represent a complete path for the MS. We illustrate this process with the help of an example.

Example: Let us consider a WSN of area size $100 \times 100\,\mathrm{m}^2$ having 100 sensor nodes and 12 rendezvous points. In this case, length of the chromosome is 12. Each rendezvous point inside the sensor network area has IDs between 1 and 100. Each of the ith position of the chromosome is assigned an ID from the set of IDs of RPs.

5.3 Fitness Function

We establish a fitness function to evaluate each of the chromosomes of the initial population. It is worth mentioning that the path length of the MS is proportional to the energy of an MS, and the location of the rendezvous points is proportional to the path length of the MS. Hence, the two relations can be mathematically represented as follows.

$$P \propto Energy(MS) \tag{5}$$

where $Energy(MS)$ represents the amount of energy spent by the MS to go through the path and

$$Energy(MS) \propto d_{i,j}, \quad 1 \leq i, j \leq m \tag{6}$$

So, from Eqs. 5 and 6 we consider the following fitness function.

$$F = P \tag{7}$$

A proper selection of coordinates of each rendezvous point is necessary because only then the most efficient path for the MS is possible. A smaller value of F will indicate a smaller path of the MS and comparatively higher fitness for a chromosome.

5.4 Population Selection

The initial population is further refined using a selection process. The selected population participates in the crossover and mutation operation to produce child chromosomes. Each individual has a fitness value, and the individuals with low value of

F will have higher chances of selection. Different selection methods are available for the selection purpose such as Roulette-wheel selection, tournament selection, and rank selection. We use Roulette-wheel selection to filter the initial population.

5.5 Crossover

Two randomly selected chromosomes undergo crossover operation. Different ways exist for this operation such as single-point crossover and multi-point crossover. We use single-point crossover for the sake of simplicity. In this process, a point is chosen at random and the chromosomes exchange their information at that point. It is noteworthy that for single-point crossover the point must be other than the first and last point of the chromosome. The process is illustrated in Fig. 3. It can be seen that the Parent 1 exchanges values after the gene value 13 and Parent 2 exchanges values after the gene value 52. The resultant child chromosome shares the path segment of both the parent chromosomes.

5.6 Mutation

Mutation is applied at specific positions of the child chromosomes. We use the center position of the chromosome length to mutate the value of the gene. The value, i.e., ID of the rendezvous point that replaces the gene value, is selected randomly. This is analogous to the natural mutation process where either the chromosome fitness improves or it degrades. We emphasize that the mutation process is applied in such

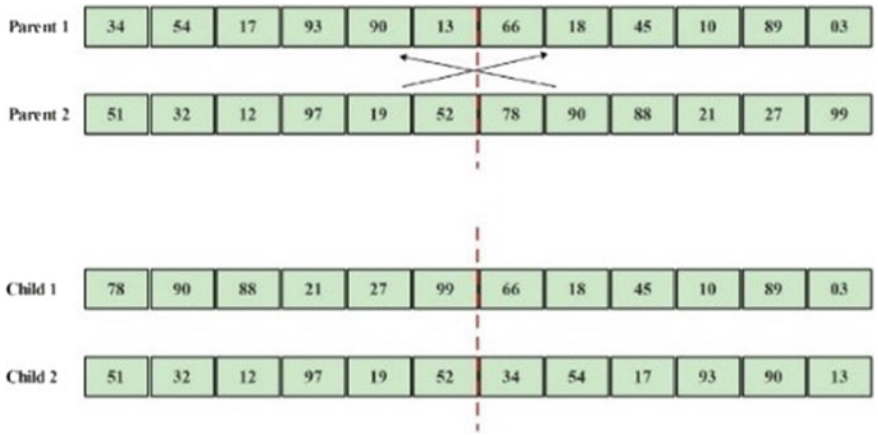


Fig. 3 Crossover operation

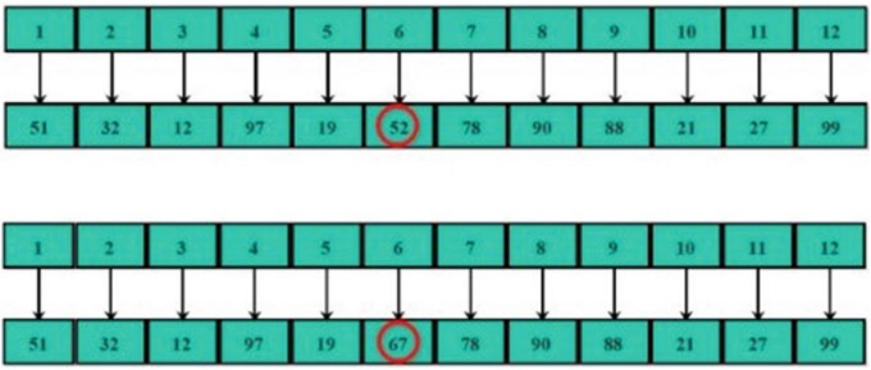


Fig. 4 Mutation operation

Table 1 Simulation parameters

Parameter	Values
Area (square)	$100 \times 100\,m^2$
No. of iterations	500
No. of sensor nodes	50–100
Communication range	30
Population size	1000
Dimension (dim)	0.4 * Population size
Crossover point	0.5 * dim
Mutation point	0.5 * dim

a way that same RP does not get repeated in a chromosome and it does not hinder the validity of the chromosome. We illustrate the mutation process in Fig. 4 with the help of a sample chromosome. We select the value at position 6, i.e., 52, and replace it with a randomly selected value 67. We made sure that the same ID of RP does not appear twice. It can be observed that the resultant chromosome is also a valid chromosome.

6 Simulation and Results

We performed extensive simulation of our proposed scheme using Python 3.5 under Anaconda environment running on Ubuntu 17.04 platform. The simulations were performed with diverse number of sensor nodes varying from 50 to 100 and area size of $100 \times 100\,m^2$ and $200 \times 200\,m^2$. The dimension of chromosome was kept fixed to 40. We used population size of 1000 chromosomes. The different parameters of simulation are listed in Table 1.

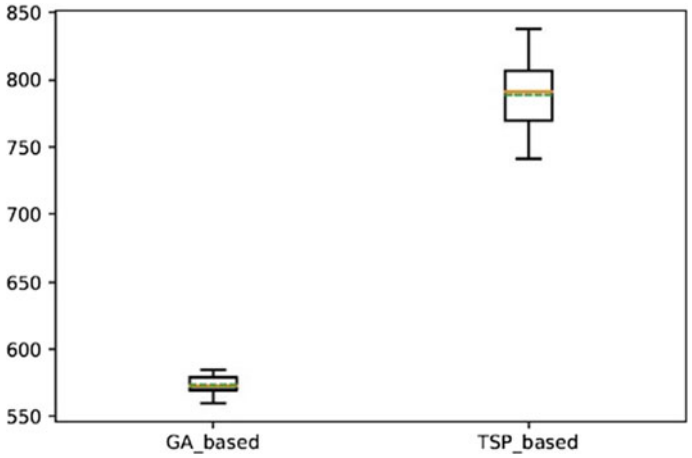


Fig. 5 Comparison of path length (area size = 200)

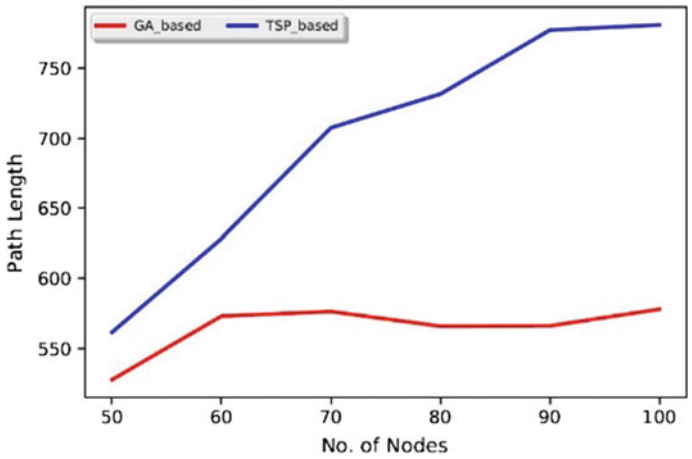


Fig. 6 Comparison of path length (area size = 100)

Figure 5 shows a box plot of our GA-based approach and TSP-based approach. We have taken a set of 20 values with area size 200×200 m^2 and communication radius of 30 m. It shows the variation between the means of the readings for the algorithms. Figures 6 and 7 show the comparison between the path lengths obtained with node number variation from 50 to 100 and area size of 100×100 m^2 and 200×200 m^2, respectively. It can be observed that our algorithm has less path length and hence performs much better.

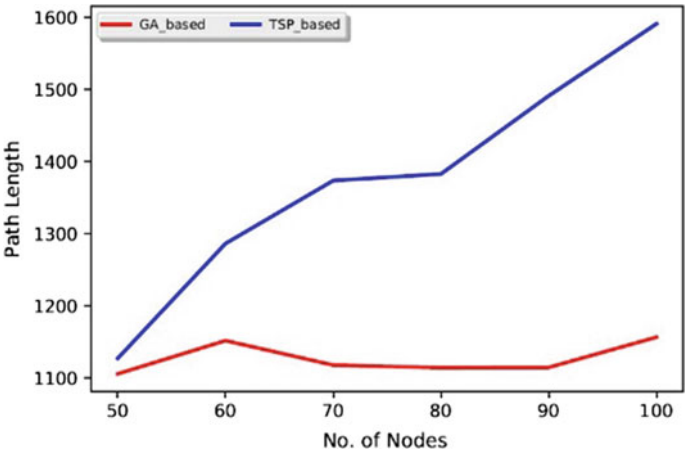


Fig. 7 Comparison of path length (area size = 200)

6.1 T Test Analysis

We apply hypothesis testing to check statistical significance of our results. We use Student t test for this purpose, since we use two groups of data, one for GA-based approach and other for TSP-based approach. We performed the simulation 20 times to obtain the sample data with area size $100 \times 100\,\text{m}^2$ and $200 \times 200\,\text{m}^2$. The null hypothesis was considered as:

$$H_0 : \mu_{GA} = \mu_{TSP} \tag{8}$$

which indicates that there is no difference in means of the groups of data of the algorithms GA based and TSP based.

The alternate hypothesis was considered as:

$$H_1 : \mu_{GA} \neq \mu_{TSP} \tag{9}$$

which indicates that there exists difference in means of the groups of data of the algorithms GA based and TSP based. It is known that t_{stat} represents the calculated value, whereas t_α represents the values obtained by looking into the t-distribution table. According to the rule, H_0 is rejected if $t_{stat} > t_\alpha$; otherwise, H_1 is accepted. Also that if the $p-value$ is less than α, then H_0 is rejected.

It can be observed from Tables 2 and 3 that the $p-value$ obtained is much less than the value of α in both the cases. Therefore, we can conclude that the null hypothesis (H_0) can be rejected and the alternate hypothesis (H_1) can be accepted in both the scenarios. The other output values of the t test can also be seen from Tables 2 and 3.

Table 2 T test output values 1

Area size	$100 \times 100\,m^2$		$200 \times 200\,m^2$	
Parameters	Mean	Variance	Mean	Variance
GA_based	573.798	46.747	1141.116	198.358
TSP_based	788.578	638.129	1590.924	3670.303

Table 3 T test output values 2

Area size	Pearson correlation	Observed mean difference	Variance of the differences	df	t-stat	P (T<=t) one-tail	t critical one-tail	P (T<=t) two-tail	t critical two-tail
100	0.677	−214.79	450.951	19	−45.232	4.11E-21	1.729	8.22E−21	2.093
200	0.319	−449.808	3323.505	19	−34.893	5.38E-19	1.729	1.07E−18	2.093

7 Conclusion

In this research proposal, we presented a GA-based approach to find out the good enough locations of the rendezvous points. We described the proposed algorithm with appropriate chromosome representation, initial population generation, selection procedure, and then crossover and mutation operators. We compared the performance of our algorithm with TSP algorithm using different simulation parameters. The simulation results showed better performance of our algorithm over the TSP algorithm in terms of path length of the MS.

References

1. A. Abuarqoub, M. Hammoudeh, B. Adebisi, S. Jabbar, A. Bounceur, H. Al-Bashar, Dynamic clustering and management of mobile wireless sensor networks. Comput. Netw. **117**, 62–75 (2017)
2. M.I. Khan, W.N. Gansterer, G. Haring, Static vs. mobile sink: the influence of basic parameters on energy efficiency in wireless sensor networks. Comput. Commun. **36**(9), 965–978 (2013)
3. B. Suh, S. Berber, Rendezvous points and routing path-selection strategies for wireless sensor networks with mobile sink. Electron. Lett. **52**(2), 167–169 (2015)
4. O. Tekdas, V. Isler, J.H. Lim, A. Terzis, Using mobile robots to harvest data from sensor fields. IEEE Wirel. Commun. **16**(1), 22–28 (2009)
5. M. Alnuaimi, K. Shuaib, K. Alnuaimi, M. Abdel-Hafez, Ferry-Based data gathering in wireless sensor networks with path selection. Procedia Comput. Sci. **52**, 286–293 (2015)
6. R.C. Shah, S. Roy, S. Jain, W. Brunette, Data mules: modeling and analysis of a three-tier architecture for sparse sensor networks. Ad Hoc Netw. **1**(2), 215–233 (2003)
7. C.F. Cheng, C.F. Yu, Data gathering in wireless sensor networks: a combine–tsp–reduce approach. IEEE Trans. Veh. Technol. **65**(4), 2309–2324 (2016)
8. Welzl, E., Smallest enclosing disks (balls and ellipsoids), in *New Results and New Trends in Computer Science* (1991), pp. 359–370
9. J. Tao, L. He, Y. Zhuang, J. Pan, M. Ahmadi, Sweeping and active skipping in wireless sensor networks with mobile elements, in *Global Communications Conference (GLOBECOM)*, (2012), pp. 106–111

10. J.H. Holland, Genetic algorithms. Sci. Am. **267**(1), 66–73 (1992)
11. D.H. Wolpert, W.G. Macready, No free lunch theorems for optimization. IEEE Trans. Evol. Comput. **1**(1), 67–82 (1997)
12. B. Yuan, M. Orlowska, S. Sadiq, On the optimal robot routing problem in wireless sensor networks. IEEE Trans. Knowl. Data Eng. **19**(9), 1252–1261 (2007)
13. M. de Berg, J. Gudmundsson, M.J. Katz, C. Levcopoulos, M.H. Overmars, A.F. van der Stappen, TSP with neighborhoods of varying size. J. Algorithms **57**(1), 22–36 (2005)
14. H. Salarian, K.W. Chin, F. Naghdy, An energy-efficient mobile-sink path selection strategy for wireless sensor networks. IEEE Trans. Veh. Technol. **63**(5), 2407–2419 (2014)
15. K. Almi'ani, A. Viglas, L. Libman, Energy-efficient data gathering with tour length-constrained mobile elements in wireless sensor networks, in 2010 *IEEE 35th Conference on Local Computer Networks (LCN)* (2010), pp. 582–589
16. N. Ghosh, I. Banerjee, An energy-efficient path determination strategy for mobile data collectors in wireless sensor network. Comput. Electr. Eng. **48**, 417–435 (2015)
17. K. Zhang, H. Du, M.W. Feldman, Maximizing influence in a social network: improved results using a genetic algorithm. Phys. A **478**, 20–30 (2017)
18. S. Didari, Y. Wang, T.A. Harris, Modeling of gas diffusion layers with curved fibers using a genetic algorithm. Int. J. Hydrogen Energy **42**(36), 23130–23140 (2017)
19. M. Elhoseny, A. Tharwat, A. Farouk, A.E. Hassanien, K-coverage model based on genetic algorithm to extend wsn lifetime. IEEE Sens. Lett. **1**(4), 1–4 (2017)

A Novel Approach for Gateway Node Election Method for Clustering in Wireless Mobile Ad Hoc Networks

Aayushi Jain, Dinesh Singh Thakur and Vijay Malviya

Abstract In few years, numerous efforts have been dedicated on clustering methods to use in wireless and ad hoc networks to result efficient and correct routing procedures. According to clustering, afterward of cluster formation, devices of a cluster nominated as cluster member, cluster head and cluster gateway on the basis of different roles. Among them, gateway device acts as conceptual bridge between clusters. Here one or more device may act as gateway between same or different clusters. A cluster head may be elected as suitable gateway among multiple cluster heads in situation of they all are belonging to same cluster. Cluster head election methods may employ in suitable gateway elections. This paper presents a method to elect suitable gateway. The proposed method elects suitable gateway on the basis of high rank of devices which determined use of maximum cluster belongings, device neighbors and maximum remaining battery. Proposed method is used in NS-2 simulation considering network parameters and evaluated on the basis of different simulation parameters.

Keywords Mobile ad hoc networks · Clustering · Cluster head election
Gateway election methods · NS-2

1 Introduction

Nowadays, mobile ad hoc network application grows rapidly due to benefit of autonomous configuration and low maintenances. At the same time, mobile ad hoc networks have certain number of design issues like routing, scalability and security.

A. Jain (✉) · D. S. Thakur · V. Malviya
MIT, Indore, Indore, India
e-mail: jain.aayushi94@gmail.com

D. S. Thakur
e-mail: dinesh.engg22@gmail.com

V. Malviya
e-mail: vijaymalviya@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_23

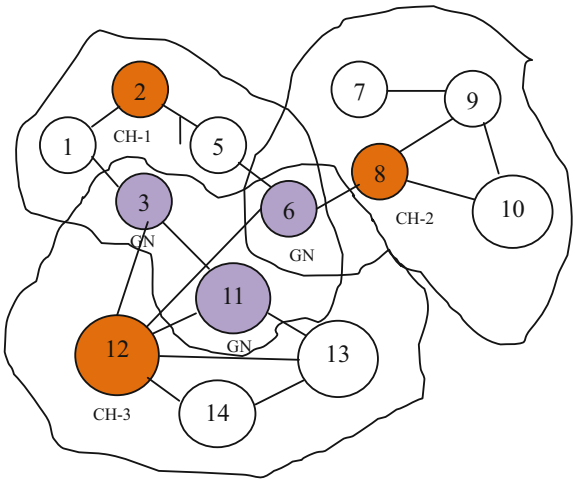


Fig. 1 One-hop cluster formation

Here, clustering is a method to deal with scalability and routing issue in mobile ad hoc networks. Network topology logical splits into definite groups. Clustering works in cluster formation, cluster head election and gateway nomination steps. According to network diameter, two types of cluster formation ways are recognized: one-hop clustering and multi-hop (k-hop) clustering. Every node is one-hop neighbor of the cluster head in case of one-hop clustering [1]. Another k-hop clustering is defined as k-hop distance between cluster head and member. In this paper, one-hop clustering is focused.

To manage the cluster behavior like maintaining cluster procedure, information of routing and new route detection, a node is appointed as cluster head by election. Other nodes are called member nodes or simply nodes. Some nodes can communicate with more than one cluster having interred cluster links called GN, that is, gateway nodes. If the purpose is within the cluster, ordinary nodes send the packets to their cluster head that distributes the packets within the cluster, or if it has to be delivered to additional cluster, then they forward them to a gateway node. In such a way, only cluster heads and gateways take part in the propagation of routing update or control [2].

Figure 1 shows one-hop cluster topology of network where one-hop clustering method is applied. This figure shows cluster head, cluster member and gateway nodes.

Clustering is very important for better performance of network. Depending on the network where different clustering algorithms apply [3].

2 Background

Clustering is a group of which nodes uses some approach to forward data effectively. In clustering, a cluster head is elected for each cluster. An elected cluster head node manages a list of nodes that are within the same cluster with a path to each of these nodes. Proactive approach is used for path update. Cluster head also maintains a list of gateways belonging to the neighboring clusters.

Various clustering algorithms have been proposed in [4–11] for ad hoc networks. Clustering algorithms are mainly used for the purpose of cluster head selection. Cluster head is any node which coordinates different tasks within the cluster. Ordinary nodes that wish to send packet first send those packets to the head of their cluster. It is the responsibility of the head of cluster now to deliver the packet properly to its destination. If destination lies within the same cluster, it can be delivered directly as cluster head has the information about other ordinary nodes which are in its cluster. But if the destination node lies in some other cluster, it takes help of a special node known as a gateway.

Therefore, gateway node works as bridge to transmit data from one cluster to another, maybe one or more clusters. Sometimes it may happen multiple gateway belongs to same two clusters. Here a decision is required to select one gateway for forwarding data. For this, appropriate method is needed to elect suitable gateway node among many. Next section presents several existing methods for electing gateway node.

3 Relevant Works

This section presents noteworthy contributions toward election of suitable gateway node in cluster network.

Fuzzy logic system-based steady load balancing gateway election [12] is advised here. The fuzzy system derives a new routing metric named cost which considers several network performance variables to select the best possible gateway. For solving the problem of fuzzy sets they are optimized by a genetic algorithm whose fitness function again employs fuzzy logic.

Based on path, load balancing, further adaptive steady load balancing gateway selection is implemented [13] which is implemented by calculating the load along a path and takes into account the route queue length metric. To address load balancing issue, gateway discovery algorithm is proposed [14].

Moreover, to address network performance, ideally scheduled route optimization (ISRO) method is proposed [15] which is the combination of three separate problems of optimization: optimal routing of gateway traffic under ideal conditions, route adjustment in light of the new link capacities and interference-free scheduling to determine link capacity.

4 Proposed Methodology

To elect suitable gateway node, a method is proposed that is known as node rank-based gateway election (NRBGE). NRBGE method uses one-hop neighbor information concept to elect gateway node. According to NRBGE, cluster head chooses high-rank node as a gateway. The rank of gateway nodes is defined by rank (R) which is determined by score calculation. Therefore, to determine a rank number of belonging clusters, a number of neighbors and remaining battery of node have considered. However, rank (R) is factorized in r1, r2 and r3 with respect to number of belonging clusters, number of neighbors and remaining battery. The value of r1, r2 and r3 is constant, and the sum of all should be 1. A formula to calculate rank (R) of node is defined in equation below.

$$R = r1 * BC + r2 * NN + r3 * B$$

where

BC–BC	is belonging clusters
NN–NN	is number of neighbors
B–B	is remaining battery of node
r1, r2 and r3	are ranking factors.

Proposed method works in two steps. First, each node belongs to same clusters by its rank via calculating score. Then, nodes exchange their rank value with every cluster head to maintain record. Node rank exchange can be achieved through periodic exchange of modified hello message. In general, hello message leads to discovery of neighbor in wireless communication. Hello signal is broadcasted in the form of message that has number of fields. General format of hello message is defined below.

Src. NID	Hello_Int	Ref_Int	VLT	Seq No	Neighbor ID	.	.	.	Options

For the NRBGE method, hello message is modified by defining new field, i.e., rank in existing format. The modified hello message is shown below.

Src. NID	Hello_Int	Ref_Int	VLT	Seq No	Neighbor ID	**Rank**	.	.	Options

In the second step, neighbor table is constructed and updated at each node with the help of modified hello signal. Constructed table keeps records of every reachable node and its rank value. Cluster head node uses the data to select high-rank node as gateway for further broadcasting.

The whole process of NRBGE method is designed in algorithm that is represented in pseudocode form.

Table 1 Network parameters and values

Parameters name	Value
Number of nodes	100
Dimension of simulated area	800 × 600
Simulation time (s)	45
Radio range	300 m
Traffic type	CBR, 3 pkts/s
Packet size (bytes)	512
Routing protocol	AODV
Connection type	TCP

Algorithm
Proposed method achieves its goal via the following steps.

Step-1 Number of belonging clusters, neighboring nodes and remaining battery are taken as inputs.
Step-2 Processing means rank calculation.
Step-3 It gives output in the form of higher score value node list.

5 Simulations

The method is simulated in NS-2 by considering various parameters that are defined in Table 1.

Simulation scenario—proposed method is simulated using Table 1 parameter that creates simulation scenario shown in Fig. 2.

6 Result Analyses

The performance of proposed method is calculated by assuming the following evaluation metrics.

- Throughput
- Routing overload
- Routing efficiency
- Battery efficiency.

Throughput—data units received per unit time are known as throughput unit which may be bit byte or packet.

Figure 3 shows the throughput of proposed method of gateway node election in which throughput determines in bytes/sec with respect to time.

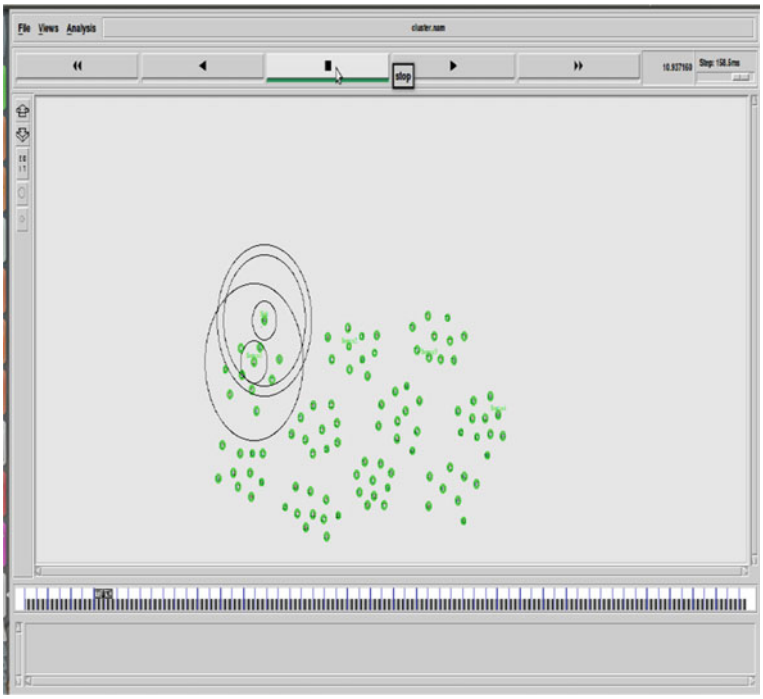


Fig. 2 Simulation scenario

Routing overload—routing overhead is defined by number of routing packets transmitted in network during route discovery and maintenance.

Figure 4 shows the routing overhead of proposed method of gateway node election in which overhead determines in number of routing packets with respect to time.

Routing efficiency—the ratio of data packets to the data and routing packets in the network is referred as routing efficiency.

Figure 5 shows the routing efficiency of proposed method of gateway node election in which efficiency determines as ratio of data packets and number of routing packets transmitted in network that is represented with respect to time.

Remaining battery—the remaining battery of nodes in the network after the network functioning is known as residual energy.

Figure 6 shows the remaining battery of proposed method of gateway node election in which it determines the energy of each node left after the complete transmission that is represented in joule unit with respect to node number. The different values of energy are resulted according to node number such as 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 … 99.

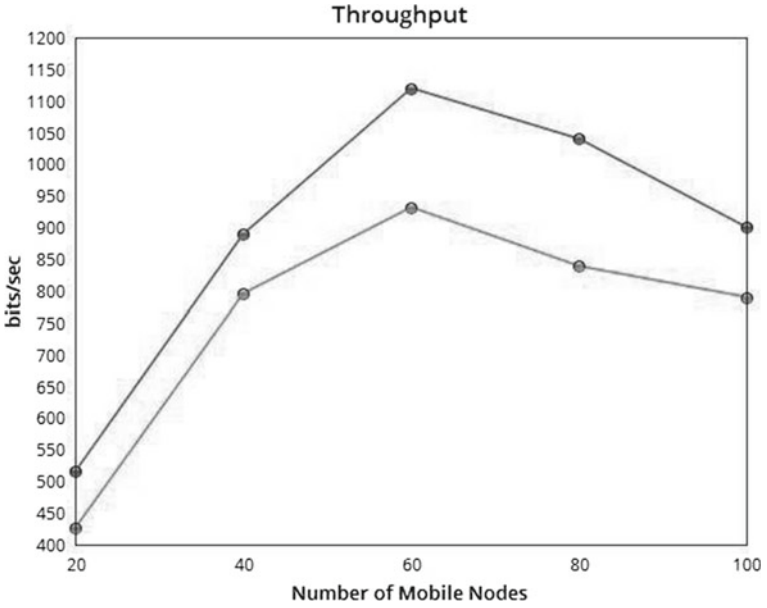


Fig. 3 Throughput

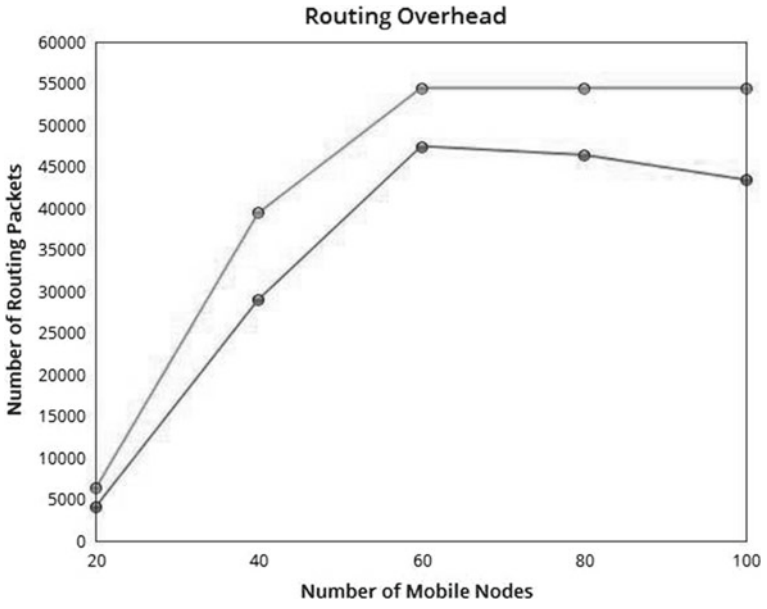


Fig. 4 Routing overhead

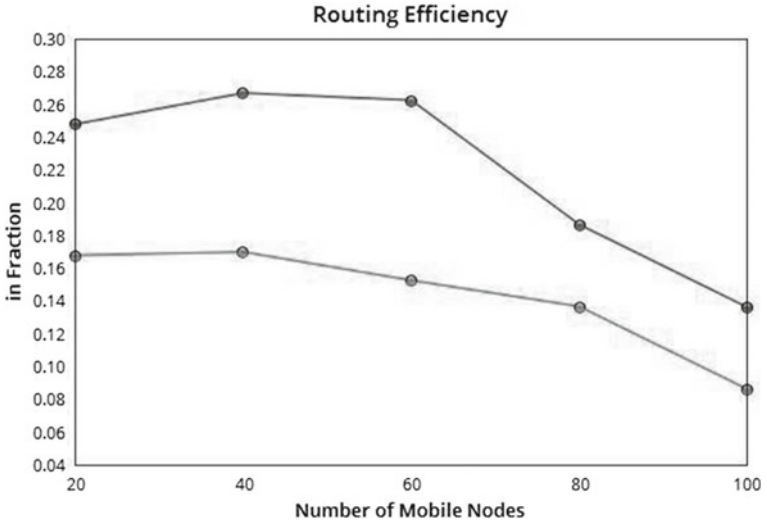


Fig. 5 Routing efficiency

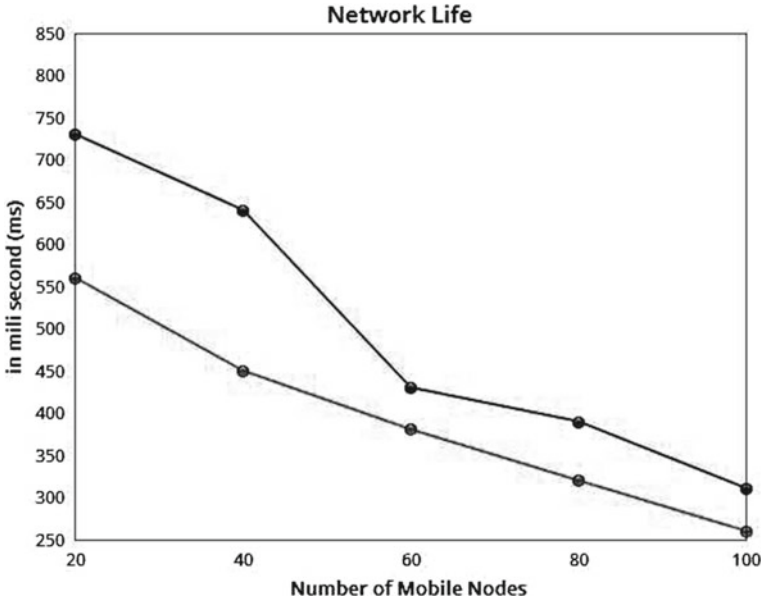


Fig. 6 Battery efficiency

7 Conclusion

With regard to mobile ad hoc network, routing is an essential approach in ad hoc network for data transmissions. To make efficient routing in high unpredictable network, clustering mechanism is evolved. The motive to clustering is to reduce routing overhead occurred due to flooding of routing packets by generic routing protocols. Clustering works by splitting network perimeter into logical regions. Each logical region is created by some criteria such as one-hop and k-hop neighbor. The whole control of constructed region is given to one capable device that is designated as cluster head. In clustering, nodes are designated as cluster head, gateway and member node according to these roles and responsibilities. Like cluster head election, suitable gateway election is also one procedure in clustering, which may enable suitable gateway for transmission. Proposed method offers election of gateway node to make more efficient clustering for mobile ad hoc network. Proposed method is simulated in NS-2 and evaluated considering various metrics that result in good performance.

References

1. S. Chinara, Analysis and Design of Protocols for Clustering in Mobile Ad Hoc Networks. Dissertation, National Institute of Technology Rourkela, 2011
2. K. Pradeepa, W. Regis Anne, Design and implementation issues of clustering in wireless sensor networks. Int. J. Comput. Appl. (IJCA), **47**(11) (2012)
3. R. Agarwal, M. Motwani, Survey of clustering algorithms for MANET. International Journal on Computer Science and Engineering **1**(2), 98–104 (2009)
4. M. Chatterjee, S.K. Das, D. Turgut, WCA: a weighted clustering algorithm for mobile Ad Hoc networks. Clust. Comput. **5**(2), 193–204 (2002)
5. A.R.H. Hussein, S. Yousef, O. Arabiyat, A load-balancing and weighted clustering algorithm in mobile Ad-Hoc Network
6. C.-C. Chiang et al. Routing in clustered multihop, mobile wireless networks with fading channel, in *Proceedings of the IEEE SICON'97* (1997)
7. J.-H. Ryu, S. Song, D.-H. Cho, New clustering schemes for energy conservation in two-tiered mobile Ad-Hoc networks, in *Proceedings of the IEEE ICC'01*, June 2001, pp. 862–66
8. R. Basavaraju, D.V. Ashoka, An enhanced geographical based minimal gateway selection method to improve connectivity in MANETs. in *Proceedings of International Conference on Emerging Research in Computing Information, Communication and Application, ERCICA 2013 Elsevier Journal*
9. S.H.Y. Wong, C.-K. Chau, K.-W. Lee, Dynamic Gateway Assignment to Support Inter Domain Networking in MANET
10. S. Ben Alla, A. Ezzati, A. Mohsen, Gateway and Cluster head election using fuzzy logic in heterogeneous wireless sensor network, in *2012 International Conference on Multimedia Computing and Systems (ICMS)* (IEEE)
11. Y. Lu, B. Zhou, I. Ku,M. Gerla, Connectivity improvement for inter-domain routing in MANETs, in *The 2010 Military Communications Conference—Unclassified Program—Networking Protocols and Performance Track*
12. A.J. Yuste-Delgado, J.C. Cuevas-Martinez, J. Canada-Bago, J.A. Fernández-Prieto, M.A. Gadeo–Martos, Improving hybrid ad hoc networks: The election of gateways. Appl. Soft Comput. J. (2015)

13. R.U. Zaman, A. Tayyaba, K.U.R. Khan, Enhancement of load balanced gateway selection in integrated Internet-MANET using genetic algorithm, in *2016 Fourth International Conference on Parallel, Distributed and Grid Computing (PDGC)* 22–24 Dec 2016
14. R.U. Zaman, A. Tayyaba, K.U.R. , Khan, *Path Load Balanced Adaptive Gateway Discovery in 2014 Fourth International Conference on Communication Systems and Network Technologies*
15. H. Livingstone, H. Nakayama, T. Matsuda, X. (Sherman) Shen, N. Kato, Gateway selection in multi-hop wireless networks using route and link optimization, in *IEEE Globecom 2010 Proceedings*

Security Vulnerability in Spectrum Allocation in Cognitive Radio Network

Wangjam Niranjan Singh and Ningrinla Marchang

Abstract The new technology of cognitive radio aims to provide opportunistic access to underutilized spectrum. Spectrum allocation which is one of the important functions of a cognitive radio network (CRN) assigns the unused spectrum to the unlicensed users. Several efficient spectrum allocation schemes have been proposed in the past which assume that the users are well behaved without any malicious behavior. But malicious users can adversely affect the spectrum assignment schemes and eventually affect the overall performance of the network. In this paper, we expose the vulnerability of CRN spectrum allocation to the Channel Ecto-Parasite Attack (CEPA). We also show the impact of this attack through simulation- based experiments.

Keywords CEPA · Spectrum allocation · Cognitive Radio · Security

1 Introduction

The increase in wireless network deployment due to increase in usage of wireless devices has saturated the spectrum giving rise to the spectrum scarcity problem. However, the spectrum which is assigned to the license users is poorly utilized. The inefficient usage of available spectrum led to a new technology known as cognitive radio (CR) which was introduced by Mitola in 1998. The CR is an intelligent radio which can sense and adapt accordingly to the environment. The main task of CR is to obtain the best available spectrum through cognitive capability and reconfigurability. A network of cognitive radio is known as cognitive radio network (CRN) [1]. The primary users (PUs) are those which have been licensed to use a certain

W. N. Singh (✉) · N. Marchang
Department of Computer Science and Engineering, North Eastern Regional Institute of Science and Technology, Nirjuli, Arunachal Pradesh, India
e-mail: niranwang@gmail.com

N. Marchang
e-mail: ningrinla@yahoo.co.in

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_24

spectrum band. The secondary users (SUs) do not have any such license, but they are opportunistic users which use free spectrum holes called white spaces.

Spectrum allocation is one of the important functions of a CRN which is also a key focus of research currently. Spectrum allocation is the assignment of free spectrum to the secondary users so as to achieve maximum channel utilization with minimum channel interference. Allocating spectrum in CRN poses different new challenges that do not arise in traditional wireless technologies such as Wi-Fi because CRs can dynamically adjust the frequency and bandwidth for each transmission, whereas traditional wireless networks use channels of fixed and predetermined width. In this paper, we explore the vulnerability that can arise in dynamic spectrum allocation schemes known as Channel Ecto-Parasite Attack (CEPA) in which the compromised node switches its interfaces to heavily loaded channels, i.e., highest used channel which decreases the overall performance of the network.

The rest of the paper is organized as follows. In Sect. 2, the related works are mentioned, followed by the system model in Sect. 3. Then, the proposed spectrum allocation method and the attack method is given in Sect. 4. In Sect. 5, we present the experiment evaluation result followed by conclusions in Sect. 6.

2 Related Works

There exist many research works on the problem of spectrum allocation in cognitive radio network. In some of these studies, conventional approaches based on centralized control [2, 3] are introduced. In some studies, the distributed approaches [4, 5] which do not need any central controller have been proposed. In some literature, we find cluster- based approaches [6–8] which are hybrids of centralized and distributed approaches which try to avoid the disadvantages of both the approaches. Different spectrum allocation techniques such as heuristics [6, 9], game theory [10, 11], linear programming [12, 13], network graph based [2, 3], nonlinear programming [14, 15] have been found. The spectrum assignment schemes proposed in [2, 4, 16] are designed for single radio interface. They are simple, and interference handling is easy, but if the channel is reclaimed by the primary user, the ongoing data communication will be interrupted. The techniques proposed in [17, 18] are designed for users with dual radios. The spectrum assignment proposed in [9, 12] is for multi-radio users. In such a scenario, when a primary user reclaims a channel, network partition does not occur. In [19], the authors exposed the different vulnerabilities in channel assignment in wireless mesh network. They evaluated the effectiveness of these attacks, and they found out that such attacks by some malicious node can significantly degrade the overall bandwidth and performance of the entire network. Motivated by the above works, we propose to study the impact of Channel Ecto-Parasite Attack (CEPA) in CRN.

3 System Model

We consider a CRN with n secondary users in which each user has two radios which can be used to access C available channels. We model the communication graph as an undirected graph G (V,E) where each node v ∈ V represents a secondary user and an edge e = (u, v) ∈ E represents the link between u and v if they are within the transmission range. Here G is a connected graph in which any two nodes are connected by either a direct link or a path with multiple nodes. Any two nodes can communicate with each other if they are on the same channel and are within the transmission range.

A channel assignment $\mathcal{A}$ generates a new undirected graph $G_{\mathcal{A}}(V, E_{\mathcal{A}})$ where $E_{\mathcal{A}}$ consists of the edges defined as follows. There is an edge e = (u, v; c) on channel c if (u, v) ∈ E and c ∈ $\mathcal{A}$(u)$\bigcap\mathcal{A}$(v) · $\mathcal{A}$(u) and $\mathcal{A}$(v) denotes the set of channels assigned to u and v. Multiple edges may exist between two neighboring nodes (u, v) if they share more than one channel, where one edge corresponds to one channel. Here, we consider all nodes have the same transmission range and the same interference range. Each link can only support transmission in one direction at one time; i.e., we have enforced half duplex mode of transmission.

4 Spectrum Allocation Method

In this section, we present the design philosophy and spectrum allocation method in detail. We use the following CRTCA [3] method for spectrum allocation.

CRTCA Algorithm

```
1.  Build the topology of the network ;
2.  Calculate the p(e) for each edge e ∈ E;
3.  Sort the edges E in the descending order of p(e);
4.  for each e in the sorted order of E do
5.    if A(u)<Q(u) and A(v)<Q(v) then
6.      c← the least used channel among the channels in C;
7.    else if A(u)=Q(u) and A(v)<Q(v) then
8.      c← the least used channel among the channels in A(u);
9.    else if A(u)<Q(u) and A(v)=Q(v) then
10.         c← the least used channel among the channels in A(v);
11.   end if
12. end for
```

Here $\mathcal{Q}$(v) represents the number of radios at node v and $\mathcal{Q}$(v) ≤ C. The algorithm first finds out the potential interference index denoted by p(e) for each edge e ∈ E in the network topology. The potential interference index of a edge is the cardinality of the set of edges that can potentially interfere with e. Mathematically, p(e) = |{(u, v)|(u, v) ∈ E, u or v ∈ D(e)}|. We sort all the edges of the network in the decreasing order of the potential interference index. Then, we go through each edge e = (u, v)

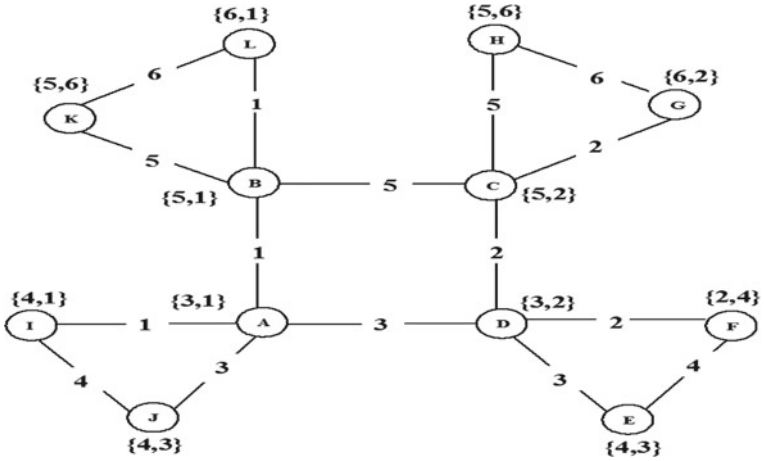


Fig. 1 Spectrum allocation under normal assignment

in the sorted list and assign channel c to e based on the following channel selection rule (as given in CRTCA algorithm).

First, if the number of assigned channels is less than the number of radios at both nodes u and v, the least used channel among the channels in C is selected (line no 6). Second, if the number of channels equals to the number of radios at u but number of channels is less than the number of radios at node v, then the least used channel from the set of channels assigned to u is selected (line no 8). Third, if the number of channels is less than the number of radios at u but number of channels is equal to the number of radios at node v, then the least used channel from the set of channels assigned to v is selected (line no 10).

In Fig. 1, we consider 12 nodes having two radios each and there are six available channels. The channel is assigned by using the above channel selection rule described above. The label at links denotes the set of channels shared by the two end nodes of that link, and the label with the node denotes the set of channels assigned to that node. The spectrum allocation under the above spectrum allocation method is represented in Fig. 1.

4.1 Channel Ecto-Parasite Attack (CEPA)

The main motive behind the Channel Ecto-Parasite Attack is to increase the interference at heavily loaded high-priority channels, i.e., highest used channel. We show how the CRTCA algorithm is modified by CEPA in the following CEPA algorithm:

CEPA Algorithm

```
1.  Build the topology of the network;
2.  Calculate the p(e) for each edge e ∈ E;
3.  Sort the edges E in the descending order of p(e);
4.  if edge is not infected by malicious node then
5.    for each e in the sorted order of E do
6.      if A(u)<Q(u) and A(v)<Q(v) then
7.        c← the least used channel among the channels in C;
8.      else if A(u)=Q(u) and A(v)<Q(v) then
9.        c← the least used channel among the channels in A(u);
10.     else if A(u)<Q(u) and A(v)=Q(v) then
11.           c← the least used channel among the channels in A(v);
12.       end if
13.     end for
14.   end if
15.   else
16.   if edge is infected by malicious node then
17.     for each e in the sorted order of E do
18.       if A(u)<Q(u) and A(v)<Q(v) then
19.         c← the highest used channel among the channels in C;
20.       else if A(u)=Q(u) and A(v)<Q(v) then
21.         c← the highest used channel among the channels in A(u);
22.       else if A(u)<Q(u) and A(v)=Q(v) then
23.         c← the highest used channel among the channels in A(v);
24.       end if
25.     end for
26.   end if
```

Under normal CRTCA algorithm, a node assigns the least used channel to the radio interfaces, but in CEPA a malicious node launches this attack by assigning its interfaces with the highest used channel. Figure 2 presents the spectrum allocation

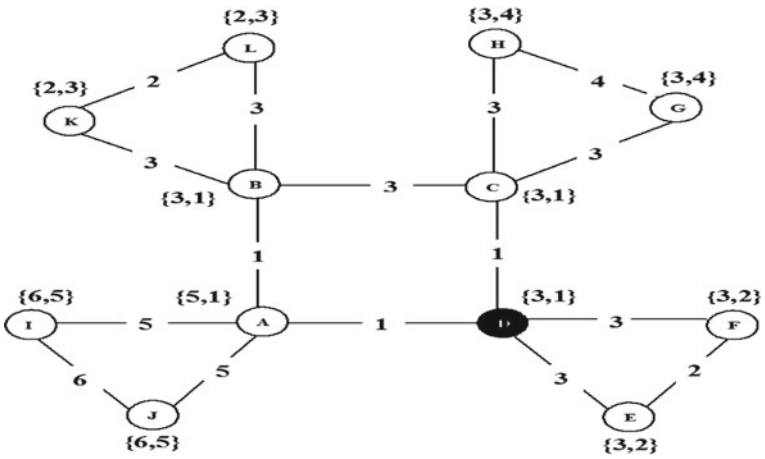


Fig. 2 Spectrum allocation under attack CEPA

under CEPA attack. Here, we introduce a malicious node D denoted by a darkened node. We can observe from the figure that with the introduction of attack, the channel assignment changes.

5 Simulation and Performance Evaluation

We have evaluated the CEPA attack through simulation-based experiments. We simulate the topology shown in Fig. 1. We inject four constant bit rate flows into the network. The source and destination for each flows are picked up randomly, and dynamic routing protocol (DSR) is used for routing.

First, we used probability of network partition as a performance metric to analyze the effectiveness of the attack. Figure 3 shows the probability of network partition versus number of channels under normal spectrum allocation and under attack. We observe that as the number of channels increase, the probability of network partition decreases as expected. However, there is a sharp contrast between the two scenarios: with attack and without attack. When there is no attack, the probability reduces to 0 from the point when the number of channels is 4. In contrast, when there is attack, the probability of partition remains at around 0.3 from the point when the number of channels is 3. We observe sudden variations in probability of network partition for under attack. This is because due to malicious behavior, many links or edges have been assigned with channel no 3. So, when primary user reclaimed this channel, network partition occurred.

Next, to measure the network performance, we measure throughput and average delay. Throughput can be defined as the rate of successful transmission of data packets. Average delay is defined as the average time taken by the data packets to propagate from source to destination across the network.

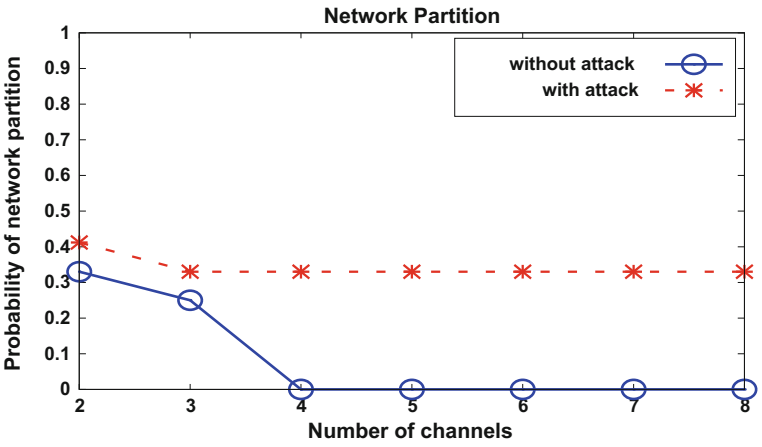


Fig. 3 Probability of network partition versus number of channel

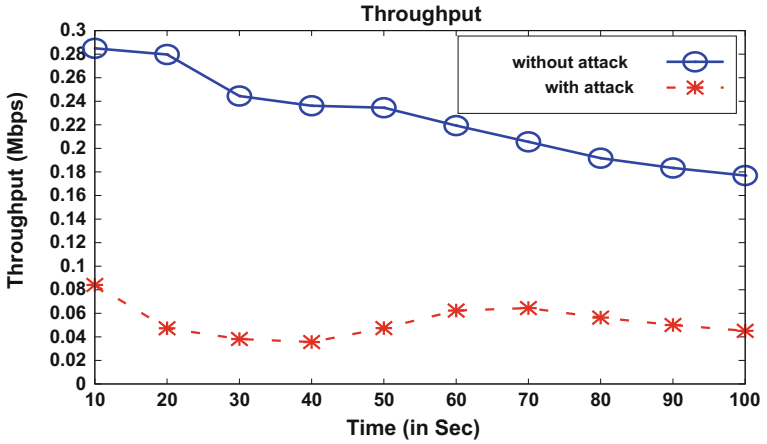


Fig. 4 Throughput versus simulation time

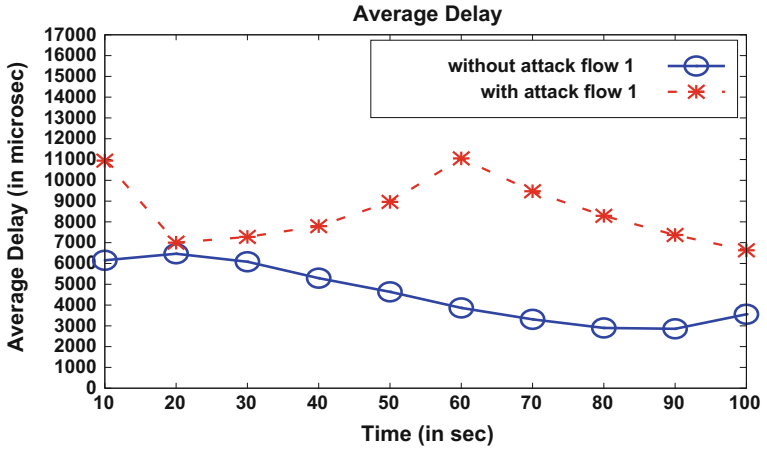


Fig. 5 Average delay versus simulation time (Flow 1)

Figure 4 shows the throughput (Mbps) versus simulation time (s) under normal spectrum allocation and under attack. We consider the scenario when the primary user reclaims channel no 3. We observe that throughput under CEPA attack is much less than when there is no attack. This is because when a primary user reclaimed channel no 3, network partition occurred. So, some packets are not able to reach the targeted destination. Hence, throughput is decreased.

Figures 5, 6, and 7 show the average delay (ms) versus simulation time (s) under normal spectrum allocation and under attack for different flows. We observe that average delay increases under the influence of CEPA attack. We observe that for the

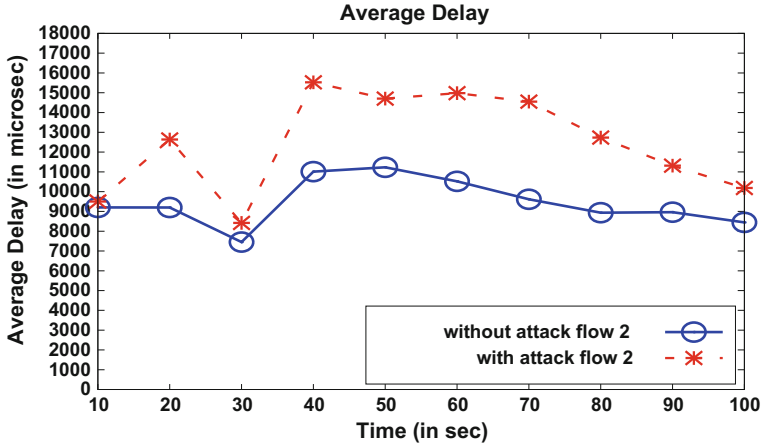


Fig. 6 Average delay versus simulation time (Flow 2)

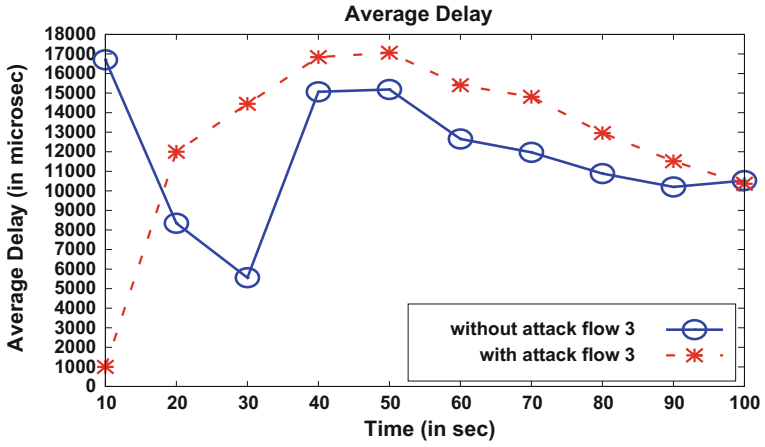


Fig. 7 Average delay versus simulation time (Flow 3)

fourth flow under attack, the network partitions occurs and packets are not able to reach the destination. Hence, it cannot be shown in a graph. However, when there is no attack network partition does not occur.

6 Conclusions

In this paper, we exposed a security vulnerability in spectrum allocation in cognitive radio network, i.e., the Channel Ecto-Parasite Attack (CEPA). It was found through experimental simulations that having such attack in spectrum allocation degrades the

overall performance of the network and also it results the network partition to occur. In future, we wish to explore ways of mitigating this attack.

Acknowledgements This work is an outcome of the R&D work undertaken in the ITRA project of Media Laboratory Asia, India, entitled "Mobile Broadband Service Support over Cognitive Radio Networks" (ITRA/15(63)/Mobile/MBSSCRN/02/2015).

References

1. J. Mitola, Cognitive radio: an integrated agent architecture for software defined radio. Ph.D. Thesis in Royal Institute of Technology, (KTH), 2000
2. C. Xin, L. Ma, C.-C. Shen, A path-centric channel assignment framework for cognitive radio wireless networks. Mobile Netw. Appl. **13**(5), 463–476 (2008)
3. J. Zhao, G. Cao, Robust topology control in multi-hop cognitive radio networks, in *2012 Proceedings of the IEEE INFOCOM* (Orlando, FL 2012), pp. 2032–2040
4. L.T. Tan, L.B. Le, Channel assignment with access contention resolution for cognitive radio networks. IEEE Trans. Veh. Technol. **61**(6), 2808–2823 (2012)
5. M. Hashem, S.I. Barakat, M.A. AttaAlla, Distributed channel selection based on channel weight for cognitive radio network, in *Proceedings of the 10th International Computer Engineering Conference (ICENCO)*, (Giza 2014), pp. 115–120
6. A. Alsarahn, A. Agarwal, Channel assignment in cognitive wireless mesh networks, in *IEEE 3rd International Symposium on Advanced Networks and Telecommunication Systems (ANTS)* (New Delhi 2009), pp. 1–3
7. T. Chen, H. Zhang, M. Matinmikko, M.D. Katz, CogMesh: cognitive wireless mesh networks, in *IEEE Globecom Workshops* (New Orleans, LO 2008), pp. 1–6
8. H. Pareek, A. K. Singh, An adaptive spectrum assignment algorithm in cognitive radio network, in *Proceedings of the 5th ACEEE International Conference on Recent Trends in Information, Telecommunication and Computing* (ITC 2014), pp. 408–418
9. W. Kim, A.J. Kassler, M. Felice, M. Di Gerla, Urban-X: towards distributed channel assignment in cognitive multi-radio mesh networks, in *Proceedings of the IFIP Wireless Days* (Venice 2010), pp. 1–5
10. H. Zhang, X. Yan, Advanced dynamic spectrum allocation algorithm based on potential game for cognitive radio, in *Proceedings of the 2nd International Symposium on Information Engineering and Electronic Commerce* (Ternopil 2010), pp. 1–3
11. Z. Wu, P. Cheng, X. Wang, X. Gan, H. Yu, H. Wang, Cooperative Spectrum allocation for cognitive radio network: an evolutionary approach, in *IEEE International Conference on Communications (ICC)* (Kyoto 2011), pp. 1–5
12. R.E. Irwin, A.B. MacKenzie, L.A. DaSilva, Resource-minimized channel assignment for multi-transceiver cognitive radio networks. IEEE J. Sel. Areas Commun. **31**(3), 442–450 (2013)
13. L. Yu, C. Liu, W. Hu, Spectrum allocation algorithm in cognitive ad-hoc networks with high energy efficiency, in *The International Conference on Green Circuits and Systems* (Shanghai 2010), pp. 349–354
14. D, Chen-li, Z, Guo-an, G. Jin-yuan, B. Zhi-hua, A route tree-based channel assignment algorithm in cognitive wireless mesh networks, in *Proceedings of the International Conference on Wireless Communications and Signal Processing* (Nanjing 2009), pp. 1–5
15. U. Pareek, D.C. Lee, Resource allocation in bidirectional cooperative cognitive radio networks using swarm intelligence, in *IEEE Symposium on Swarm Intelligence* (Paris 2011), pp. 1–7
16. D.H. Lee, W.S. Jeon, Channel assignment and routing with overhead reduction for cognitive radio-based wireless mesh networks, in *International Conference on Wireless Communications and Signal Processing (WCSP)* (Nanjing 2011), pp. 1–5

17. J. Wang, H. Yuqing, A cross-layer design of channel assignment and routing in Cognitive Radio Networks, in *Proceedings of the 3rd International Conference on Computer Science and Information Technology* (Chengdu 2010), pp. 542–547
18. E. Anifantis, V. Karyotis, S. Papavassiliou, A markov random field framework for channel assignment in cognitive radio networks, in *IEEE International Conference on Pervasive Computing and Communications Workshops* (Lugano 2012), pp. 770–775
19. A. Naveed, S.S. Kanhere, NIS07-5: security vulnerabilities in channel assignment of multi-radio multi-channel wireless mesh networks, in *IEEE Globecom* (San Francisco, CA 2006), pp. 1–5

Energy-Efficient Clustering in Wireless Sensor Network with Mobile Sink

Sonakshi Soni and Saumya Bajpai

Abstract Energy efficiency is a main concern for clustering–routing protocols in wireless sensor networks. A recent work has proposed a clustering technique for WSNs in which sink moves along a circular path, but it suffers from the hot-spot problem. This paper suggests how to solve this issue. To achieve uniform distribution of energy a ping-pong strategy is adopted. Cluster heads are elected according to a potential which is a combination of the proportional residual energy of a node with respect to its neighbours and its distance from the path of sink. Cluster formation is done based on a cost factor. The proposed method achieves a longer lifetime due to energy saving as demonstrated in simulation experiments.

Keywords Wireless sensor networks · Clustering–routing · Mobile sink
Ad hoc network

1 Introduction

Wireless sensor networks (WSNs) are continuing to be the most effective aid to sensing and monitoring tasks. Ability to work in harsh environments, easy implementation and high performances are some of the many reasons that contribute to the popularity of the sensor networks and its use in various applications such as defence and healthcare [1]. A WSN is an ad hoc network established among a large number of sensor nodes sensing a region or target to send their data to a sink or base station (BS) using wireless technology. A major issue is to prolong the life of a network by minimizing energy consumption as the nodes are battery-operated devices. Much of energy can be saved by having a clustered topology [2] where the cluster heads (CHs) collect data from member nodes, aggregate and transmit to the sink. This reduces

S. Soni (✉) · S. Bajpai
Department of Computer Science, Rajasthan College of Engineering for Women (RCEW), Ajmer Road, Bhankrota, Jaipur, Rajasthan, India
e-mail: sonakshi.soni2011@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_25

redundant packets floating in the network, thereby reducing energy consumption. Challenges of clustered topology are election of CH and load distribution.

Due to limitation of space, we briefly mention here few works related to our work. For a detailed survey on clustering and routing protocols in WSN the reader may refer to [3, 4]. A very popular approach towards clustered topology is LEACH [5] that constructs a hierarchy of packet routing between nodes and sink. Nodes at primary level transmit data to their next level nodes or CHs. The CHs at last level transmit to the sink. Intermediate levels of secondary CHs may or may not exist depending on the area, sensor population and location of BS. The CHs are selected from the nodes themselves on a rotary basis probabilistically. Though easy, the cluster head election technique is not good as it does not consider any resource-based election. Election of CHs based on their residual energy, distance from BS and the number of neighbours has been suggested by many later proposals [6, 7] and proved to be efficient. EEUC [8] suggests decreasing the size of cluster as the distance of CH from BS increases. This style of regulated uneven clustering is good when BS is at the centre of RoI. Ellateif et al. [9] proposed EED to differentiate between nodes as interior and exterior based on density of neighbours. The CH election and energy load distribution are based on this.

Works considering the mobility of nodes or BS are few. Banerjee et al. [10] have suggested having a few mobile nodes in the otherwise static network. These nodes act as CHs and collect data from nodes and then transmit it to the BS. The mobility of these nodes is controllable. Er and Seah [11] proposed a dynamic energy-efficient clustering algorithm (DEECA) for mobile nodes. It elects the CHs with high residual energy and low mobility. When the energy of a CH falls below a yellow threshold, some load balancing among the cluster members is done. If red threshold is reached then re-clustering is done. Jafri et al. [12] proposed MIEEPB to have mobile sink in improved energy-efficient PEGASIS-based protocol (IEEPB) [13]. The sink has a fixed trajectory for predictable position at all time instances. The purpose is to save the energy that chain leaders consume when the distance between the leader and sink is large. Amine et al. [14] proposed an energy-efficient and safe weighted clustering algorithm (ES-WCA) for mobile WSNs which are computed using five metrics. The behavioural level (BL) metric is a unique and very good contribution of the authors. Other criteria are mobility of node, residual energy, distance between node and its neighbours and degree of connectivity.

The recent proposal by Mazumdar and Om [15] proposes clustering in situations where BS moves in a circular path of fixed radius and uniform velocity. Few advanced nodes having as much as five times more initial energy than normal nodes are deployed in the network to act as CHs. So no CH election method has been used. Appointing advanced nodes as CHs is not wise as they become hot spots for load and sooner or later will deplete. Once the CH is dead or failed, there is no method to cope up with the situation. Hence, we propose in this paper a CH election method to be used with the topology suggested in [15].

The rest of the paper is organized as follows: Sect. 2 sets the preliminary knowledge required for the proposal; Sect. 3 describes topology construction and the proposed cluster head election method. Later, experimental results and evaluation are presented.

2 Model and Assumptions

2.1 Network Model

Let $\mathcal{N} = \{n_1, n_2, \ldots, n_s\}$ be the set of S homogeneous sensor nodes deployed randomly in RoI. Every node n_i has (x_i, y_i) as its location. Distance between any two nodes is the Euclidean distance denoted as $\|n_i - n_j\|$. The distance of any node from the BS is the shortest distance between the node and the path of BS, which is a circle of radius R. Thus, distance of a node from BS is $R - \sqrt{(x_i - R)^2 + (y_i - R)^2}$.

Definition 1 Connected Node and Connected Set—A sensor node is said to be a connected node if it has a CH within its communication range. The set of all connected nods is called the connected set.

Definition 2 Unconnected Node and Unconnected Set—A sensor node is said to be an unconnected node if it has no CH within its communication range. The set of all unconnected nods is called the unconnected set.

Definition 3 Relay Node and Relay set—The connected neighbour of n unconnected nodes through which the data packets of the unconnected node can be routed is the relay node. The set of relay nodes for any particular sensor node is the relay set of that node.

2.2 Mobility of the Base Station

We assume a BS or sink moving with uniform velocity v along a circular path of radius R with its centre at (R, R). At the time of initiation, the BS broadcasts its current location in the form of angle it subtends at the centre, denoted as θ. After a time interval Δt the change in position of the BS is $\Delta\theta$. The nodes can keep track of the current position of BS by computing $(\theta + \Delta\theta)\%360$, where $\Delta\theta = \frac{v \times \Delta t}{R}$. Any sensor node can also compute the angle it subtends at the centre of the RoI by computing $\Phi_i = \tan^{-1} \frac{R - y_i}{R - x_i}$. A sensor node sends packets to the BS only when $(\theta + \Delta\theta)\%360 = \Phi_i$. Thus, a schedule for communication is automatically produced.

2.3 Energy Model

The radio model of first order is used. The energy dissipated in the transmission and the receiver modes is denoted by E_{Tx} and E_{Rx}. E_{elec} is the energy used for running the transmitter or receiver circuitry. ε_{mp} is the energy used for multipath communication, and ε_{fs} is free space energy dissipation consumed to send a packet of k bytes to distance (d). E_{DA} is the energy consumed in aggregating data. The complete equations are as follows:

$$E_{TX} = E_{elec} \cdot k + \varepsilon_{fs} \cdot k \cdot d^2, d < D_0 \quad (1)$$

$$E_{TX} = E_{elec} \cdot k + \varepsilon_{mp} \cdot k \cdot d^4, d \geq D_0 \quad (2)$$

$$E_{RX} = E_{elec} \cdot k \quad (3)$$

$$E_{Agg} = E_{DA} \cdot k. \quad (4)$$

3 Proposed Protocol

3.1 Control Messages

The control messages used in the proposed protocol are as follows:

- INFO: nodes form this message bearing id and value of residual energy and send it to only nodes within communication range.
- CH-ADV: during CH election every node sends this message with its id and potential value to all the nodes within its communication range.
- CH-WIN: any node when discovers that it is winner of CH election broadcasts this message to nodes within the communication range to advertise in the network. The message has the id of the node and its location.
- JOIN: a non-CH node sends this message to the CH node it wishes to join as member. The message contains the requesting node's id.
- HELP: when a node discovers that it is an unconnected node, it broadcasts this message to find a relay node. This message contains the requesting node's id and its intended CH's id.
- ACK: a node that receives HELP message and is able to be a relay node according to the request sends back this message. This message states the acknowledging node's id and its residual energy.

3.2 Cluster Head Election

The CHs are elected based on the potential of a node to become a CH. The potential is computed from two criteria: the residual energy of a node and its distance from the path of BS as $\varphi_i = \frac{energy_i}{\sum_{\|n_i - n_j\| < range} energy_j} + \frac{1}{\|BS - n_i\|}$. Thus, a node having more energy than its neighbours has more potential to become a CH. But if two candidates for CH have equal energy factor, the node nearer to BS will win. Ideally, to save energy consumption, a CH should have high energy and be near to BS. But if in every round the nodes closer to BS are selected as CHs, their energy will get consumed faster as they may also be relay node for others. If the selection criterion has energy also as its factor, the nodes with higher energy than the previously selected nodes will become CH in next round. In successive rounds the combined factor of distance and residual energy will again make the nodes nearer to BS path as winners of CH election. This ping-pong strategy works well for load balancing. And the uniform energy consumption throughout the network gives longer lifetime.

3.3 Cluster Formation

A non-CH node picks any one CH to join according to lowest cost factor. The cost factor of node n_i selecting the CH n_j is computed as $CF(n_i, n_j) = E_{Tx}(n_i, n_j) + E_{Tx}(n_j, BS)$ where $E_{Tx}(n_i, n_j)$ is the energy consumed in transmission of data from node n_i to node n_j. And $E_{Tx}(n_j, BS)$ is the transmission energy consumed for sending data from node n_j to the BS. Taking the distance to CH alone as a deciding factor may not actually be a good decision. The cost factor computed as above ensures that the total cost of sending data is less.

3.4 Consolidated Protocol

At the time of deployment each node computes its Φ_i. All nodes broadcast and receive INFO messages. As per received INFO messages the sum of energies of the neighbours is computed and the value of potential φ_i is computed. Every node broadcasts the CH_ADV message carrying its id and φ_i. The nodes receive CH_ADV messages from neighbours. Every node that has maximum value of potential as compared to those received by it declares itself as winner. It broadcasts CH_WIN message carrying its id and location. The nodes that are not winners listen for all CH_WIN messages. Once the CH nodes are appointed, cluster formation is completed by each non-CH node joining one of the CHs. The non-CH nodes compute cost factors associated with every CH node from which it has received the CH_WIN message. It sends JOIN message to the CH it selects to join. In case a node discovers that the CH it has joined is not within its communication range, then it also sends HELP

message to its neighbour nodes within range. A node ready to become relay node for the requesting node responds with an ACK message.

After clustering, each CH assigns a time slot to each of its cluster members (CMs) using time-division multiple access (TDMA) method such that each CM sends its sense data to CH during the assigned time slot only. Sensing phase begins here. Each sensor node senses its region to collect data. The packets are formed and forwarded to the CHs. The CHs receive the data from their CMs and aggregate the received packets (along with their own data). The aggregate packets are finally sent to the mobile BS when the distance of the CH from BS is minimal. This is possible for all CHs at unique time instance due to different positions. Hence, no TDMA or other scheduling is required for the CHs.

The clustering protocol is repeated after every $\alpha \geq 1$ complete rounds of communication protocol. One complete round implies one revolution of BS around the region. Hence, every CH transmits data to BS only once in a round. The time duration of one round is $\frac{2\pi R}{v}$. Since it is a constant, every CH knows when it can transmit the data to BS.

3.5 Message Complexity

Message complexity of clustering protocol can be assessed as the number of INFO messages, the number of CH_ADV messages, the number of CH_WIN messages and the number of JOIN messages. Since the INFO and CH_ADV messages are exchanged between nodes and neighbours, the number is a multiple of S, the total number of nodes in the network. That is, messages required to initiate election are $O(S * d)$. The multiple d is the average number of neighbours a node has in the network. An estimated value of d, assuming uniform distribution in the region is $\frac{S * range}{\pi R^2}$. Since, in practice $S \approx R^2$, the messages are $O(S * range)$. The number of CH_WIN messages are $O(S)$, assuming very little overlap between regions covered by the CH nodes. The number of JOIN messages is again $O(S)$ as every node selects only one CH to join. Thus, overall message complexity of the clustering protocol is $O(S * range)$. If the value of α is large, no re-election is done and CHs are fixed for α rounds, meaning no communication overhead for election before α rounds. So message overhead of the proposed protocol can be further decreased by setting α to a suitable value larger than 1. In worst case, if election takes place every round, and the network is alive for ρ rounds, total message overhead incurred is $O(\rho * S * range)$.

Table 1 Values of simulation parameters

Parameters	Values
Initial energy of nodes	1 J
E_{elec}	50 nJ/bit
ε_{fs}	10 pJ/bit/m^2
ε_{mp}	0.0013 pJ/bit/m^4
E_{DA}	5 nJ/bit/signal
Packet size	1000 bytes
Range	20 m

4 Experiment Results and Evaluation

4.1 Simulation Set-up

The values of the simulation parameters corresponding to the energy depletion radio model are listed in Table 1. The number of rounds for the proposed protocol before re-election is set to minimum, that is $\alpha = 1$. The experiments are done by varying the area and the number of nodes. The radius of the path of the BS is kept at five different levels: 25, 50, 75, 100 and 125 m. Keeping nodes constant and increasing area means that density is decreased to observe the effect. To increase the density five different numbers of nodes in experiments are 100, 150, 200, 250 and 300 at constant radius of 50 m.

The protocols simulated for comparison are as follows:

(i) DECA—Distributed energy-efficient clustering algorithm suggested by [15]. The number of advanced nodes is 10% of the total nodes.
(ii) Random Baseline—A cluster head election that randomly elects 10% of alive nodes as CHs.
(iii) Proposed—The proposed cluster head election protocol.

4.2 Evaluation for Network Life

A longer network life is indicated by larger number of rounds a protocol executes. The random baseline has almost double life than DECA, and the proposed protocol has double than the random baseline protocol. The reason for short life of DECA is not nodes dying pre-maturely; rather, the energy of the advanced nodes which are permanently the cluster heads gets consumed very fast. Whenever any CH dies, the load of other CHs increases. Figure 1 shows the comparison of the three protocols for network life when the area of region is increased. Density is decreasing, and the distances between the nodes increase. This makes more energy consumption.

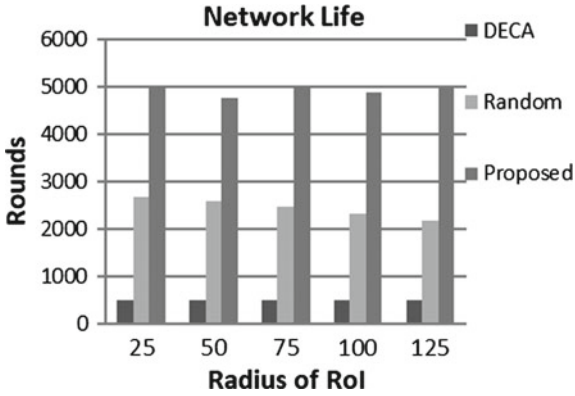


Fig. 1 Comparison of network life at different radius settings

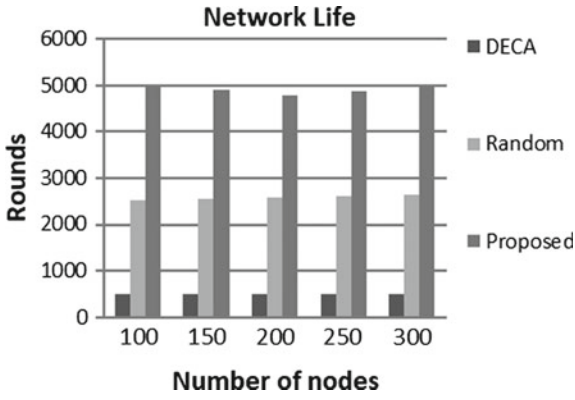


Fig. 2 Comparison of network life at different numbers of nodes

The comparison of the protocols for network life when the number of nodes is increased is shown in Fig. 2. As the number of nodes increases, density increases and distances between nodes are less. But the number of CHs is fixed in random baseline and DECA, so members joining a cluster head increase, consuming more energy. The proposed protocol runs for largest number of rounds as the number of CHs and nodes that become CH is selected according to situations and done in best possible way.

4.3 Evaluation for FND and HND

The number of rounds executed before the first node dies is indicated as FND and for 50% nodes dead is indicated as HND. The proposed protocol has very low FND as compared to the DECA and random baseline. It implies that the proposal is susceptible to hot-spot problem. But gradually the energy load is distributed and the value of HND is very high. A high value of HND means that a protocol is able

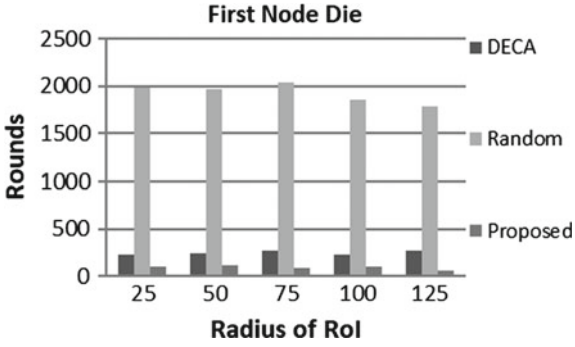


Fig. 3 Comparison of FND at different radius settings

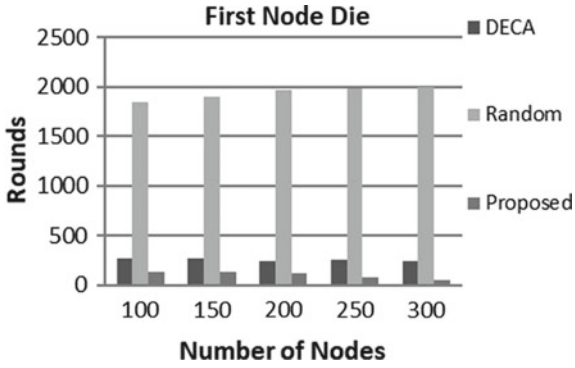


Fig. 4 Comparison of FND at different numbers of nodes

to distribute load in the network among all nodes and energy consumption is more uniform. Figures 3 and 4 show comparison of FND values for increasing area and increasing nodes, respectively, for the three protocols. The random baseline has a very high value for FND. The comparison of HND could be done only between random baseline and the proposed protocols. For DECA, the value of HND cannot be observed as the protocol stops when all advanced nodes have died. Figure 5 shows the comparison when the radius of the region is increased. Figure 6 shows the comparison when the number of nodes is increased. The value of HND for the proposed CH election is very high as compared to the random appointment of CHs.

4.4 Evaluation for Energy Consumption

Though uniform load distribution and energy consumption are indicated by values of FND and HND, the energy of entire network that gets consumed in one round of protocol should be measured. A low value is preferable by practitioners. It can be seen in Figs. 7 and 8 that the energy consumption of proposed protocol is very low. This is because the CHs are elected wisely according to energy and distance.

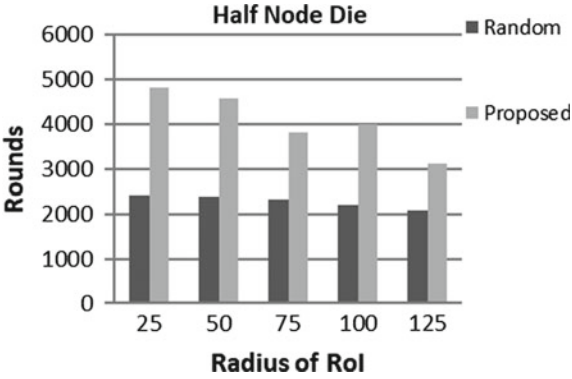


Fig. 5 Comparison of HND at different radius settings

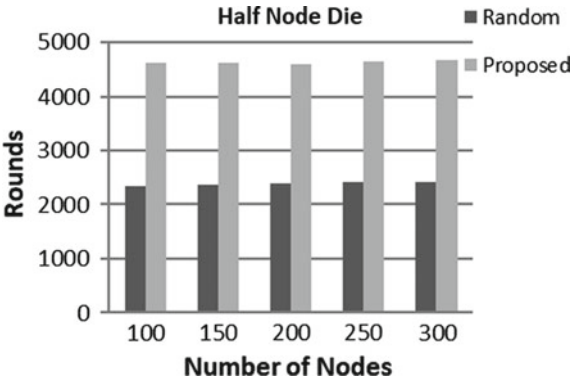


Fig. 6 Comparison of HND at different numbers of nodes

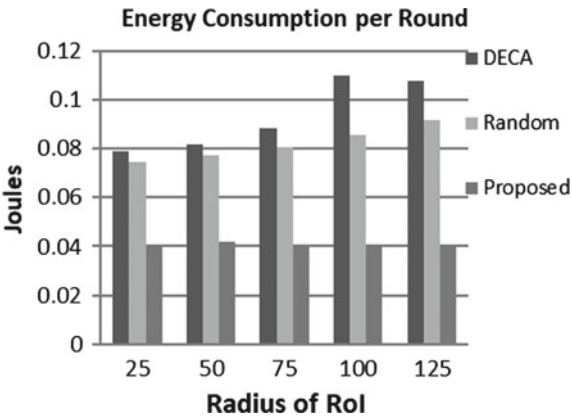


Fig. 7 Comparison of energy consumption per round at different radius settings

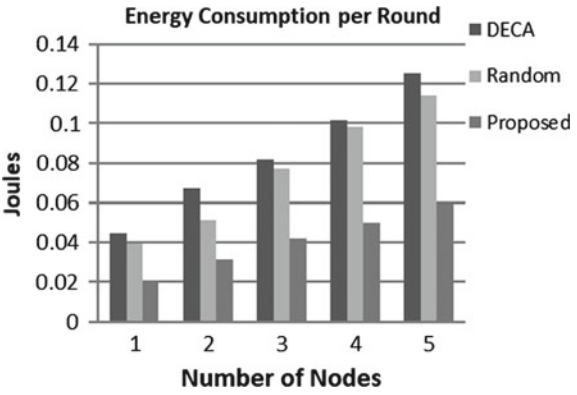


Fig. 8 Comparison of energy consumption per round at different numbers of nodes

5 Conclusion

There exist very few works in the literature that deal with mobile BS. We assume a BS moving in a circle of fixed radius with uniform velocity. The CHs are elected based on their potential: a quantity computed as a combination of the residual energy with respect to neighbours and distance from BS. For cluster formation decision is taken based on both the distance of node from CH and the distance of that CH from BS. This results in a protocol where the nodes near to BS and farther from BS are elected as CHs alternatively. It has a very desirable effect of uniform load distribution over the nodes and maintains good coverage of the RoI for longer period of time.

The only issue is that the proposed protocol has very low value of FND making it susceptible to hot-spot problem in early stage of operation, but in later stage no hot-spot problem is observed. A very high value of HND indicates this. Rigorous analysis of the proposal may reveal the cause of low value of FND. We leave this an open problem at present.

References

1. W. Dargie, C. Poellabauer, *Fundamentals of Wireless Sensor Networks: Theory and Practice* (Wiley 2010), pp. 168–183
2. S. Tarannum, Energy conservation challenges in wireless sensor networks: a comprehensive study. Wirel. Sen. Netw. **2**, 483–491 (2010)
3. X. Liu, A survey on clustering routing protocols in wireless sensor networks. Sensors **12**, 11113–11153 (2012)
4. S. Kaur, R.N. Mir, Clustering in wireless sensor networks—surevy. Int. J. Comput. Netw. Inf. Secur. **6**, 38–51 (2016)
5. W.R. Heinzelman, A. Chandrakasan, H. Balakrishnan, Energy-efficient communication protocol for wireless microsensor networks, in *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences* (2000), pp. 10–19

6. M. Ye, C.F. Li, G.H. Chen, J. Wu, EECS: an energy efficient clustering scheme in wireless sensor networks, in *Proceedings of the 24th IEEE International Performance, Computing, and Communication Conference (IPCCC 2005)* (2005), pp. 535–540
7. Z. Xu, Y. Yin, J. Wang, An density-based energy-efficient routing algorithm in wireless sensor networks using game theory. Int. J. Future Gener. Commun. Netw. **5**, 99–112 (2012)
8. C.F. Li, M. Ye, G.H. Chen, J. Wu, An energy-efficient unequal clustering mechanism for wireless sensor network, in *Proceedings of the IEEE International Conference on Mobile Adhoc and Sensor Systems Conference* (2005), pp. 596–640
9. W.A. Ellatief, O. Younes, H. Ahmed, M. Hadhoud, Energy efficient density-based clustering technique for wireless sensor network, in *Proceedings of the 8th International Conference on Knowledge and Smart Technology (KST)* (2016)
10. T. Banerjee, B. Xie, J.H. Jun, D.P. Agrawal, Increasing lifetime of wireless sensor networks using controllable mobile cluster heads. Wirel. Commun. Mobile Comput. **10**(3), 313–336 (2010)
11. I.I. Er, W.K.G. Seah, Mobility based d-hop clustering algorithm for mobile and ad-hoc networks, in *Proceedings of the IEEE Wireless Communications and Networking Conference*, pp. 2359–2364, March 2004
12. M.R Jafri, N. Javaid, A. Javaid, Z.A. Khan, *Maximizing the Lifetime of Multi-Chain Pegasis Using Sink Mobility* (2013), arXiv:1303.4347
13. S. Feng, B. Qi, L. Tang, An improved energy-efficient PEGASIS-based protocol in wireless sensor networks, in *2011 Eighth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)* (Shanghai 2011), pp. 2230–2233, https://doi.org/10.1109/fskd.2011.6020058
14. D. Amine, B. Nasr-Eddine, L. Abdelhamid, A distributed and safe weighted clustering algorithm for mobile wireless sensor networks. Procedia Comput. Sci. **52**, 641–646 (2015)
15. N. Mazumdar, H. Om, *Distributed Energy Efficient Clustering Algorithm for Mobile Sink Based Wireless Sensor Networks* (2016)

A Novel Algorithm to Improve Quality of Service of Cell Edge Users in LTE

Himani Lodwal, Anjulata Yadav and Manish Panchal

Abstract Long-Term Evolution (LTE) system represents an important milestone in the context of cellular networks. The evolved networks are projected to provide enormous data rate. The cell edge users suffer from very low quality of service (QoS) due to a very large distance from base station called eNodeB (eNB). The paper presents an algorithm named as improved extended modified largest weighted delay first (P- EMWDF) to improve the QoS for cell edge users that provide improvement in the system throughput. This technique provides better network goals. The paper compares improved EMLWDF with extended modified largest weighted delay first (EMLWD) and modified largest weighted delay first (MLWDF), and the results show that improved EMWDF provides better performance than MLWDF and EMLWDF.

Keywords CDMA · EDGE · EPC · EPS · E-UTRAN · GSM-EDGE · LTE UMTS-HSPA

1 Introduction

LTE is an evolved technology for faster and high-speed wireless communication for the data terminals. This technology is based on the universal mobile telecommunication system–high-speed packet access (UMTS-HSPA) and the well-known global system for mobiles-enhanced data for global evolution (GSM-EDGE) technologies. It also increases the speed and capacity of a mobile network using a different type

H. Lodwal (✉) · A. Yadav · M. Panchal
Department of Electronics and Telecommunication Engineering,
Shri G. S. Institute of Technology and Science, 23, Sir M. Visvesvaraya Marg,
Indore 452003, Madhya Pradesh, India
e-mail: himanilodwal06@gmail.com

A. Yadav
e-mail: yadawanjulata@gmail.com

M. Panchal
e-mail: hellopanchal@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_26

of radio interface along with the most important core network improvements. Three GPP developed the improved LTE in its Release-8 series, and some minor improvement in quality is done in Release-9. Different LTE frequencies are used in different countries. Only, multi-band cell phones are able to use LTE in the countries where it is supported. It is not compatible with 2G or 3G networks. A separate radio spectrum is required for LTE [1]. For downlink transmission, LTE employs orthogonal frequency division multiple access (OFDMA), and for the uplink transmission, it uses single carrier frequency division multiple access (SCFDMA). LTE downlink supports 100 Mbps, and LTE uplink supports 50 Mbps.

2 System Architecture of LTE

The principle behind LTE architecture is the discrete functional decomposition. The whole system can be decomposed in terms of the functional entities. Three GPP has defined evolved packet core (EPC) network architecture to support the Evolved Universal Terrestrial Radio Access Network (E-UTRAN) through a decline in the number of network components [2]. E-UTRAN contains user equipment (UE) and evolved node B (eNB). The Internet service provider experiences a flawless mobility with LTE architecture. It provides simpler functionality and high redundancy.

Figure 1 shows the LTE system architecture. This is a logical depiction of the network architecture. The network reference model identifies the functional components in the architecture and the reference points between the functional components. The access network stands for the Evolved Universal Terrestrial Radio Access Network (E-UTRAN) [3]. The core network in the architecture is called evolved packet core (EPC). In LTE, the length of a frame is 10 ms. This 10 ms frame is divided into ten sub-frames. Each sub-frame is composed of two time slots, i.e., 5 ms each; therefore, each sub-frame spans over 1 ms which is known as transmission time interval (TTI), corresponding to 6 or 7 OFDM symbols with normal or short cyclic prefix in default configuration [4]. Figure 2 illustrates the LTE radio frame structure. In time domain, 1 TTI has two time slots to carry 14 OFDM symbols, where 3 OFDM symbols are used for control message signaling. In the frequency domain, the total bandwidth is fragmented into 180 kHz sub-channels [5]. One physical resource block is the smallest radio resource unit which can be assigned to UEs for data transmission [2].

LTE deploys OFDMA for downlink and SCFDMA for uplink as radio access technologies. OFDMA is a form of a signal modulation, i.e., multiple access techniques for orthogonal frequency division multiplexing (OFDM) scheme. It fragments high data rate modulating stream into many slowly modulated narrowband closely spaced sub-carriers. OFDMA is viewed as slowly modulated narrowband signal rather one rapidly modulated wideband signal. OFDM symbols are combined into physical

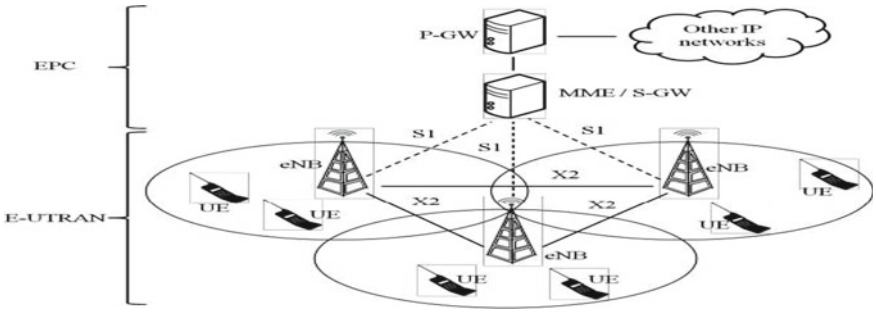


Fig. 1 LTE/LTE-A system architecture [8]

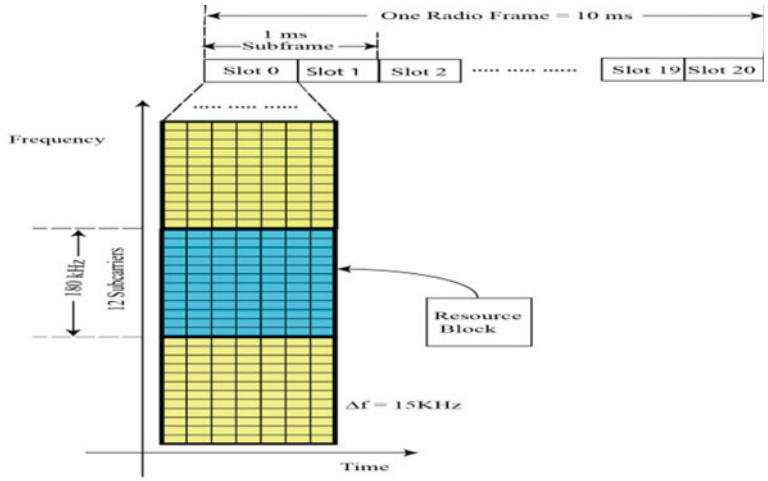


Fig. 2 Time–frequency radio resource grid [1]

resource blocks (PRBs). Each PRB is composed of 12 sub-carriers and has a total size of 180 kHz in the frequency domain and 0.5 ms in time domain. Sub-carrier spacing is 15 kHz; this gives a symbol rate of 66.7 μs [5]. Power consumption is a key attribute for UE terminals, and for this there is a need to adopt a transmission scheme which would be compatible with the requirements of LTE in terms of transmission power efficiency. Uplink means data flows from UE to eNB LTE network. For uplink transmission, the use of OFDMA is not ideal because of its high peak-to-average power ratio (PAPR) and hence SCFDMA is adopted due to its low PAPR [4].

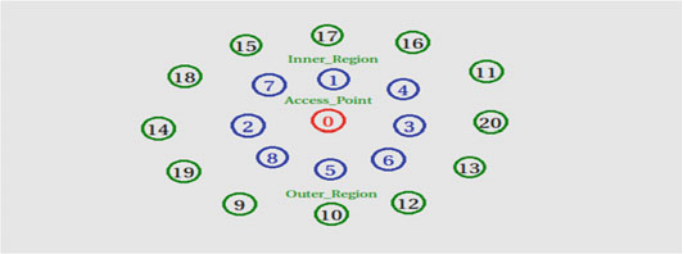


Fig. 3 Simulation scenario

3 Packet Scheduler

Packet scheduling is a key governing function by which decision is taken to achieve QoS. Scheduling refers to the decision process by which the allocation of a portion of the available spectrum is performed by using specific policies [6]. It is not possible for every packet to reach the destination successfully. Therefore, scheduler selects certain data packets on the basis of various algorithms. The packet scheduler checks and decides the priority of the users to be served. Figure 3 shows the packet scheduler. The procedure of packet scheduling is very simple. First, the reference signal is decoded by user equipment (UE). UE determines the CQI and sends or feedbacks to eNB. Channel quality information (CQI) is utilized by eNB for making the allocation decisions. The best modulation and coding scheme (MCS) is selected by adaptive modulation and coding (AMC) unit. The information or data of user, selected MCS and RB are transmitted to UE through physical downlink control channel (PDCCH). The PDCCH payload is read by each UE [7].

4 System Model of Proposed EMLWDF

Figure 3 shows the implementation scenario. The scenario has been assumed to be divided into two regions. First region is the inner region, and the second region is the outer region. Two separate queues are defined for real and non-real-time data. The utility function is calculated for each node according to Eq. (2). The distance is calculated between UE and eNB. If the distance is less than the radius of inner circle, the weight is assumed to be 1. If the distance is greater than the radius of inner circle, the weight is calculated according to Eq. (3). The metric is calculated for each user, and the user having the highest metric value will be scheduled first (Fig. 4 and Table 1).

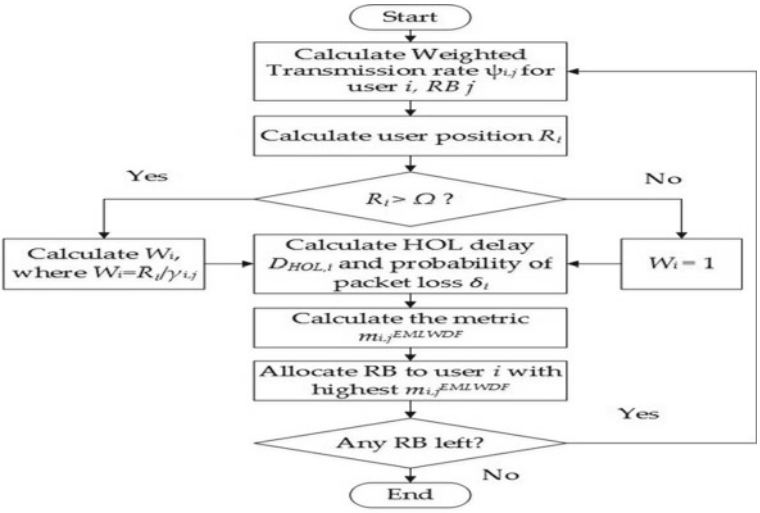


Fig. 4 Flowchart for improved EMLWDF

Table 1 Notations used

Symbols	Definitions
i	User
j	RB
t	Time (TTI index)
ri	Instantaneous transmission rate of user
$D_{(HOL),i}$	HOL delay of user i, RB j
δ_i	Probability of packet loss of user i
τ_i	Delay threshold of user i
Ω	Radius of the inner region
Ri	The distance from the eNB to user i
Si	Probability from failure rate to success rate of user i
εi	The error probability of user i
$\gamma_{i,j}$	Received SINR of user i, RB j
$G_{i,j}$	Channel gain from serving eNB of user i, RB j
$P_{i,j}$	Transmit power of serving eNB for user i, RB j
I	Inter-cell interference

The metric for the proposed method can be expressed as:

$$\mathrm{Mi,\,j(EMLWDF)} = \frac{-\log(\delta \mathrm{i}/\mathrm{w_i})}{\tau_\mathrm{i}} \cdot \mathrm{D(HOL),_i} \cdot \Psi_{\mathrm{i,j}} \tag{1}$$

where utility function $\Psi_{\mathrm{i,j}}$ is defined as:

$$\Psi_{\mathrm{i,j}} = \frac{\mathrm{r_i(t)}}{\mathrm{r_i(t-1)}} \tag{2}$$

Weight is calculated as

$$\begin{array}{ll} \mathrm{w_i} = \mathrm{Ri}/\gamma \mathrm{i,\,j;} & \mathrm{R_i} > \Omega \mathrm{(outer\ region)} \\ 1; & \mathrm{R_i} \leq \Omega \mathrm{(inner\ region)} \end{array} \tag{3}$$

$$\mathrm{R_i} = \sqrt{\sum_{\mathrm{k=1}}^{\mathrm{n}} \left|\mathrm{x_k} - \mathrm{y_k}\right|^2} \tag{4}$$

$$\gamma_{\mathrm{i,j}} = \mathrm{Gi,\,j\,Pi,\,j}/(\sigma^2 + \mathrm{I}) \tag{5}$$

$$\delta_i = \varepsilon_i (1 - S_i)^{\tau i}. \tag{6}$$

5 Results and Discussion

The network simulator NS2 (version 2.35) is used for simulation of the proposed method. Table 2 gives the simulation parameters. At a different number of nodes, the throughput of improved EMLWDF has been observed. The user outside the inner region that is the users in the outer region suffers very poor QoS. The paper compares improved EMLWDF with MLWDF and EMLWDF. EMLWDF had same queue for all the data, but in improved EMLWDF method two separate queues are defined for the real-time data and non-real-time data. Figure 5 compares throughput of the proposed method with MLWDF and EMLWDF. Table 3 shows that throughput of the proposed method is higher as compared to EMLWDF and MLWDF as the method considers the factor of distance between user and eNB, while MLWDF and EMLWDF do not take this factor in consideration. The MLWDF and EMLWDF methods provide good QoS for the users inside the inner region, but as they do not consider the distance factor, methods provide poor QoS for the users outside the inner region, reducing average throughput of the system. Proposed method reduces delay and congestion, reducing packet drop which leads to improved average throughput of the system. Figure 6 compares the packet loss ratio (PLR) of the proposed EMLWDF (P-EMLWDF) with other two methods. Similarly from Table 4, we can conclude that PLR of the proposed method is observed to be very less as compared to MLWDF and EMWDF because of reduced packet delay and higher throughput.

Table 2 Simulation parameters with corresponding values

Parameters	Values
System bandwidth	20 MHz
Operating frequency	900 MHz
Scenario	Urban scenario
Number of users	20, 25, 30, 35, 40, 45, 50
User speed	5 Kmph
Simulation time	1000 TTI

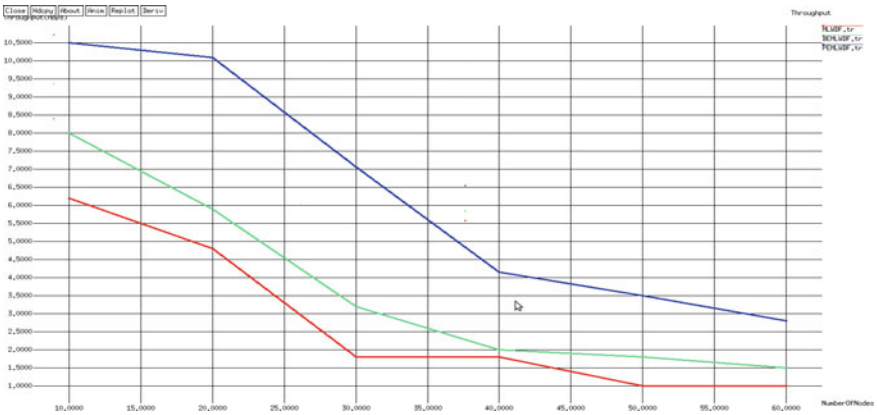


Fig. 5 Throughput versus number of nodes. X-axis—number of users; Y-axis—throughput in Mbps

Table 3 Percentage increment in throughput of the proposed method

Number of nodes	15	35	55
Throughput of the proposed method	10.25	5.40	3.30
Throughput of the EMLWDF (Mbps)	6.90	2.70	1.60
Throughput of the MLWDF (Mbps)	5.50	1.80	0.10
Percentage increment in throughput w.r.t. EMLWDF (%)	32.68	50.00	51.51
Percentage increment in throughput w.r.t. MLWDF (%)	46.34	66.66	96.90

6 Conclusion

LTE is building block for any technology, and management is a primary requirement in LTE. Schedulers are the management body in any network. Scheduling plays a vital role in taking significant decisions in networking. Scheduling leads to a network to be soft and smooth. Hence, schedulers play an important and vital role in a network

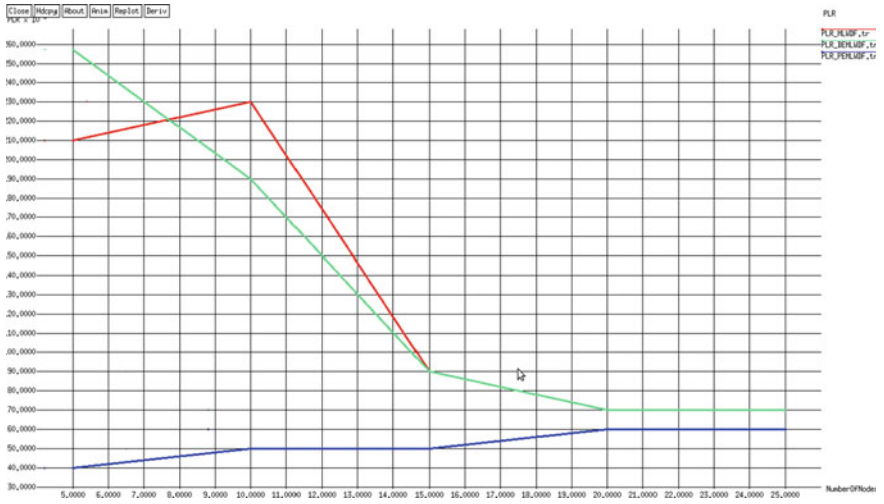


Fig. 6 Packet loss rate versus number of nodes. X-axis—number of users; Y-axis—packet loss rate

Table 4 PLR of the proposed method, EMLWDF and MLWDF

Number of nodes	10	14	16	20
PLR of the proposed method	50	50	55	60
PLR of the EMLWDF	90	110	85	70
PLR of the MLWDF	130	120	85	70

as a regulating entity, taking care of QoS, throughput, delay, bandwidth, efficiency of the network. The proposed scheduler named improved EMLWDF gives a better performance in terms of QoS for cell edge users which further lead to improvement in system throughput. The limitation of the proposed method is, when the number of users increase the system throughput reduces. The method can be extended to 5G. The method can be further extended with MIMO technology also.

Acknowledgements MHRD, India (NPIU), has supported this research activity through the TEQIP Phase-II Project; the authors are highly obliged and wish to thank SGSITS, Indore, (M.P.).

References

1. H.R. Chayon, K.B. Dimyati, H. Ramiah, A.W. Reza, Enhanced quality of service of cell-edge user by extending modified largest weighted delay first algorithm in LTE networks. MDPI, Basel Switzerland **1**(1), 1–14 (2017)
2. D. Nguyen, H. Ngyuyen, E. Renault, A new Channel- and QoS-Aware Scheduling Scheme for Real-time Services in LTE Network. Int. J. Appl. Inf. Syst. **11**(4), 1–8 (2016)

3. P. Chand, R. Mahapatra, R. Prakash, Energy efficient radio resource management for heterogeneous wireless network using CoMP. Wirel. Netw. **22**, 1093–1106 (2016)
4. A. Nagate, D. Ogata, T. Fujii, Cell edge throughput improvement by base station cooperative transmission control with reference signal interference canceller in LTE System, in *75th IEEE Vehicular Technology Conference (VTC Spring)* (Yokohama, Japan 2012), pp. 1–5
5. R. Basukala, H.M. Ramli, K. Sandrasegaran, Performance analysis of EXP/PF and M-LWDF in downlink 3GPP LTE system, in *Proceedings of the First Asian Himalayas International Conference on Internet (AH-ICI)* (Kathmandu 2009), pp. 1–5
6. D. Singh, Performance Analysis of QOS-aware Resource Scheduling Strategies in LTE Femtocell Networks. Int. J. Eng. Trends Technol **1**, 2994–2999 (2013)
7. Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN). Overall Description: Stage 2: TS 36.300, 3rd Generation Partnership Project (3GPP). Valbonne, France (2012)
8. F. Capozzi, G. Piro, L.A. Grieco, G. Boggia, P. Camarda, Downlink packet scheduling in LTE cellular networks: Key design issues and a survey. IEEE Commun. Surv. Tutor. **15**(2), 678–700 (2013)

Performance Comparison of Transmission Control Protocol Variants in WiMAX Network with Bandwidth Asymmetry

Kailash Chandra Bandhu

Abstract Worldwide Interoperability for Microwave Access is known as WiMAX technology which provides high-speed Internet to urban and rural areas. This work focused on the performance comparison of Transmission Control Protocol variants such as New Reno, Sack1, Linux and Vegas in WiMAX network with medium access control and operating parameters. The research carried out determined the optimal values for different parameters to get better performance for different Transmission Control Protocol variants. The performance is evaluated using the data sending rate, data receiving rate and packets dropped which is used to calculate the packet error rate.

Keywords Worldwide Interoperability for Microwave Access (WiMAX) technology · Transmission Control Protocol (TCP) · Physical layer · Medium access control layer · Throughput · Goodput · Packets dropped · Packet error rate Downlink · Uplink and orthogonal frequency division multiplexing (OFDM)

1 Introduction

WiMAX technology is wireless broadband technology that uses the concept of orthogonal frequency division multiplexing and provides high-throughput broadband connection at long distances, and it is based on IEEE 802.16 [1, 2]. The WiMAX network is able to provide faster data transfer, more intelligent and seamless communication systems, adaptive modulation and coding schemes, flexible channel spectrum, flexible data rate access and cyclic prefix [2].

Network asymmetry means the characteristics of the network in one direction do not match the other direction [3]. Channel bandwidth and cyclic prefix affect the TCP performance with network inequality, as TCP relies on timely arrival of the acknowledgement to increase the rate of data sending and congestion window. In

K. C. Bandhu (✉)
Symbiosis University of Applied Sciences, Indore, Madhya Pradesh, India
e-mail: kailash_bandhu@yahoo.co.in

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_27

normal network conditions, an acknowledgement is duly received for the packet sent, and it helps in increasing the rate of data sending of sender. When the congestion occurs in network, usually indicated by packet loss, Transmission Control Protocol suddenly reduces its congestion window and retransmits the lost packets [4, 5], and retransmission can increase the congestion. Generally, there are two ways to indicate packet loss or congestion: (1) expiry of transmission timer and (2) the receipt of three or more duplicate acknowledgements.

In the presence of an imperfect acknowledge channel, the acknowledge timer is interrupted; i.e., sent packets are not duly acknowledged. Consequently, on the sender the timer ends, in which the Transmission Control Protocol interprets as congestion, the congestion window abruptly reduced and packets are retransmitted, even if these packets have reached to the receiver correctly. This means that the throughput and goodput of Transmission Control Protocol not only depend on the characteristics of the data sending channel, but also depend on the reverse channel used by acknowledgement [3, 6, 7].

New Reno: New Reno [8] maintains two variables, the size of the congestion window, which is initially set to 1 segment, and the SS threshold (SSThresh). At the beginning of the TCP connection, the sender enters into the slow start (SS) phase, in which it increases the congestion window to 1 segment for each received acknowledgement. Whenever the congestion window reaches to the SS threshold (SSThresh), the sender enters into congestion avoidance phase, in which it increases the congestion window by 1 / congestion window for each received acknowledgement to slowly probe the available network bandwidth. This linear expansion ends when the congestion window reaches to the receiver's advertised window or by the reception of three duplicate acknowledgements. In TCP, the packet was lost due to the congestion of links, and this network reduced the congestion window to 1/2 of its present value in an attempt to stop the network collapse (fast recovery). In Transmission Control Protocol, the reactive congestion control, based on additive-increase/multiplicative decrease and avoidance mechanism, was not able to efficiently manage the mixed type packets loss in heterogeneous wireless network [9].

Sack1: Sack is an expansion of Reno with 'select acknowledge' and works around problems faced by Reno and New Reno, i.e., multiple lost packets detection and retransmission of more than one lost packet in every RTT [10]. Sack retains the slow start of Reno and fast retransition. In Sack, there is no provision of cumulative acknowledgement but must have selective acknowledged concept. Thus, each acknowledgement has a block which tells which segments are being acknowledged. Thus, the sender knows that which segments have been acknowledged and which are still outstanding [10]. Initialize variable pipe which estimates how much data outstating in the network and also sets the congestion window half of the current size whenever the sender enters into fast recovery. Every time it receives an acknowledgement, it reduces pipes by 1, and every time it retransmits a segment, again it increases to 1 [10]. Whenever the pipe goes smaller than the congestion window, it checks which segments are not received and sends them. If such segment is not outstanding, then it sends a new packet [10]. Thus, more than one lost segment can be sent in an RTT.

The major problem with Sack is that currently the receiver does not provide selective acknowledgement feature, and there is a need to implement selective acknowledgement at receiver, which is not very easy [10].

Linux: Ns-2 Transmission Control Protocol Linux implementation has embedded source code of Transmission Control Protocol Congestion Control module in Linux kernel. If compared to the existing Ns-2 Transmission Control Protocol implementation, there are three improvements in the Ns-2 Transmission Control Protocol Linux:

(1) A standard interface for its similar congestion control algorithms in Linux 2.6, ensuring better expansion for emerging congestion control algorithms.
(2) A redesigned loss detection module (i.e., scoreboard) which is more accurate.
(3) A new event queue scheduler which enhances the simulation speed.

As a result, Ns-2 Transmission Control Protocol Linux is more extensible, runs faster and generates simulation results, which are very close to Linux's actual Transmission Control Protocol behaviour. In addition to helping the network research community, the Ns-2 Transmission Control Protocol Linux will also help Linux kernel community to debug and test their new congestion control algorithm [11].

Vegas: Vegas [12] abandons Reno linear window growth and deploys a separate congestion avoidance mechanism, tries to assess the level of congestion of the network before happening and avoids it. Its decision is based on throughput measurement per round trip time (RTT), which reflects the status of the network. Therefore, the Vegas calculates the difference between the real rate (per packet per RTT) and the expected rate (per packet sent by the best RTT) on the sender. If the difference is less than threshold value, it indicates that network resources are underutilized, and it increases its window by one segment. If the difference is greater than the threshold value, it indicates the network is experiencing a congestion, and it reduces its window size by one segment to prevent the congestion. Otherwise, it keeps the congestion window same, because this network utilizes the best resources. The advantage of Vegas is that it has the ability to use more efficient bandwidth, which is the ability to avoid the congestion and unnecessary retransmission of dropped packets due to linear growth of Reno. However, it still considers all packet loss as a sign of the congestion, thus reducing unnecessary windows in the wireless network.

2 Topology

Figure 1 represents the network topology which is used to evaluate the performance of different Transmission Control Protocol Variants in which all subscriber stations (SSs) are downloading stations, and individual Transmission Control Protocol connections are established between fixed station and subscriber stations via base station. Fixed node is connected to base station through physical communication medium of 100 Mbps capacity and 10 ms delay.

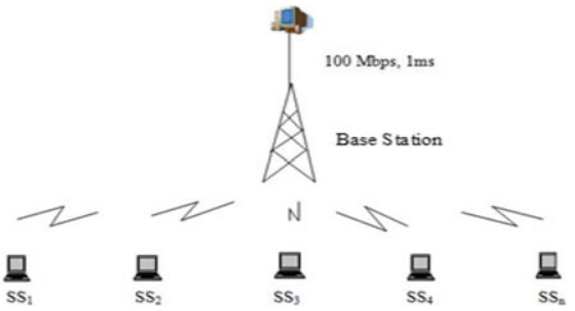


Fig. 1 Network topology

3 Performance Metrics

The performance of Transmission Control Protocol variants is studied using three metrics:

3.1 Throughput

The amount of raw data (bytes) sent by a sender without overhead.

$$Througput = \frac{No.of\ Pkts\ send * (Pkts\ Size - Overhead) * 8 * 0.0000006}{Simulation\ Duration}\ \text{Mbps}$$

where TCP overhead = 80 bytes.

3.2 Goodput

The amount of data (bytes) received from sender at receiver end without overhead and successfully acknowledged.

$$Goodput = \frac{No.of\ Pkts\ Recd * (Pkts\ Size - Overhead) * 8 * 0.0000006}{Simulation\ Duration}\ \text{Mbps}$$

where TCP overhead = 80 bytes.

3.3 Packets Dropped

The number of packets dropped to calculate the packet error rate.

Table 1 Parameters

WiMAX technology parameters	
Channel bandwidth	3, 7, 10, 20, 28 MHz
Frame duration	5 ms
Modulation and coding schemes	64QAM ¾
Cyclic prefix	1/2, 1/4, 1/8, 1/16
Contention size	5
Propagation model	Two Ray Ground
Transmission Control Protocol and other parameters	
Variants of TCP	New Reno, Sack1, Linux and Vegas
Segment size of TCP	1040 bytes
Downloading subscriber stations	5, 10, 15, 20, 25
Acknowledgement delayed	2 s
Start time of TCP	20 s
Duration of simulation	300 s

4 Simulation Parameters

The Ns-2 (Ns2.31) with WiMAX module is used for the performance comparison of Transmission Control Protocol variants, and it was developed by National Institute of Standards and Technology (NIST) [13, 14].

Table 1 shows the parameters for simulation.

5 Results

This section presents the performance comparison of Transmission Control Protocol variants such as New Reno, Sack1, Linux and Vegas. The performance of Transmission Control Protocol variants is evaluated by varying offered load and medium access control layer parameters, and comparison is done on the basis of throughput, goodput and packets dropped.

5.1 Effect of Downloading Subscribers

This section presents the performance comparison of Transmission Control Protocol variants by varying number of downloading wireless subscriber stations for different channel bandwidths with constant 0.75 DL:UL ratio. The performance comparison is done by using throughput, goodput and packets dropped.

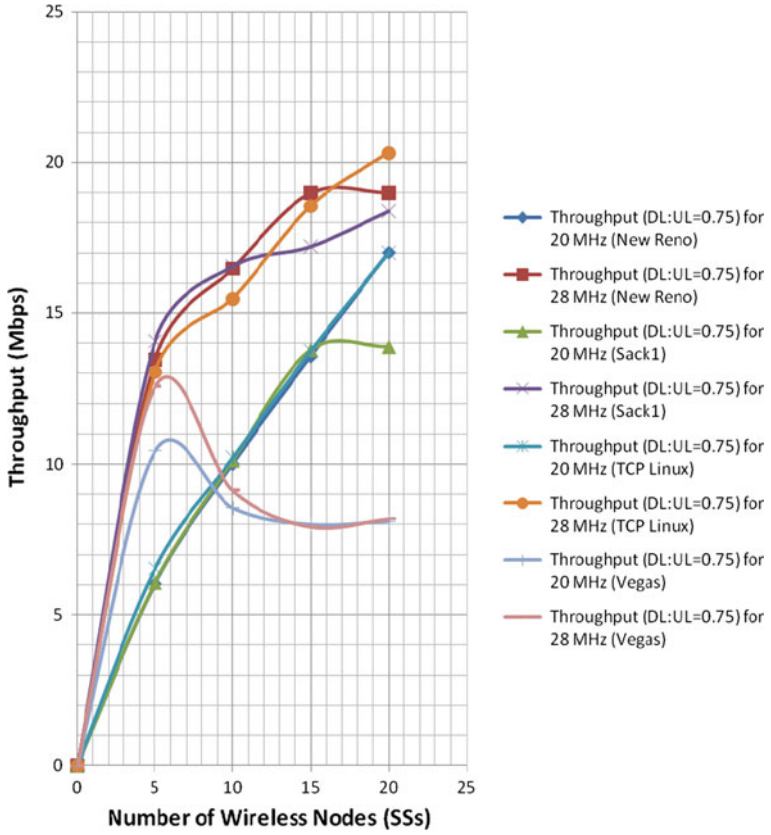


Fig. 2 Performance comparison of transmission control protocol variants using throughput for different numbers of subscriber stations (SSs) with constant DL:UL ratio

It is observed in Fig. 2 that the throughput increases when the downloading wireless nodes increase for different channel bandwidths with 0.75 DL:UL ratio for different Transmission Control Protocol variants.

A large number of downloading wireless nodes has high throughput and goodput because larger wireless nodes generate the high traffic.

Maximum throughput obtained for 28 MHz channel bandwidth with 20 wireless nodes (SSs) with 0.75 DL:UL ratio in case of Transmission Control Protocol Linux is shown in Fig. 2.

It means the TCP Linux gives high throughput as compared to New Reno, Sack1 and Vegas due to better designing of expansion and compaction of congestion window. TCP Vegas gives minimum throughput for 28 MHz with 20 wireless nodes (SSs) with 0.75 DL:UL ratio due linear expansion of congestion window.

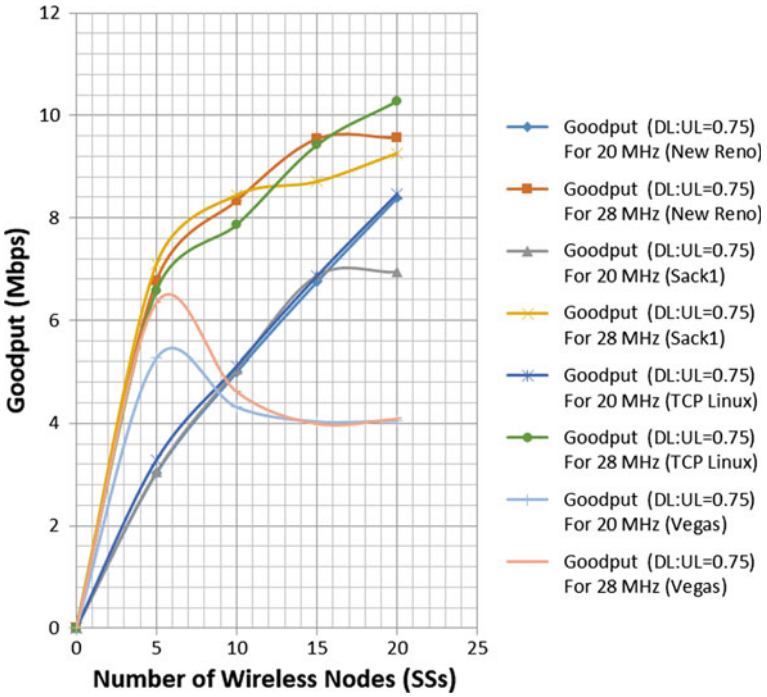


Fig. 3 Performance comparison of transmission control protocol variants using goodput for different numbers of subscriber stations (SSs) with constant DL:UL ratio

Similarly, it is observed in Fig. 3 that the goodput increases when the number of nodes increases for different channel bandwidths with 0.75 DL:UL ratio for different Transmission Control Protocol variants.

Maximum goodput obtained for 28 MHz channel bandwidth with 20 wireless nodes (SSs) with 0.75 DL:UL ratio of Transmission Control Protocol Linux is shown in Fig. 3.

It means the Transmission Control Protocol Linux gives high goodput as compared to New Reno, Sack1 and Vegas due to better designing of expansion and compaction of congestion window.

TCP Vegas gives minimum goodput for 28 MHz with 20 wireless nodes (SSs) with 0.75 DL:UL ratio due to the linear expansion of congestion window.

The goodput is less than the throughput due to loss of packets during wireless transmission. Wireless channel is unreliable communication medium, and this is the reason of packet loss, so the throughput and goodput are not equal.

Figure 4 shows the packets dropped for different numbers of wireless nodes (SSs) with 0.75 DL:UL ratio and different channel bandwidths in New Reno, Sack1, Vegas and Linux.

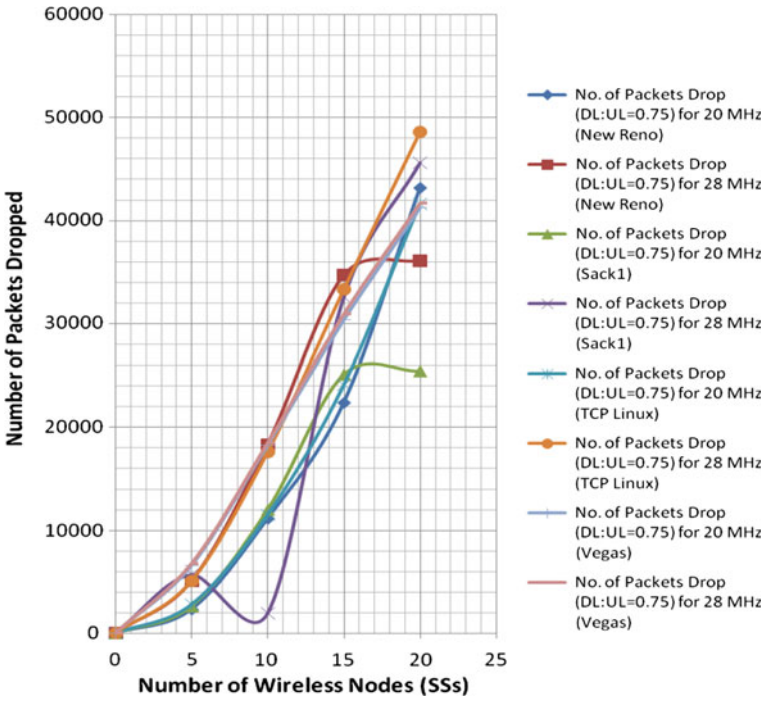


Fig. 4 Performance comparison of transmission control protocol variants using packets dropped for different numbers of subscriber stations (SSs) with constant DL:UL ratio

Figure 4 shows that the packets dropped increases when the downloading wireless nodes (SSs) increase for different channel bandwidths for different Transmission Control Protocol variants.

The maximum Packets Dropped was obtained at 28 MHz for Transmission Control Protocol Linux with 20 wireless nodes which is considerably better than TCP new Reno [15]. Similarly the minimum Packets Dropped was obtained for 20 MHz channel bandwidth for Transmission Control Protocol Sack1 with 20 wireless nodes (SSs).

The number of packets dropped for 20 MHz is less than the 28 MHz channel bandwidth.

5.2 Effect of Channel Size

This scenario presents the performance comparison of Transmission Control Protocol variants by varying channel bandwidth and constant offered load with 0.75 DL:UL ratio.

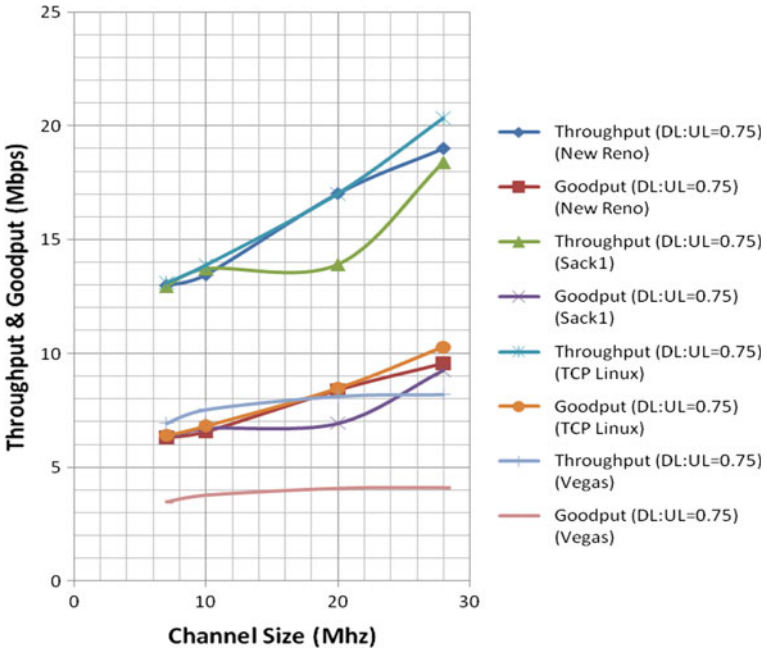


Fig. 5 Performance comparison of transmission control protocol variants using throughput and goodput for different channel sizes with constant subscriber stations (SSs) and DL:UL ratio

Figure 5 represents the impact of channel bandwidth for different Transmission Control Protocol variants with the constant number of wireless nodes and DL:UL ratio. The impact is evaluated using throughput, goodput and packets dropped.

Figure 5 shows that larger channel bandwidth gives high throughput, goodput and low channel bandwidth gives low throughput and goodput for different Transmission Control Protocol variants because large channel bandwidth has capability to carrying high traffic.

Transmission Control Protocol Linux provides high throughput and goodput for 28 MHz channel bandwidth as compared to other Transmission Control Protocol variants and Vegas provides the low throughput and goodput as compared to other Transmission Control Protocol variants which are represented by Fig. 5.

Figures 6 shows that the number of packets with different channel bandwidths has been dropped for various Transmission Control Protocol variants, and it has been observed that a large number of packets have been dropped in high channel bandwidth.

It is also observed that Transmission Control Protocol Linux has a large number of packets dropped as compared to other TCP variants for 28 MHz channel bandwidth because the large bandwidth can carry high traffic, and TCP Vegas has consistent packets dropped for different channel bandwidths.

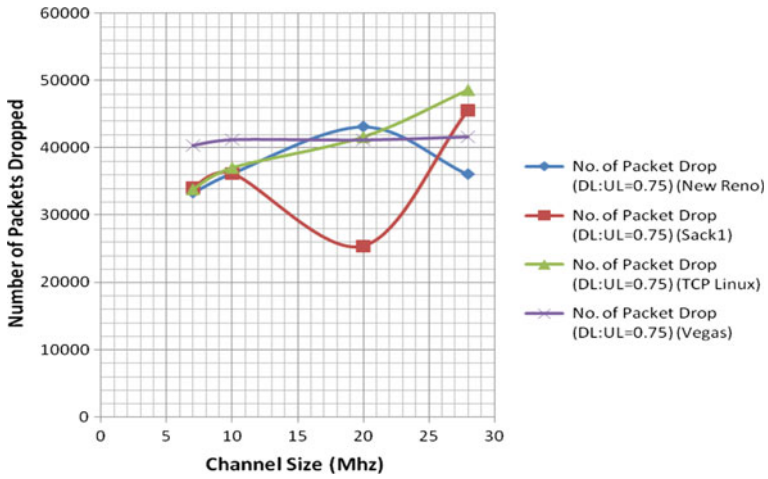


Fig. 6 Performance comparison of transmission control protocol variants using packets dropped for different channel sizes with constant subscriber stations (SSs) and DL:UL ratio

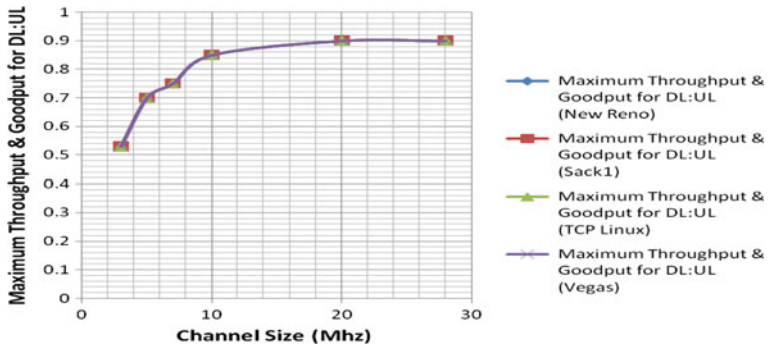


Fig. 7 Performance comparison of transmission control protocol variants using maximum throughput and goodput for DL:UL ratio for different channel sizes with constant subscriber stations (SSs)

Figure 7 shows the maximum supporting DL:UL ratio for different TCP variants with different channel bandwidths.

Figure 7 represents the high channel bandwidth support to high DL:UL ratio because high channel bandwidth has capability to handle high traffic in downloading which is represented by DL:UL ratio and low channel bandwidth support to low DL:UL ratio because the low bandwidth does not carry high traffic in downloading which is expressed by DL:UL ratio.

The highest DL:UL ratio was achieved at 28 MHz channel bandwidth and the lowest DL:UL ratio was achieved at 3 MHz. Whenever channel bandwidth increases, the supporting DL:UL ratio also increases for different Transmission Control Protocol variants.

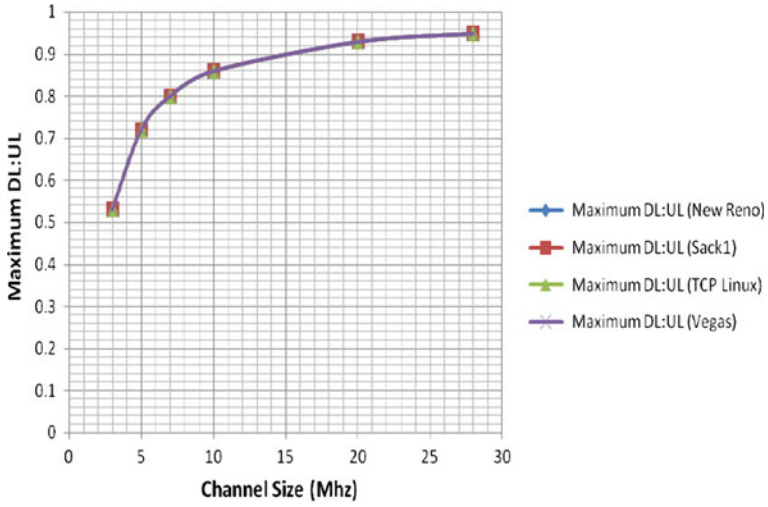


Fig. 8 Performance comparison of transmission control protocol using maximum DL:UL ratio for different channel sizes with constant subscriber stations (SSs)

The Transmission Control Protocol variants such as New Reno, Sack1, Linux and Vegas have same maximum and minimum supporting DL:UL ratio.

The maximum and minimum supporting DL:UL ratios for Transmission Control Protocol variants are only affected by channel bandwidth.

Figures 8 shows the maximum throughput and goodput for DL:UL ratio with different channel bandwidths for various Transmission Control Protocol variants.

It is observed in Fig. 8 that high channel bandwidth support to high DL:UL ratio which gives high throughput and goodput because high DL:UL ratio can handle high traffic in downloading which enhance the throughput and goodput for various Transmission Control Protocol variants. It is also observed in Fig. 8 that the low channel bandwidth support to low DL:UL ratio which reduces the throughput and goodput because low DL:UL ratio cannot handle high traffic in downloading which reduces throughput and goodput for various Transmission Control Protocol variants.

The 28 MHz channel bandwidth provides high throughput and goodput for large DL:UL ratio and 3 MHz channel bandwidth provides low throughput and goodput for low DL:UL ratio for various Transmission Control Protocol variants which are shown in Fig. 8. The Transmission Control Protocol variants such as New Reno, Sack1, Linux and Vegas have same maximum and minimum throughput and goodput for high and low DL:UL ratios, respectively.

Maximum throughput, goodput and minimum throughput, goodput for high and low DL:UL ratios for Transmission Control Protocol variants, respectively, are affected by channel bandwidth and DL:UL ratio.

5.3 *Effect of Cyclic Prefix*

This section presents the performance comparison of Transmission Control Protocol variants by varying cyclic prefix, offered load with 0.90 DL:UL ratio.

Figure 9 represent the impact of cyclic prefix with DL:UL ratio in different Transmission Control Protocol variants, and it is observed that when the cyclic prefix increases, then throughput and goodput decrease with 0.90 DL:UL ratio in various Transmission Control Protocol variants with constant wireless nodes (SSs).

The cyclic prefix 0.03125 has higher throughput and goodput with 0.90 DL:UL ratio in Linux as compared to Sack1, New Reno and Vegas. The cyclic prefix 0.25 has low throughput and goodput with 0.90 DL:UL ratio in Vegas as compared to Linux, Sack1 and New Reno.

When the cyclic prefix increases, the packets dropped decreases for 0.90 DL:UL ratio in various Transmission Control Protocol variants which is shown in Fig. 10.

The cyclic prefix 0.0625 has a larger number of packets dropped with 0.90 DL:UL ratio in New Reno as compared to Linux, Sack1 and Vegas. The cyclic prefix 0.25 has lesser number of packets dropped with 0.90 DL:UL ratio in Vegas as compared to Linux, New Reno and Sack1.

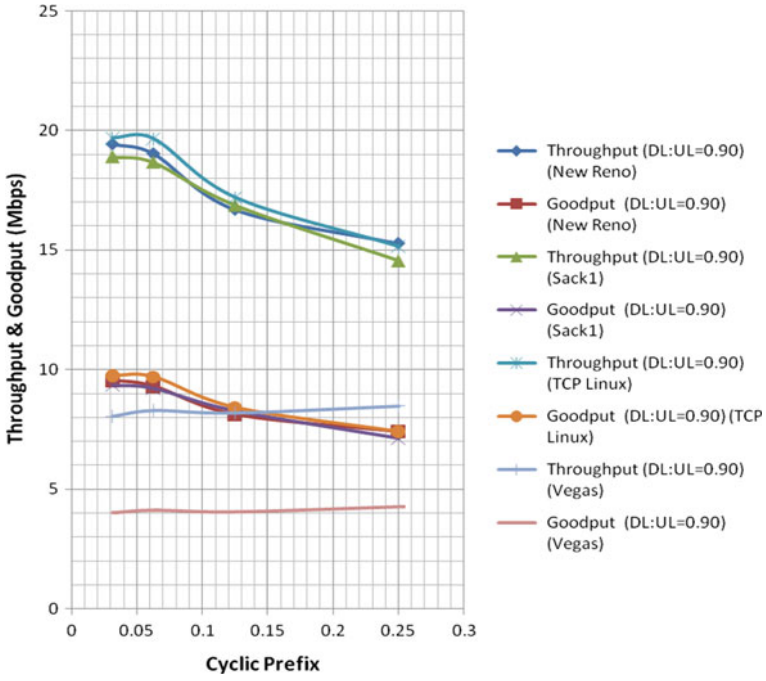


Fig. 9 Performance comparison of transmission control protocol variants using throughput and goodput for different cyclic prefixes with constant subscriber stations (SSs) and DL:UL ratio

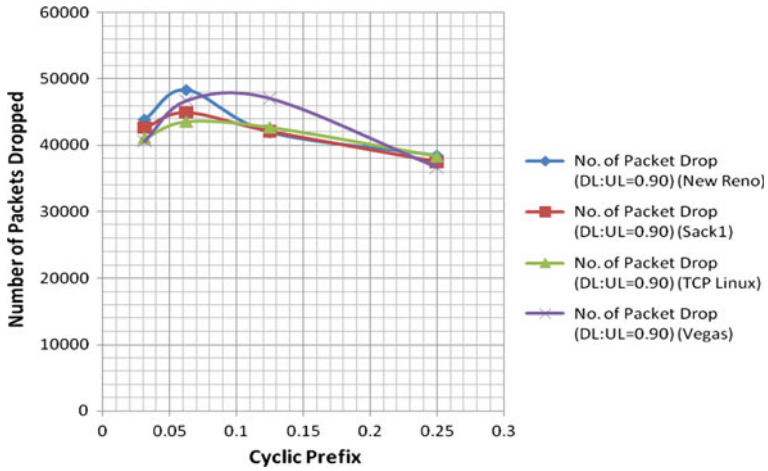


Fig. 10 Performance comparison of transmission control protocol variants using packets dropped for different cyclic prefixes with constant subscriber stations (SSs) and DL:UL ratio

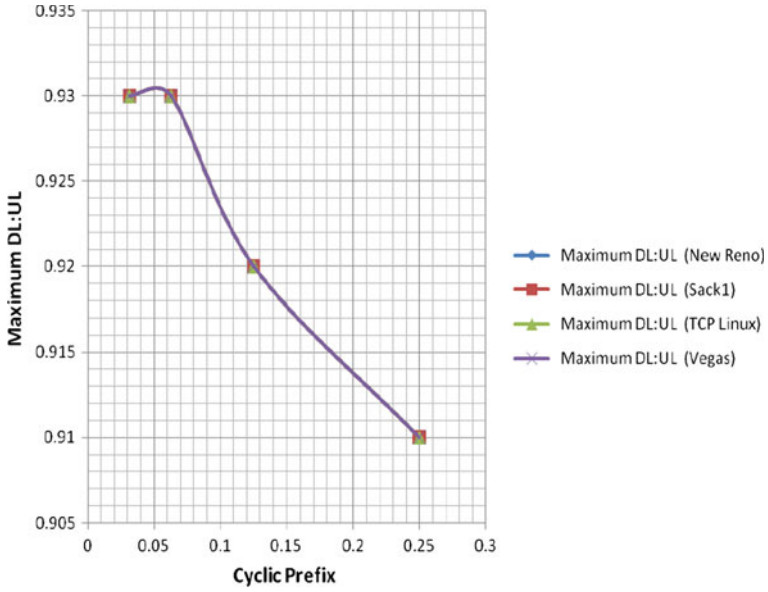


Fig. 11 Performance comparison of transmission control protocol variants using maximum DL:UL ratio for different cyclic prefixes with constant subscriber stations (SSs)

Figure 11 represents maximum DL:UL ratio for different cyclic prefixes in various Transmission Control Protocol variants, and it is observed that when cyclic prefix increases, the maximum supporting DL:UL ratio decreases for various Transmission Control Protocol variants.

The cyclic prefix 0.03125 has high maximum supporting DL:UL ratio, and cyclic prefix 0.25 has low maximum supporting DL:UL ratio in various Transmission Control Protocol variants.

Maximum supporting DL:UL ratio is same for Linux, Sack1, New Reno and Vegas.

6 Conclusion

This work presents the impact of channel bandwidth in WiMAX network with network asymmetry using Transmission Control Protocol variants such as Linux, Sack1, New Reno and Vegas.

The work is done by considering channel bandwidth, number of downloading wireless nodes (SSs), DL:UL ratio in Linux, Sack1, New Reno and Vegas.

In this work, it is observed that whenever the downloading wireless nodes are increased, then throughput and goodput are also increased for different channel bandwidths and different DL:UL ratios in various Transmission Control Protocol variants.

It is also observed that the large number of downloading wireless stations and large DL:UL ratio for different channel bandwidths provide higher throughput and goodput in various TCP variants.

TCP Linux has better throughput and goodput than Sack1, New Reno and Vegas for a large number of downloading wireless nodes.

TCP Linux has high packets dropped than Sack1, New Reno and Vegas for a large number of downloading wireless stations.

Whenever channel bandwidth increases, the throughput and goodput also increase for various DL:UL ratios in various TCP variants.

The higher channel bandwidth and higher DL:UL ratio provide higher throughput and goodput in various Transmission Control Protocol variants.

TCP Linux is better than Sack1, New Reno and Vegas for large channel bandwidth in terms of throughput and goodput.

Whenever channel bandwidth increases, the packets dropped also increases for large DL:UL ratio in various TCP variants.

TCP Linux has high packets dropped than Sack1, New Reno and Vegas for large channel bandwidth.

Large channel bandwidth support to high DL:UL ratio and all TCP variants have same value of DL:UL ratio for different channel bandwidths.

The DL:UL ratio remains same for all TCP variants for a given channel bandwidth.

This work presents the effect of cyclic prefix in WiMAX technology with bandwidth asymmetry using various TCP variants such as Linux, Sack1, New Reno and Vegas.

Simulation study shows that higher cyclic prefix reduces the throughput, goodput and packet drop for various TCP variants. However TCP Linux outperforms than

other TCP variants including TCP Reno [16]. The low cyclic prefix enhanced the throughput, goodput and packets dropped for various TCP variants.

TCP Linux gives high throughput and goodput for low cyclic prefix than Sack1, New Reno and Vegas.

The TCP Linux provides less packets dropped, and New Reno has large packets dropped for low cyclic prefix.

All TCP variants have same value for maximum supporting DL:UL ratio for different cyclic prefixes.

References

1. IEEE 802.16 Working Group.: IEEE standard for local and metropolitan area networks. Part 16: Air interface for fixed broadband wireless access systems. IEEE Std, 802, 16-2004
2. T. Cooklev, Air interface for fixed broadband wireless access systems. Wirel.Commun. Stand. Study IEEE 802.11™, 802.15™, and 802.16™, 225–333 (2011)
3. H. Balakrishnan, V.N. Padmanabhan, TCP performance implications of network path asymmetry (2002)
4. M. Gerla, R. Bagrodia, L. Zhang, K. Tang, L. Wang, TCP over wireless multi-hop protocols: simulation and experiments, in *1999 IEEE International Conference on Communications (ICC'99)*, vol. 2 (IEEE, 1999), pp. 1089–1094
5. M. Gerla, K. Tang, R. Bagrodia, TCP performance in wireless multi-hop networks, in *1999 Proceedings of Second IEEE Workshop on Mobile Computing Systems and Applications (WMCSA'99)* (IEEE, 1999), pp. 41–50
6. K. Tsiknas, G. Stamatelos, Comparative performance evaluation of TCP variants in WiMAX (and WLANs) network configurations. J. Comput. Netw. Commun. (2012)
7. K. Xu, Y. Tian, N. Ansari, Improving TCP performance in integrated wireless communications networks. Comput. Netw. **47**(2), 219–237 (2005)
8. A. Kumar, Modified TCP for time critical applications. Glob. J. Comput. Sci. Technol. **14**(1-E), 11 (2014)
9. F. Lefevre, G. Vivier, Understanding TCP's behavior over wireless links, in *2000 Symposium on Communications and Vehicular Technology* (*SCVT-200)* (IEEE, 2000), pp. 123–130
10. K. Fall, S. Floyd, Simulation-based comparisons of Tahoe, Reno and SACK TCP. ACM SIGCOMM Comput. Commun. Rev. **26**(3), 5–21 (1996)
11. M. Jehan, G. Radhamani, Scalable TCP: better throughput in TCP congestion control algorithms on MANETs. Int. J. Adv. Comput. Sci. Appl. Spec. Issue Wirel. Mob. Netw.
12. L.S. Brakmo, L.L. Peterson, TCP Vegas: end to end congestion avoidance on a global internet. IEEE J. Sel. Areas Commun. **13**(8), 1465–1480 (1995)
13. K. Fall, K. Varadhan, The network simulator (ns-2) (2007), http://www.isi.edu/nsnam/ns
14. R. Rouil, The NIST WiMAX network simulator. NIST Technical Report (2007)
15. K.C. Bandhu, R.G. Vishwakarma, The impact of cyclic prefix modulation coding scheme frame duration two way transfer and propagation model with network asymmetry in WiMAX network using TCP new reno. Int. J. Eng. Res. Technol. (IJERT) **3**(3) (2014)
16. K.C. Bandhu, R.G. Vishwakarma, The impact of channel bandwidth with network asymmetry in WiMAX network using TCP new reno. Traffic **1**(16), 5 (2013)

A Band Jamming Technique with Non-coherent Detection for Wireless Network Security

Sapna Patidar and Ravi Khatri

Abstract Security in wireless networks is a serious challenge due to the access of the channel to adversaries because of the channel being unguided. Conventional encryption mechanisms cannot guarantee security in wireless networks since even the most complex encryption algorithms can be broken. Hence, deliberate jamming is one way to safeguard data from possible attacks. This paper proposes a band jamming technique using fast frequency hopping (FFH) along with non-coherent detection for evading possible attacks. The performance parameters evaluated the bit error rate (BER) and the outage probability of the proposed system. Non-coherent detection has been employed since maintaining coherence for a fast frequency hopping technique is extremely challenging under practical noisy channel conditions. The results show the various stages of the jamming and de-jamming processes.

Keywords Band jamming · Fast frequency hopping (FFH)
Non-coherent detection · Bit error rate · Outage probability

1 Introduction

Band jamming is a technique to deliberately jam the signal bandwidth so as to make the signal imperceptible to possible adversaries [1]. The principal goal of the proposed work is to create and work towards a faster version of the frequency hopping method intended for wireless networks which in turn utilizes the frequency hopping of spread spectrum. This is to ensure the effectiveness of the proposed model to prevent the attacks of adversaries. An algorithm has been suggested based on fast frequency hopping aiming to prevent the probable interception attacks by the intruders. Spread of the signal makes it unrecognizable for the attackers, but it has a con that makes the signal more vulnerable to noise that can lead to excessive bit error rate. So another priority is to also handle the BER and to keep it in lower levels

S. Patidar (✉) · R. Khatri
VITM, Indore, India
e-mail: spatidar57@yahoo.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_28

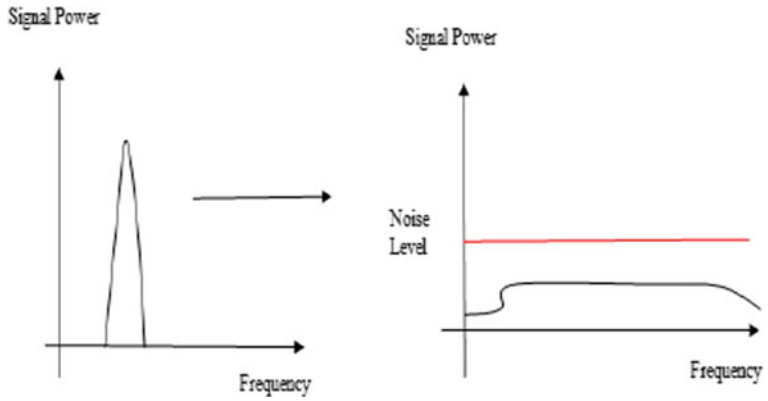


Fig. 1 Concept of spread spectrum

to maintain the quality of service norms. Spreading the signal more though helps in making it more secure, but it weakens the strength of signal. So in this context, an efficient mechanism is proposed comprising of coherent conjugate detection of the signal after de-jamming of signal. It mainly focuses on reducing effects of errors obtained in the recovered signal. The performance is measured in terms of the bit error rate (BER) of the system. The BER of the system varies in accordance with the spreading factor of the jamming process. The concept of spread spectrum can be understood using Fig. 1.

2 Fast Frequency Hopping

Fast frequency hopping is a technique wherein the bandwidth of a given signal is spread out in a larger bandwidth by changing or hopping the carrier frequency continuously [2]. Considering the bit period of a signal to be T_b and the hopping period or the chip period to be T_c

$$\text{If } T_b < T_c \tag{1}$$

Then, the case is called slow frequency hopping (SFH). On the contrary, if the following relation holds true,

$$T_b > T_c \tag{2}$$

Then, the case is called fast frequency hopping (FFH).

The ratio of the bandwidth after spreading to the bandwidth before spreading is called the spreading factor (L) defined mathematically as [3]:

$$L = B_S/B \tag{3}$$

Here,

B_S denotes the signal bandwidth after spreading, and
B denotes the signal bandwidth before spreading.

3 Proposed Methodology

(1) Generate a serial binary data stream 's_data'.
(2) Generate carriers for a hop-based modulation. Let the carriers be designated by $f_1, f_2, f_3, \ldots. f_n$.
(3) Generate the jammed signal s_jam which would be a modulated signal with a pseudo-random frequency pattern **'g'**.
(4) Then, s_jam can be given by:

$$s\_jam = \mathbf{g}(s\_data, f_1, f_2, f_3, \ldots f_n) \tag{4}$$

(5) Considering the spreaded bandwidth to be **βs** and the power of the transmitting source to be **Pi,** then the normalized interference caused to the jamming source can be given by:

$$I = (\beta s/Pi) \tag{5}$$

And the signal-to-interference ratio can be given by [4]:

$$(SIR)_1 = E_b/(\beta s/Pi) \tag{6}$$

where E_b represents the energy per bit.

(6) Considering additive white Gaussian noise (AWGN) conditions, generate noise and add to the jammed signal to emulate a practical channel [5]. Considering the noise to be given by n(t), the signal after the addition of noise can be given by:

$$s\_jam\_noise = s\_jam + n(t) \tag{7}$$

(7) De-jam the signal at the receiving end by using non-coherent conjugate detection described as:

If the transmitted carrier is given by [6]:

$$E_T(z, t) = E_{0.} \exp(-\alpha z) \cdot \exp(\omega t - \beta z) \tag{8}$$

And the receiving carrier is given by

$$E_R(z, t) = E_{0.} \exp(-\alpha z) \cdot \exp(\omega t - \beta z + \varphi) \tag{9}$$

Here, φ represents the phase difference between the transmitting and the receiving carriers. The design of a band-pass filter stops the carrier components out of band and recovers the signal 's_data'.

(8) Compute the BER and outage probability of the system.

4 Experimental Results

The results of the proposed system are evaluated in terms of:

(1) Bit error rate or probability of error given by: [2]

$$P_{re} = \frac{1}{\sqrt{2\,\pi\,\varphi_N^2}} \int_{\frac{b-b_2}{2\sigma_\varphi}}^{\infty} e^{(-x^2/2)} \varphi_N dx \tag{10}$$

Here, N represents the double-sided power spectral density of the noise.
Using the Q function, the BER of the system can be given by:

$$P_{re} = Q[(b_1 - b_2)/2\varphi_N) \tag{11}$$

Here,

b_1 and b_2 represent the bits 0 and 1,
φ_N represents the noise power spectral density (psd),
x is the random variable assuming two values for bits 0 and 1, and
Q represents the Q function.

The outage probability can be computed using the signal-to-noise-plus-interference ratio given by [7]:

$$\text{Prob}(S_{Rec} < S_T)$$

where

S_{Rec} represents the received signal power, and
S_T represents the threshold of signal power above which the quality of service is satisfactory.

Explanation of Results:
Figure 2a–d is combined and shown as a single figure for the sake of brevity and to augment the sequential flow of steps in the jamming and the de-jamming processes.

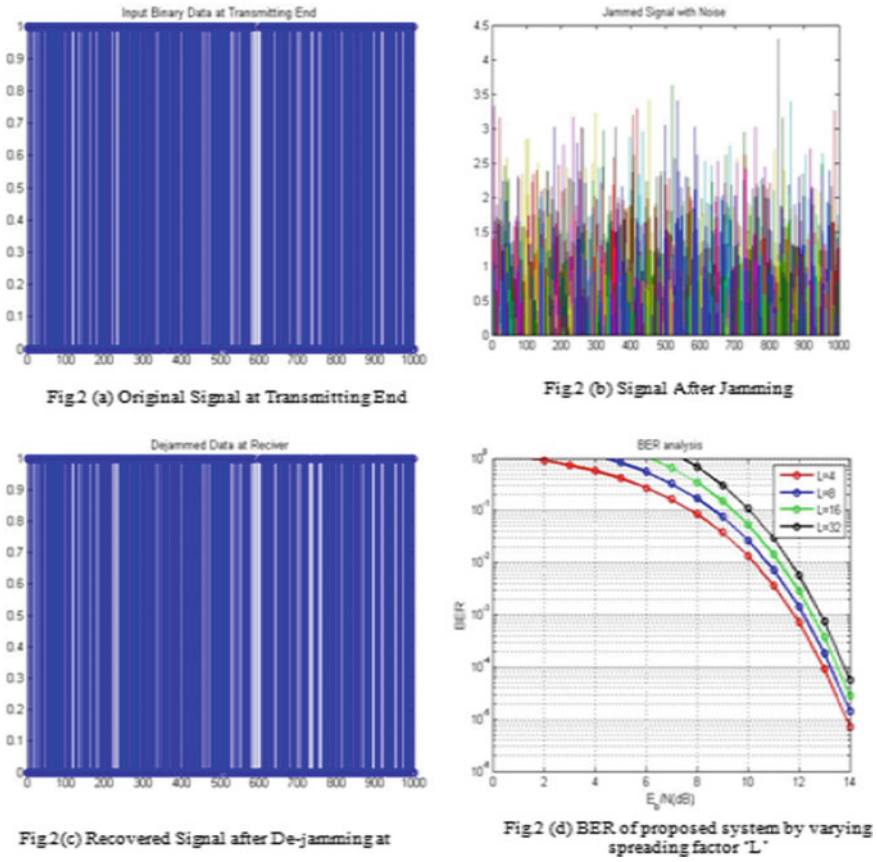


Fig. 2 **a–d** Represents the original binary message, binary message after jamming, de-jammed signal at receiving end and BER performance by varying spreading factor (L), respectively

Figure 2a depicts the discrete counterpart of the binary serial data which acts as the secret message. The discrete plot of 0 s and 1 s has been used to depict the binary message.

Figure 2b depicts the composite noisy jammed signal where noise is added in the channel.

Figure 2c depicts the de-jammed signal at the receiving end. The noise added is AWGN in nature, and it can be seen that after noise is added, the composite signal resembles noise to deceive potent adversaries.

Figure 2d depicts the variation of the BER of the system with increasing spreading factor. It can be seen that the BER reduces below 10^{-4} for spreading lengths up to 32. It can be seen that the BER degrades with increasing spreading factor which exhibits coherence with theoretical concepts, given by Eq. (11)

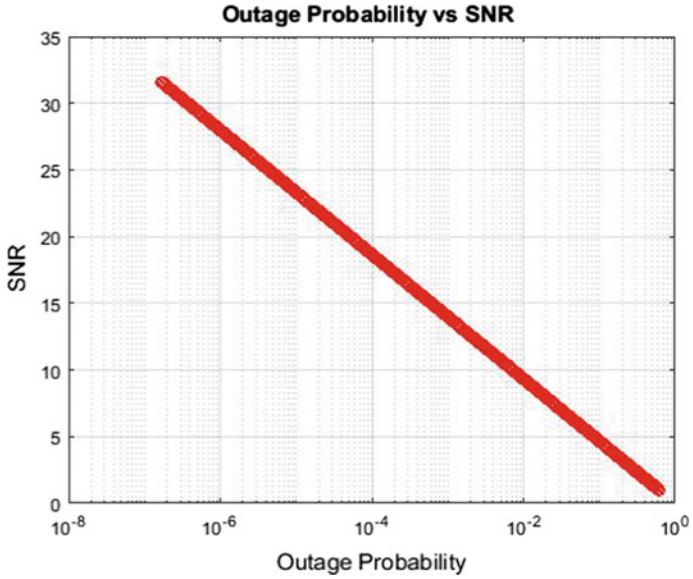


Fig. 3 Depicts the outage probability of the system as a function of signal-to-noise ratio

The outage probability of the system shows the relation among the quality of data communication as a function of signal-to-noise ratio. Thus, attaining relatively high SNR for high spreading factors would mean acceptable communication quality (Fig. 3).

5 Conclusion

This paper presents a band jamming technique in conjugation with non-coherent conjugate detection which is an effective technique for recovering the original signal at the receiving end in case the frequency hops frequently within a bit period and maintaining coherence between the transmitted and received carriers becomes difficult. Moreover, it can be seen that the proposed approach attains a BER of almost 10^{-5} at a relatively less SNR of 14 decibel. Thus, the technique is thought to mitigate the challenge of attack by adversaries and noisy nature of the wireless channel. It can also be seen that the proposed system's outage reduces with increase in signal-to-noise ratio (SNR).

References

1. W. Stallings, *Network Security and Cryptography*. 4th edn (Pearson Publication)
2. A.F. Molisch, *Wireless Communication* (Wiley India Publications)
3. J. Zhang, K. Teh, K. Li, Performance study of fast frequency hopped/ M-ary frequency-shift keying systems with timing and frequency offsets over Rician-fading channels with both multitone jamming and partial-band noise jamming. IET Commun. **4**(10), 1153–1163 (2010)
4. J. Zhang, K. Teh, K.H. Li, Maximum-likelihood FFH/MFSK receiver over Rayleigh-fading channels with composite effects of MTJ and PBNJ. IEEE Trans. Commun. **59**(3), 675–679 (2011)
5. L.-M.-D. Le, K.C. Teh, K.H. Li, Jamming rejection using FFH/MFSK ML receiver over fading channels with the presence of timing and frequency offsets. IEEE Trans. Inf. Forensics Security. **8**(7), 1195–1200 (2013)
6. L.-M.-D. Le, K. Teh, K. Li, Performance analysis of a suboptimum fast frequency hopped/M-ary frequency-shift-keying maximum likelihood receiver over Rician-fading channels with composite effects of partialband noise jamming and multitone jamming. IET Commun. **6**(13), 1903–1911 (2012)
7. F. Yang, L.-L. Yang, A single-user non coherent combining scheme achieving multiuser interference mitigation for FFH/MFSK systems. IEEE Trans. Wireless Commun. **12**(9), 4306–4314 (2013)

Modeling and Simulation of Secure Data Transfer in High Level Language Using Quantum Communication Protocol

Manoj E. Patil, M. Hussain and Swati Sharma

Abstract Secure data transfer is the main important task in the data transfer between the two or multiple nodes in the network. There are very large number of protocols and security mechanisms available. These security mechanisms can be broken by using the very high computing power processors. Quantum cryptography can be the promising secure communication using quantum physical laws. Quantum communication itself makes the communication more promising and reliable in terms of security. Many quantum cryptography protocols are proposed till date for secure communication including a popular "three-stage quantum cryptography protocol" (Kak in Phys Lett 293–296, 2006 [1]). In the current quantum cryptography protocols, the unit of data is a binary bit. Here, the proposed communication system is capable of transferring the data in the form of character rather than the bit-by-bit data transfer. This methodology uses the simple logic, i.e., the high level language is used for the data transfer. The main problem of secure key exchange is solved. The angle of rotation of the photon beam is being shared securely. Here, a secure multistage quantum communication protocol is implemented. This proposed protocol has minimized the network traffic drastically. The number of signals required for the transfer of the character is decreased six times of the previous one.

Keywords Quantum communication · Polarizer · TSQC

M. E. Patil (✉) · M. Hussain
SSBT's College of Engineering and Technology, Bambhori, Jalgaon, India
e-mail: mepatil@gmail.com

M. Hussain
e-mail: ermujahidhusain@yahoo.com

M. E. Patil · S. Sharma
Department of CSE, Jodhpur National University, Jodhpur, India
e-mail: er.swati.sharma15@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_29

1 Introduction

During this transit of the data through the communication channel, there is the possibility of information interception. The important data may leak, and the organization may be in trouble. To secure these data from the intruder, there are many more cryptography methods proposed.

Basically, the cryptography is divided into two parts: symmetric keys cryptography and asymmetric key cryptography. These methods basically work on the mathematical formulas and use the same or different keys for encryption and decryption. But nowadays the computers are coming with the huge computing power. Due to this, the traditional cryptography methods may be compromised. To overcome this disadvantage, new cryptography method is proposed called as quantum cryptography.

Quantum cryptography [2] is an approach for secure communications by applying the properties of quantum physics. In traditional classical cryptography, mathematical techniques are used to restrict eavesdroppers. The quantum cryptography is focused on the physics of information. According to quantum physics, information is abstract. It acquires the physical properties of the medium on which it is stored or through which it is communicated. For example, information will be governed by the laws of magnetic field when it is written on hard disk, and it will be governed by the laws of light when it is transmitted optically. Quantum cryptography provides a way of secure communication, as its security is guaranteed directly by the laws of quantum physics. The field of quantum cryptography began after the formative work of Charles Bennett and Gilles Brassard in 1984. They proposed the BB84 algorithm [3]. Today quantum cryptography is supposed to become a dominant part of communication domain.

Subhash Kak implemented the three-stage quantum cryptography protocol (TSQC) [1] first time in the laboratory. Implementation details are given in these papers [4–6]. Current implementation of this protocol tolerates multiple photons in the secure communication process. This makes it superior to BB84 protocol [3] and its variants that require prefect single photon source for their secure implementation.

Multistage quantum communication protocol (MSQC) is proposed in this paper. The fundamental modifications are proposed in TSQC protocol so that the latter can transfer data in high level language rather than the currently adopted bit-by-bit data transfer approach.

2 Background

Quantum cryptography started with the introduction of BB84 protocol in the year 1984 [7]. The protocol used single light photon to transfer one bit of data. To be precise, the protocol used the polarization states of photons to realize the key distribution.

First quantum cryptography protocol, BB84 protocol, was designed to establish a secret key between two parties. The established key was then used to scramble actual communication. This key was established over the communication channel popularly (quantum channel). The actual communication was carried out on classical channel. As the time passed, many modifications were made in the original definition of this protocol, but the basic idea remained the same, "quantum channel will be used for key distribution, and once the key is established, actual communication will be done on the classical channel using that previously shared key." This communication approach dominated quantum cryptography domain for nearly two decades.

In year 2006, Subhash Kak introduced the TSQC protocol. This protocol brought the paradigm shift; it obliterated the need of classical channel and thus broke the 20-year-long tradition of using two channels for communication. A big achievement indeed! Due to the use of only a quantum channel, TSQC protocol enjoys the benefits of "no cloning theorem" during the whole communication process [8, 9, 10].

The main drawback of TSQC protocol is that it needs three signals to send one-bit data from source to destination, thus the name—three-stage cryptographic protocol. If the ASCII character encoding system is considered, total $3 \times 8 = 24$ signals are to be exchanged to transmit 1 alphabet. The proposed MSQC protocol, theoretically, is capable of doing the same work within 4 signals.

3 Methodology

The foundation of the MSQC protocol lies in its encoding system. Instead of converting the alphabet into a combination of binary bits, each character of the communication language is assigned a particular unique light intensity. The upper boundary of this light intensity is established in advance by the agreement between communicating parties. MSQC protocol also uses a filtering polarizer at the destination to filter out the incoming light before measuring its intensity [11, 12].

In the MSQC protocol, intensity of output light depends upon the angle between the direction of polarization of light and the direction of the axis of filtering polarizer. This property of light is explained by Malu's law in optical physics. Thus, the intensities assigned to each character in the encoding system correspond to angles between the prepared state of polarization of photons (at the sender's end) and the axis of polarizer used as filter before measuring the output intensity (at the receiver's end).

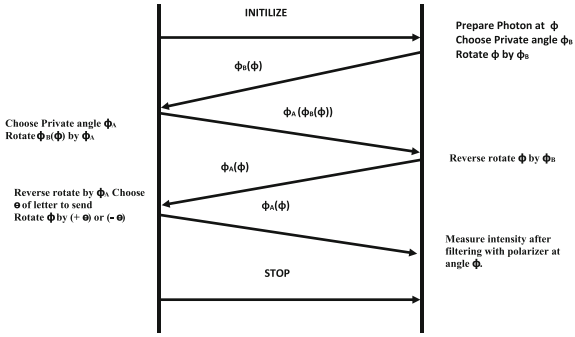


Fig. 1 Multi stage quantum communication protocol

4 Multistage Quantum Communication Protocol

The whole process of MSQC protocol is shown in Fig. 1. This process is explained in step-by-step manner in the algorithm given below.

Sender will send INITIATE signal to receiver.

On receiving INITIATE signal, the following will take place:

a. Receiver will choose random angle between 0° and 360° as the angle of filtering polarizer φ and will prepare photon pulse polarized at this angle.
b. Then, he will select a random private rotation angle φB and will rotate the photon pulse with φB and send the result to sender.
c. Sender will apply rotation equal to her privately chosen angle φA, set the flag bit and send the modified photon pulse to receiver.

Receiver will reverse his private rotation φB and send the pulse back to sender.

Sender will also reverse her rotation φA.

At this moment, she will have photon pulse polarized at an angle chosen by receiver as an angle of filtering polarizer φ.

Now she will rotate the resultant pulse with the angle θ corresponding to the letter she wants to communicate with receiver and finally send the resultant light pulse to him.

This rotation θ can either be clockwise or anticlockwise. Choice will be made by sender.

Receiver will now measure the intensity of the incoming photon pulse after passing it through the polarizer aligned at an angle φ, thus receiving intensity corresponding to the letter sent by sender.

The receiver also checks the flag along with the incoming signal intensity.

Steps 2 through 6 will be repeated until STOP signal comes from sender.

Algorithm 1: Implementation of TSQC protocol is presented here. Algorithm accepts three inputs, viz. linear horizontally polarized light pulse, linear vertically polarized light source and the message string to be communicated.

Algorithm 1 TSQC Protocol

```
  procedure TSQC Protocol
(ψLHP,ψLVP, String msg)
  msg = input(msg)
  for η = 0 to msg.lenght() -1 do
  enc_char = encode_to
ASCII(msg[η])
  for ρ=0 to enc_char.lenght() -
1
  do
  if enc_char [ρ] == 0 then
  Ψ = turn_on_light_source("LHP")
  else
  Ψ = turn_on_light_source("LVP")
  end if
  α = random (0,360);
  β = random (0,360)
  TS1= transmt
       ( prep_for_stage1(ψ, -α)
  TS2= transmt
       ( prep_for_stage2(TS1, -β)
  TS3= transmt
       ( prep_for_stage3(TS2,α,
enc_char[ρ])
  output_bit = decode(TS3,β)
  end Procedure
```

Algorithm 2: MSQC Protocol {Unsecured Mode}

```
   Algorithm 2 MSQC Protocol

   Unsecured Mode
   procedure MSQC_ proto-
col_unsecured (ψLHP,
String_msg)
   msg = input(msg)
   encoded _msg = encode(msg)
   for η = 0 to encod-
ed_msg.lenght() - 1 do
   Ψ=
turn_on_light_source("LHP")
   ω = get_rotator
         (encoded_msg[η])
   T = ψ × ω
   T0 = transmit(T)
   output_char = decode (T0 )
   output_chars.append
         (output_char)
   end for
   ς = generate_trace_file
   (encoded_msg.output_chars)
   animate_communication(ς)
   end procedure
```

An implementation for MSQC protocol in unsecured mode is presented here. It accepts two inputs, viz. linear horizontally polarized light pulse and a message string to be communicated.

5 Results and Discussion

The simulator is started by using the command Python3 simulation test.py. The input to this program is string JNU. The simulation is done step by step. The results are stored in trace file. First, the light source is generated with the intensity given in the encoding table. The angle of polarization for the light beam is selected, and the polarized light is passed through rotator which is set at the given angle. The polarized light with the flag bit is received at receiver. The receiver passes this light beam through the rotator and reads the light at the polarization angle shared by the sender along with the flag. Hence, the output is displayed.

Algorithms 1 and 2: It is observed that the former required 2 for loops to implement the communication cycle, while the latter requires just one. From these algorithms, it can be easily deduced that signal requirements for MSQC protocol unsecured mode can be given by Eq. 4n and for TSQC protocol it is obtained by Eq. 24n

(ASCII encoding system is considered to get this equation), where n is the number of characters transferred. On the other hand, if unsecured mode of MSQC protocol is considered, only n signals are required to transfer n characters.

6 Conclusion and Future Work

The implementation results indicate that the proposed communication system reduces network traffic of TSQC protocol approximately by four to six times, depending on message size. Network traffic is reduced approximately by 5 times when the message length is greater than 3 characters, and it reduces to 6 times when the message and secure banking transaction communication.

Attempt could be made to fit more characters in the 90° window of the encoder, and efforts can be made to provide implementation specific prototypes, length goes over 40 characters. This new protocol transfers data in high level representation of language like English directly, instead of its binary translation. The flag can be used to increase the number of characters of the high level language to be included. The proposed protocol also gives a chance to judge the integrity of communication medium before sending actual data.

This feature is not present in TSQC protocol or any of its variants. The protocol opens new opportunities and possibilities of using light as a medium for secure data transfer. Almost all the characters of the high level language (English) are incorporated.

Here, the implementation of intensity-based communication environment is done in a procedural manner. It can be developed in an object-oriented fashion. System developed in the object-oriented fashion can implement the role-based access to various equipments of the communication system which will become easy. Actual in-lab implementation of the proposed system needs to be done. Results of practical implementation could be used to configure simulation environment more precisely. Protocol has the potential scope in cloud server communication and secure banking transaction communication.

References

1. S. Kak, A three-stage quantum cryptography protocol. Found. Phys. Lett. 293–296 (2006), http://arxiv.org/abs/quant-ph/0503027. Accessed 20 Nov 2016
2. W.P. Eleanor, G. Rieffel, An introduction to quantum computing for non-physicists. ACM Comput. Surv. **32**(3), 300–335, http://doi.acm.org/10.1145/367701.367709. Accessed 19 Nov 2016
3. P. Basuchowdhuri, *Comparing BB84 and Authentication-Aided kak's Three-Stage Quantum Protocol* (2007), CoRR http://arxiv.org/abs/cs/0703092. Accessed 10 Nov 2016

4. S. Mandal, G. Macdonald, M.E. Rifai, N. Punekar, F. Zamani, Y. Chen, S. Kak, P.K. Verma, R.C. Huck, J. Sluss, *Implementation of Secure Quantum Protocol using Multiple Photons for Communication* (2012). CoRR http://arxiv.org/abs/1208.6198. Accessed 20 Nov 2016
5. Y. Chen, S. Kak, P.K. Verma, G. Macdonald, M.E. Rifai, N. Punekar, Multi-photon tolerant secure quantum communication—From theory to practice, in *IEEE 'ICC'* (2013), pp. 2111–2116
6. S. Mandal, G. Macdonald, M. El Rifai, N. Punekar, F. Zamani, C. Yuhua, S. Kak, P.K. Verma, R.C. Huck, J. Sluss, Multi-photon implementation of three-stage quantum cryptography protocol. in *IEEE 2013 International Conference—Bangkok Information Networking (ICOIN)* (2013), pp. 6–11
7. C.H. Bennett, G. Brassard, Quantum cryptography: Public key distribution and coin tossing, in *Proceedings of IEEE International Conference on Computers, Systems, and Signal Processing* (IEEE Press, India 1984), pp. 175–179
8. S. Kak, *Threshold Quantum Cryptography* (2013). CoRR http://arxiv.org/pdf/1310.6333. Accessed 20 Nov 2016
9. J.H. Thomas, *Variations on kak's Three Stage Quantum Cryptography Protocol (2007)*. CoRR http://arxiv.org/abs/0706.2888. Accessed 12 Nov 2016
10. V. Scarani, C. Kurtsiefer, *The Black Paper of Quantum Cryptography: Real Implementation Problems* (2012), http://arxiv.org/abs/0906.4547v2. Accessed 15 Nov 2016
11. S. Kak, Y. Chen, P.K. Verma, *iAQC: The Intensity-Aware Quantum Cryptography Protocol* (2012). CoRR http://dblp.uni-trier.de/db/journals/corr/corr1206.html#abs-1206-6778. Accessed 20 Nov 2016
12. S. Chitikela, *Intensity and State Estimation in Quantum Cryptography* (2013) CoRR http://arxiv.org/abs/1302.1823. Accessed 14 Nov 2016

Secure and Efficient Data Privacy, Authentication and Integrity Schemes Using Hybrid Cryptography

Sourabh Bhat and Vivek Kapoor

Abstract Information security is the one of the utmost substantial concerns in every field of today's era. In order to secure the information transmission over the uncertain channel, currently numerous cryptographic techniques are practiced. But many precincts are present in the existing symmetric encryption and asymmetric encryption techniques. The key interchange is a foremost problem of symmetric encryption techniques but sooner in encryption. Extended encryption time is major problem accompanied with asymmetric techniques though key interchange is simple. Now, to overwhelm these glitches, in this paper a hybrid model is proposed which offers excessive security with lessened key maintenance and encryption time using amalgamation of both symmetric and asymmetric cryptographic techniques. In this hybrid model, to achieve the embryonic security services as integrity, confidentiality and authentication by encompassing message digest, symmetric encryption and digital signature, respectively, a digital envelope is also included which comprises all of this to transfer them firmly over the network.

Keywords Asymmetric encryption technique · Symmetric encryption technique Digital signature · Digital envelope · Hybrid cryptography model

1 Introduction

In this projected work, a hybrid model is employed which is the amalgamation of symmetric and asymmetric cryptographic techniques and digital signature. This proposed model offers an enhanced hybrid cryptographic method which delivered secured environment with less resource ingestion.

Hybrid cryptosystem: This cryptosystem (provides excessive security with minimised key maintenance) is the mishmash of both symmetric and asymmetric

S. Bhat · V. Kapoor (✉)
Information Technology Department, Institute of Engineering & Technology,
Devi Ahilya University, Indore, Madhya Pradesh, India
e-mail: vkapoor13@yahoo.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_30

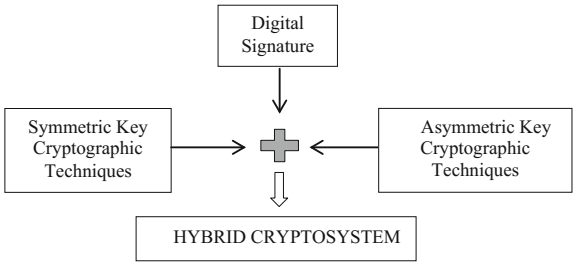


Fig. 1 Hybrid cryptosystem

cryptography techniques, taking pluses of both cryptography and left their downsides. Symmetric encryption process is faster than the asymmetric encryption, but sharing of the secret key is the problem accompanied with symmetric cryptography. In Fig. 1, a hybrid cryptosystem is presented which mainly comprises of symmetric and asymmetric techniques and digital signature.

Message Digest: A message digest is a mathematical cryptographic hash function comprising a series of alpha numerals created by a one-way hashing principle. Message digests are aimed to preserve the integrity of message to sense amendments to any portion of a message. Unique message digest is allotted to specific data volume. They are algorithmic numbers.

Digital Signature: It is an asymmetric encrypted form of the mathematical hash numbers. It is the digital equivalent of a handwritten signature used for authentication of a message/document. The digital signature is created by applying private key to encrypt the hash numbers [1].

This hybrid model comprises only benefits of a symmetric and an asymmetric cryptography method. To acquire confidentiality and efficient performance to the hybrid system, symmetric key encryption is applied using private key of the sender's side user. Symmetric key encryption is drastically quicker encryption process than all other archaic public key cryptographic encryption methods. To contract integrity of the input message, one-way hashing is applied on it to yield message digest of input message, which must be relating with message digest of the recuperated message from decryption process at receiver's side. If message digests is not matched, shows the interruption with the original message in the channel implies no integrity. The encryption of hash value of input message with private key of the public key cryptography technique creates a digital signature which authenticates the desired receiver. A digital envelope is applied to get assurance over the channel and consists of digital signature, encrypted input message and the double encrypted private key of symmetric key technique.

2 Literature Search

In the literature search, description of understanding on some research papers of different cryptographic approaches is used for securing the information from the untrustworthy networks.

This paper aims on security enhancing by improving the level of encryption in channel. This study's main objective is to reveal the significance of security in network and provide the enhanced encryption technique for presently implemented encryption methods. In this research, the author has projected a combination of MD5, RSA and DSA as a fusion link for wireless devices and also derived a case study for MANET networks suitable to suggest the algorithm's applications [2].

This novel method enciphers the image and implants the digital signature into the image. A digital signature is a method used to accomplish integrity of data, digital documents, software and endorse the authenticity. Images are well regarded to cover objects used in steganography which is also an encryption procedure. Here, the Java methodology is planned to substantiate the performance of the proposed model in circumstances of key length, encrypted text length, message length, encryption time and decryption time [3].

In this paper, a new security protocol is designed to improve the strength of security algorithms for online transaction using grouping of both symmetric and asymmetric cryptographic techniques and offers three cryptographic services such as confidentiality, integrity and authentication. These services are conquered by using elliptic curve cryptography (ECC) (for encryption), message digest (MD5) (for integrity) and dual RSA cryptographic method (for authentication) [4].

This study refers to several aspects of cryptographic procedures and many issues related to cryptography. The proposed work is screening some of the crucial issues of cryptography along with their solutions. DES cryptographic method is used for encryption purpose, and RSA is used for symmetric and asymmetric cryptography techniques; also, hash method (SH1) is used in this paper [5].

In this paper recommend cipher scheme that is advances the Diffie–Hellman key interchange by using truncated polynomial in discrete logarithm problem to rises the complexity of this scheme over the unsafe channel, also adding the MD5 hashing algorithm, the AES symmetric key technique and the Modification of Diffie–Hellman (MDH) asymmetric key technique [6].

Diffie–Hellman is amended to offer authentication and elude primeval root generation stage to attain speed and authentication to elude key interchange with unauthenticated operator. In this paper, hybrid cryptography arrangement is proposed to accomplish secret message interchange. RSA uses any prime figures P and Q which multiplied an get value N is shared to the receiver side [7].

This paper studied as the information security can be interpreted into three key primitives: integrity, availability and data safety. In this work, present their efficiency by comparing the several kinds of cryptographic algorithms and by demonstrating their flaws and assets. In order to enhance the merits of the crypto procedures, we suggest a hybrid methodology that chains three cryptographic procedures [8].

Secure Electronic Medical Records (SEMR), which ambitions of delivering several facilities which will offer protected and competent access of the EMRs to the doctors, consultants and patients. In this paper work, the author proposes an implementation of a hybrid model that combines the MD5 hashing algorithm, AES symmetric key algorithm and HECC asymmetric key algorithm in the digital envelope. The outcome demonstrates the finest substitute digital envelope hybrid cryptographic-based system for EMR [9].

3 Rationale

Due to existence of great computing stimulus processors which is crafting diversified circumstances for secret key encryption with the smaller size of the key because dispersed computation methods can break smaller size of key simply. Another difficulty is the key interchange in secret key encryption method over the unsecured network. A disadvantage of using asymmetric procedure is slow speed encryption procedure. All popular secret key encryption procedures are significantly faster than any presently existing symmetric encryption procedure. Public key cryptography procedure is not viable in case of enormous information. Numerous hybrid prototypes are made to overcome these situations.

4 Problem Statement

With the antiquated cryptographic scheme, two parties essentially come to an agreement to share a secret key and retain it secretly among them. If they are in different places, they must faith a dispatcher or some other protected communication channels to inhibit the revelation of the secret key during broadcast. Anyone who eavesdrops the key in transport can later read, amend all information encoded or legitimate with that key.

In a cryptographic system, the difficulty is that it cannot achieve this entire problem in a single stage like confidentiality, integrity and authentication. In a present scheme, the invader outbreaks on a system simply, and brute force attacks effortlessly occur because of the necessity of security augmentation. Conventional system takes more time in encryption and decryption; because of this, our system may behave poorly and produce difficulties. Hacking is the most significant crunch in the existing system.

5 Proposed Solution

The main goal of the proposed model is to acquire the primitive aims of the cryptography, that is, authentication, integrity and confidentiality. A proficient, secure hybrid model is to be implemented using amalgamation of the cryptographic methods that mends the performance of the outmoded cryptographic security procedures. In this

proposed model, symmetric encryption of message data (attained confidentiality) and digital signature provide authentication and relating the hash values gives integrity and digital envelope provides add-on security over the network.

6 Proposed Methodology

Encryption Process

Sender Side Process:

- Symmetric encryption: [Psym, PT] → CT
- One-way hashing: [F hash(), PT] → MD(1)
- Asymmetric encryption: [Kpri(s), MD(1)] → DS
- Asymmetric encryption: [Epri(s), Ksym] → ESK
- Asymmetric encryption: [Kpub(R), ESK] → DESK
- Digital enveloping: [CT, DESK, DS] → DE

In Fig. 2, sender side scenario is shown which is described in the followings steps.

1. Plain text (PT) is encrypted using symmetric key (Ksym) to get cipher text (CT).
2. Message digest (fixed length numerals) is formed by applying one-way hashing algorithm (mathematical hash function) on input message (any length), and this message digest is encrypted using sender's private key (Kpri(s)) to get the digital signature (DS).
3. Firstly, symmetric key (Ksym) is encrypted using sender private key (Kpri(S)), and then, this encrypted symmetric key (ESK) is encrypted using receiver's public key (Kpub(R)) to get double encrypted symmetric key (DESK).
4. A digital envelope (DE) is made to transfer three CT, DS, DESK over the channel.

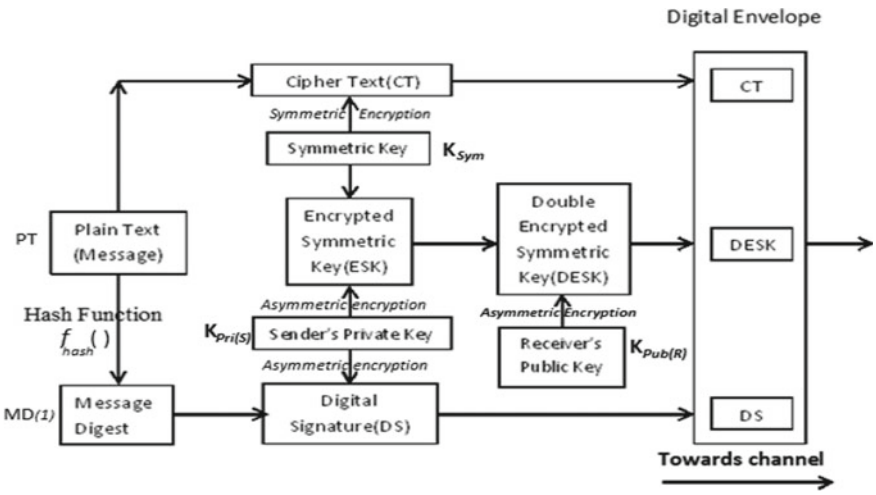


Fig. 2 Sender side scenario of hybrid model

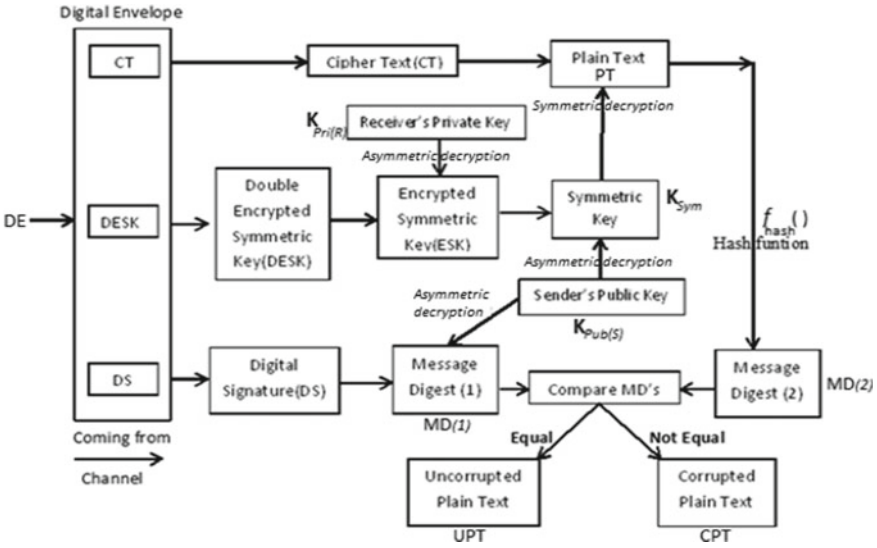


Fig. 3 Receiver side scenario of hybrid model

Decryption process

In Fig. 3, receiver side scenario is shown which is described in the following steps.

1. Incoming digital envelope (DE) from the channel consisted of CT, DESK, DS.
2. The double encrypted key is first encrypted with receiver's private key, and the encrypted symmetric key is further encrypted with sender's public key, and finally get symmetric key.
3. This symmetric key is used to decrypt the cipher text (CT) to plain text. This plain text (PT) is converted into message digest using hashing algorithm and get message digest(2).
4. The digital signature is obtained from digital envelope is decrypted using sender's public key to get message digest(1).
5. Message digest(1) is compared with message digest(2). If both are same, then uncorrupted plain text (UPT) is obtained, and when both are not same, then corrupted plain text (CPT) is obtained.

Receiver Side Process:

- Open digital envelope: DE → [CT, DESK, DS]
- Asymmetric decryption: [Kpri(R), DESK] → ESK
- Asymmetric decryption: [Kpub(S), ESK] → Ksym
- Symmetric decryption: [Ksym, CT] → PT
- One-way hashing: [F hash(), PT] → MD(2)
- Asymmetric decryption: [DS, Kpub(S)] → MD(1)
- Comparing message digest: [MD(1), MD(2)] → CPT or UPT.

7 Expected Outcomes and Conclusion

Expected outcomes: This hybrid model offers an efficient, inherent secured with excessive performance cryptographic system than the conventional cryptography method-based systems. Also, the encryption time of this approach will be slighter than traditional public key cryptographic methods and will deliver better key maintenance than the other outmoded symmetric key techniques. The main outcomes of this model are the important security services as confidentiality integrity and authentication and additional confidence over the network.

Conclusion: Today's modest and digitalised surroundings with progressively fraught from monetarily inspired hackers and unsatisfied employees are generating excessive demand for effectual, automated, dominant and risk-alleviating ways to manage and protect keys all over their lifespan, so that the access must be granted only to the certified users. For guaranteeing the attainment of key security aims a robust hybrid cryptosystem along with authentication based management system, and a secure key encryption-based system is proposed.

References

1. W. Stallings, Cryptography and Network Security, 3rd edn. (Prentice-Hall Inc., 2005)
2. K. Kaur, E. Seema, Hybrid algorithm with DSA, RSA and MD5 encryption algorithm for wireless devices. Int. J. Eng. Res. Appl. (IJERA) **2**(5) (2012)
3. S. Sharma, V. Kapoor, A novel approach for improving security by digital signature and image steganography. Int. J. Comput. Appl. **171**(8) (2017)
4. S. Subasree, N.K. Sakthivel, Design of a new security protocol using hybrid cryptographic algorithm. IJRRAS **2**(2) (2010)
5. D.V. Kapoor, R. Yadav, A hybrid cryptography technique to support cyber security infrastructure. Int. J. Adv. Res. Comput. Eng. Technol. **4**(11) (2015)
6. A.M. Rahma, R.N. Farhan, H.J. Mohammad, Hybrid model for securing E-commerse transaction. Int. J. Adv. Eng. Technol. Nov 2011. © IJAET 14 **1**(5), 14–20
7. S. Deshmukh, R. Patil, Hybrid cryptography technique using modified Diffie-Hellman and RSA. Int. J. Comput. Sci. Inf. Technol. (IJCSIT), **5**(6) (2014)
8. G. Mateescu, M. Vladescu, A hybrid approach of system security for small and medium enterprises: combining different cryptography technique, in *Proceedings of the 2013 Federated Conference on Computer Science and Information Systems*
9. M. Gobi, K. Vivekanandan, A new digital envelope approach for secure electronic medical records. IJCSNS Int. J. Comput. Sci. Netw. Secur. **9**(1) (2009)

An Enhanced Cryptographic System for Fast and Efficient Data Transmission

Sandeep Verma, Vivek Kapoor and Rahul Maheshwari

Abstract As technology is growing, mobile devices are replacing the traditional systems, and it is increasing the ease of access; but at the cost of resource availability, these lightweighted devices come with lesser memory and processing power. The network available for such devices is often wireless; hence, data transmission rate is generally slower than that of traditional system. Thus, in order to provide security measures for such devices, various parameters need to be considered. The security mechanism should be fast as well as efficient enough to overcome those limitations. In order to implement security over network, various techniques can be used, and cryptography is one of the most commonly used methods. It converts the readable plain text into a non-readable format known as cipher text. There are various techniques to implement cryptography. The proposed system is using an enhanced cryptographic technique which will be secure as well as efficient. It uses compression prior to encryption; hence, both time and space are reduced for further operations, and as a result, both time and space are saved which play a vital role in enhancing the performance, especially when it is a mobile device. In order to enhance, the security file is converted into byte code and then it is broken into chunks which are further treated with XOR operation to increase the randomness of data. Several passes of exor are performed to make data as random as possible so as to make it less vulnerable against brute force attack. In this way, the proposed system enhances the security as well as performance.

Keywords Cryptanalysis · XOR · Data compression · Fragmentation · MD5 Crypto-compression system

S. Verma
Finastra Software Pvt. Ltd., Trivandrum, Kerala, India

V. Kapoor (✉)
Information Technology Department, Institute of Engineering & Technology, Devi Ahilya University, Indore, Madhya Pradesh, India
e-mail: vkapoor13@yahoo.com

R. Maheshwari
Sushila Devi Bansal College of Technology, Indore, Indore, India

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_31

1 Introduction

Data security means applying some security measures to data in order to avoid illegal access. In today's world of globalization, more and more information are being shared across the globe. The value of information is also increasing. Hence to protect this information from being compromised, various security measures are required. In order to protect the information from unauthorised access, cryptography has been used. Although it is being used since very long time, it has been evolved over the years. It reduces the risk of data being compromised by changing the data being transmitted into unreadable form. There are certain security aspects which need to be considered, i.e.

Confidentiality—It ensures that information is available to its intended user only. There should not be any unauthorised access to data while transmission. As data is sent over public network, it needs to be protected from being compromised. It can be achieved by encrypting the data so that even if it is compromised, it would be very hard to get the information out of it.
Authentication—It ensures that the identity of the user is same as that of claimed. The identity of user should be verified before providing access to information.
Integrity—It ensures that the content of information should not be altered while transmission.
Non-Repudiation—It ensures that user cannot deny any action performed. It provides a proof of work performed by the user so that he cannot deny it in the future.
Access Control—It ensures that resources should always be available to its intended user uninterrupted.

2 Cryptography

It is an art which hides the information from unauthorised users over the network. While transmission at senders end, readable text is converted into unreadable text using a predefined algorithm. It works with the following components:

Plain Text—It is the original file that needs to be sent over network.
Cipher Text—It is the unreadable form of original file that is converted into unreadable form.
Encryption—It converts the original file (plain text) into unreadable file (Cipher text) as shown in Fig. 1.
Decryption—It converts the unreadable file (cipher text) into readable file (plain text) as shown in Fig. 2.
Cryptographic Algorithm—It consists of set of steps that need to perform to either encrypt or decrypt the file.
Public Key—It is the key which is used in encryption and known to everyone.
Private Key—It is the key which is used in decryption and is kept secret.

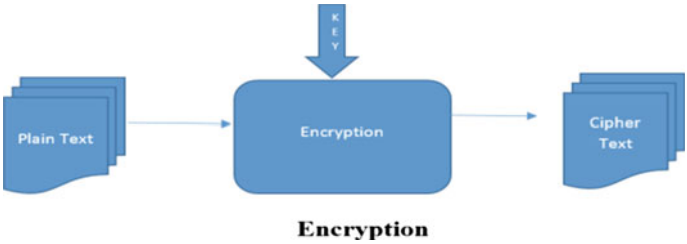


Fig. 1 Encryption

Fig. 2 Decryption

Cryptography is broadly classified into two types depending on the number of keys being used, i.e.

2.1 Symmetric Key Cryptography

In this technique, same key is used for encryption as well as decryption. Both sender and receiver share secret key by a secure transmission method. This is further divided into block and stream cipher. In block cipher data is divided into blocks, and they are processed individually while in stream cipher several chunks of data known as stream are processed. The strength of this technique often relies on the length of key being used. The security of data against various attacks would be greater if the key length is greater. One of the issues with this technique is sharing of key. It requires a proper mechanism to share the key via network, but at the same time it is much easier to implement and efficient as well. It is shown in Fig. 3.

2.2 Asymmetric Key Cryptography

In this technique, pair of keys are used known as public and private keys as shown in Fig. 4. The sender encrypts the data using public key while receiver decrypts the

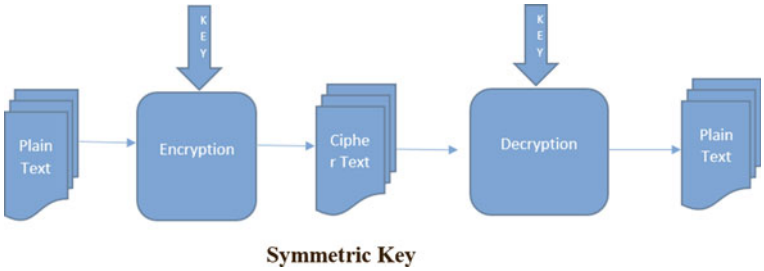


Fig. 3 Symmetric key encryption

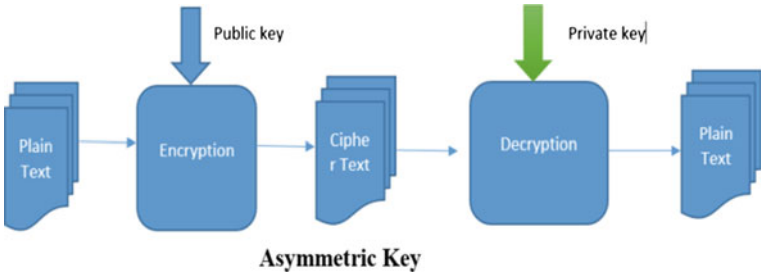


Fig. 4 Asymmetric key encryption

data using private key. This adds one more security aspect to this technique as key needs not be shared in secure manner because private key is intended for single user only, i.e. receiver. But at the same time, this process of key distribution sometimes becomes very complex and time-consuming.

3 Cryptanalysis

Cryptanalysis is a science of recovering plain text without having any knowledge of key or cryptographic algorithm being used. There are various types of cryptanalysis, such as

(1) *Known-plaintext*: In this type of cryptanalysis, attacker gets some portion of plain text from cipher text and then tries to deduce the key from it.
(2) *Chosen-plaintext*: Here attacker chooses a plain text encrypted with a key and gets the resulting cipher text. The cryptanalyst is able to have any plaintext encrypted with a key and obtain the resulting cipher text, and then this resulting pair of plain text and cipher text is analysed to get the original key. RSA algorithm can be compromised with this technique.
(3) *Cipher-text-only*: Here attacker gets some portion of cipher text without having any knowledge of plain text. In this case, proper analysis is required to get the

lain txt. The cryptanalyst has no knowledge of the plaintext and must work only from the cipher text.

(4) *Man-in-the-middle*: In this type of attack, attacker places himself between sender and receiver. The attacker performs key exchange with two parties while two parties have no idea of existence of attacker. In this way, man-in-the-middle attack occurs.

(5) *Timing analysis*: It is a new type of attack which uses the difference in the consumption of electricity in certain time period whenever a function is performed by microchip over a secure information. In this way, it predicts the information about computation of key used in the cryptographic algorithm.

4 Literature and Survey

In the recent years, there is an explosion in the amount of information being exchanged over network; hence, it becomes very important to provide proper security measures. In order to provide a secure environment to send data over network, a proper analysis of present security mechanism needs to be done. Recently, one of the existing systems uses compression-based mechanism along with RSA algorithm for lightweighted devices such as mobile phones and PDA's [1]. This system provides a secure way to send message over the network [2]. This system uses compresses technique to reduce the length of message, and then encrypts it by using RSA algorithm. RSA algorithm is an asymmetric algorithm which uses public-key encryption method. One disadvantage of using asymmetric cryptography is speed and time complexity. This system is also known as hybrid compression encryption system (HCE). One another method uses compression-based cryptography to transmit medical related information. It uses compression methods like sequitur for reducing the size of data being sent. The combination of McEliece public-key cryptosystem with compression provides confidentiality in the transmission [3]. This system has a drawback as its efficiency drops with increase in data. The proposed system uses symmetric key cryptography, and hence, it is faster than asymmetric key cryptography and uses compression technique to reduce the size of cipher text.

5 Performance Parameters

As data is being transformed while performing encryption or decryption, there are some parameters which need to be considered while designing any cryptographic algorithm which affects the performance of cryptographic technique. Some of the major parameters which need to be focused are as follows:

Time Complexity: It quantifies the amount of time required by any cryptographic algorithm to perform its operation. The execution of any algorithm should be as efficient as possible. One of the ways to improve its efficiency is to minimize the

time taken by it to perform various operations by applying some optimization technique. Hence, performance of any cryptographic algorithm can be judged using this parameter.

Space Complexity: It quantifies the amount of space required by any cryptographic algorithm to perform its operation. As discussed earlier in lightweighted devices, there is a limitation of resources, i.e. memory and processing power; hence, it becomes very important for any cryptographic algorithm to use these resources as optimally as possible. Space complexity deals with the space required by the algorithm to perform its operations. It should be designed in such a way that the space required is as minimal as possible. This is another parameter to check cryptographic performance.

Size of Cipher Text: To analyse the performance of cryptographic algorithm, the size of cipher should also be considered. It should not be very large as compared with plain text. As data is being sent over network, size of data should not be changed drastically after applying cryptographic algorithm. Hence, size of cipher text also decides the performance factor.

Security: It is the most important performance aspect of any cryptographic algorithms, and it deals with the robustness of algorithm against various kinds of attacks. The algorithm should be designed in such a way that it should not be cracked by any attacker by any means. It should be strong enough to dodge various attacks over network. Even in case if data being sent is compromised, the attacker should not be able to get the information out of it.

6 Proposed Work

The proposed system is an enhanced cryptographic system which is a compression-based cryptographic technique [4]. It takes any file format as an input and compresses it to reduce its size and then encrypts the file into unreadable format (cipher text). In the same way, it can also take unreadable file as input and converts it back into its original form. In this way, proposed system is an efficient way to encrypt the data to send it over the network. It uses the basic functionality of XOR operation to scramble the data to make it unreadable. The number of XOR operations depends upon the size of the data being encrypted. To improve the performance, first of all the data is compressed and its size is reduced from its original form; hence, it takes lesser time to encrypt the data. The encryption technique is also kept simple yet hard to break to make system more efficient especially for mobile devices. In the end, the data is converted into hex format to further reduce its size. In this way, the proposed system applies several enhancements to data to make it as efficient as possible to transmit over the network.

Steps of proposed Algorithm:

Encryption Algorithm

- Initially, select the file (Plain text) to be encrypted (any format).

- Apply compression and fragmentation (16 byte each) on selected file.
- Apply MD5 on selected file to create its hash function (16 byte).
- Apply XOR operation between key (hash function) and first block of fragmented file. Result will store as result1.
- Apply XOR operation between result1 and second block of fragmented file. Result will store as result2.
- Repeat last step until whole data gets converted into Result.
- Save all the blocks of results (result1, result2 … resultn) to Result (Cipher text).

Decryption

- From Result (Cipher text) get the blocks (result1, result2 … resultn) from Result.
- Apply compression and fragmentation (16 byte each) on selected file.
- Apply XOR operation between key (hash function) and first block of fragmented file. Result will store as text1.
- Apply XOR operation between result1 and result2 to get text2.
- Repeat last step until whole data gets converted into text.
- Join all the blocks of texts (text1, text2 … textn) as Text (Plain text).

7 Proposed Methodology

The overview of the proposed systems components and discussed in this section of document. It shows the complete working of proposed system along with its various components (Fig. 5).

Input File: This is the file which is to be encrypted. It can be of any format.

Compression: It is used to compress the input file to reduce its size. Its reduction rate depends on the redundancy present in the file. This causes the size of cipher text to be lesser than that of plain text. Compression algorithms reduce the redundancy in data and decrease the amount of storage needed to store the data [5].

MD5: It is used for two purposes, i.e. to check the integrity of data through the creation of 128-bit hash function and to produce a secret key for encryption. The compressed file is operated with MD5 [6] algorithm to produce it hash function of 128-bit. This function is used as secret key to encrypt as well as decrypt the data as well as a data integrity checker.

Key (*16 bytes*): The 128-bit key generated from MD5 algorithm is used as secret key to encrypt the data.

Extraction: The compressed file is further extracted into its equivalent byte code. This makes the system independent from file format.

Fragmentation: The byte code extracted is further fragmented into data chunks each of 16 byte.

XOR Passes: In this step, several XOR passes are applied to the data chunks which are fragmented earlier. In first phase, first data chunk of 16 byte is XOR with hash function (16 byte) from MD5. The result of this operation is saved as Result1. In the next phase, the second chunk of data is ex or with the Result1 to get Result2. In this

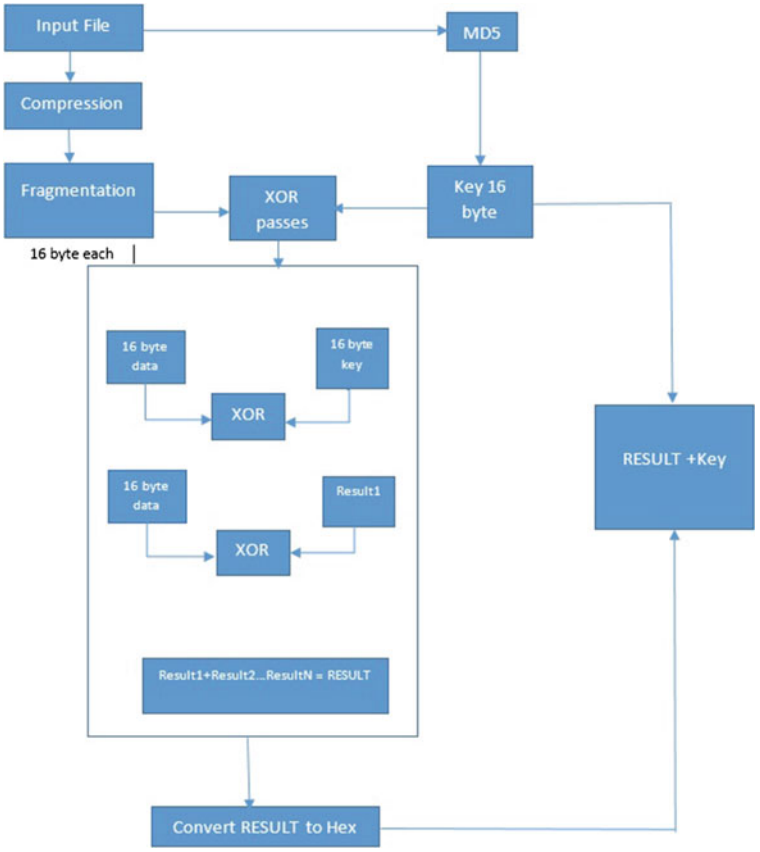


Fig. 5 Flow chart

way, this whole procedure is repeated till we get the last result chunk, i.e. ResultN. In the end, this set of result chunks along with hash function is set over the network to receiver. In this phase, the 16-byte data blocks and 16-byte key blocks are treated using XOR operator. The resultant of each pass is input for new 16-byte data, and the process goes on until whole data is processed. The combination of result of each phase is combined to form final result.

Hex Code: In this step, all the chunks of results (Result1, Result2 … ResultN) are combined and converted into hex format.

Data Transmission: In this step, the entire results of XOR passes are organized into one file and then the generated hash code are added to data file and send to the receiver end.

8 Result Analysis

In order to evaluate the results of proposed system, we are comparing it with a standard method which is data encryption standard (DES). In order to show the above results in a comparative manner, three parameters are used, i.e. memory used during encryption, time taken and difference in file size due to compression:

(1) *Memory Used*: This parameter shows the memory used by two methods in executing the same files. It deals with the space complexity of proposed algorithm in comparison with standard method. This comparison is shown in Fig. 6 (Table 1).

(2) *Time Taken*: This parameter shows the time taken by two methods in executing the same files. It deals with the time complexity of proposed algorithm in comparison with standard method. This comparison is shown in Table 2 (Fig. 7).

(3) *File Size* (*Cipher text*): This parameter shows the size of file produced as cipher text by two methods in executing the same file. As proposed method is using compression, hence size of cipher text is lesser than in case of standard method. This comparison is shown in Table 3 (Fig. 8).

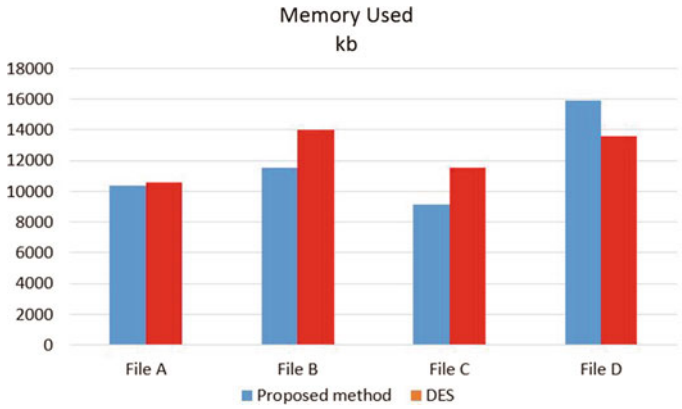


Fig. 6 Memory used comparison

Table 1 File size (kb)

File	Proposed method	Base method	Difference
File A	10357.0	12541.0	2184.0
File B	11508.0	13960.0	2492.0
File C	9115.0	11526.0	2411.0
File D	15893.0	13577.0	2316

Table 2 File size (kb) comparison

File	Proposed method	Base method	Difference
File A	31.0	78.0	47
File B	20.0	60.0	40
File C	35.0	82.0	47
File D	19.0	61.0	42

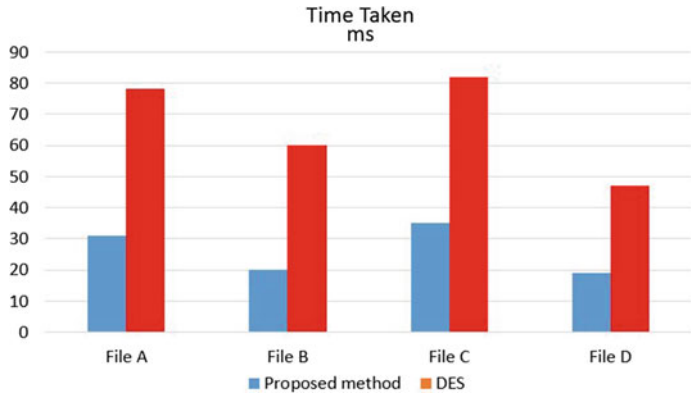


Fig. 7 Time taken (ms)

Table 3 File size (kb)

File	Proposed method	Base method	Difference
File A	899.847	967.075	67.228
File B	1388.550	1488.937	100.338
File C	1109.044	1028.850	80.194
File D	961.565	1031.612	70.047

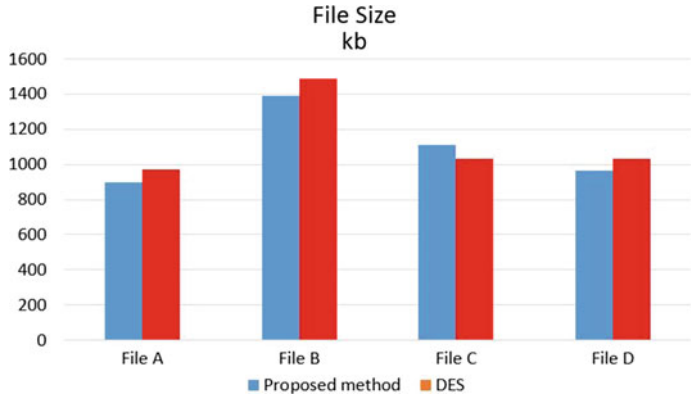


Fig. 8 File size comparison

9 Conclusion

From the results, it is proven that the proposed system is better as compared to DES; hence, it can be concluded that the proposed system is more secure and efficient to use. It consumes less memory as well as time; hence, it can be used in mobile devices where there is shortage of resources. It is also very difficult to break the algorithm as XOR passes to make the data almost unrecoverable for any attacker; hence, it can withstand any of the attacks. The proposed system is an excellent combination of compression and cryptography to reduce the size of data being sent and to provide it proper security mechanism over network. Proposed system fulfils the requirements for a secure and efficient data transmission.

References

1. B. Jasuja, A. Pandya, Crypto-compression system: an integrated approach using stream cipher cryptography and entropy encoding. Int. J. Comput. Appl. **116**(21), 34 (2015) (0975-8887)
2. T.M. Mahmoud, B.A. Abdel-Latef, A.A. Ahmed, A.M. Mahfouz, Hybrid compression encryption technique for securing SMS
3. K. Ilanthenral, K.S. Easwarakumar, HexiMcEliece Public Key Cryptosystem Department of Computer Science and Engineering, Anna University, Chennai 600 025, India
4. V. Gupta (M.Tech), G. Singh (Head CSE), R. Gupta (IT SSSIST Sehore), Advance cryptography algorithm for improving data security
5. T.M. Mahmoud, B.A. Abdel-Latef, A.A. Ahmed, A.M. Mahfouz, Hybrid compression encryption technique for securing SMS. Int. J. Comput. Sci. Secur. (IJCSS) **3**(6)
6. MD5, http://cs.sjsu.edu/faculty/stamp/crypto/PowerPoint_PDF/16_MD5.pdf
7. V. Gupta, G. Singh, R. Gupta, Advance cryptography algorithm for improving data security. Int. J. Adv. Res. Comput. Sci. Softw. Eng. **2**(1) (2012)
8. I. Mavridis, G. Pangalos, Security issues in a mobile computing paradigm, Informatics Laboratory, Computers Division, Faculty of Technology Aristotle University of Thessaloniki, Thessaloniki 540 06, Greece
9. A. Apoorva, P. Singla, A review of information sharing through shared key cryptography. Int. J. Res. Eng. Technol. Manage, ISSN 2347-7539

Development of More Secure and Time Efficient Encryption Method

Vinod Raghuvanshi, Pradeep Mewada and Praneet Saurabh

Abstract Data in the network is not secure and can be accessed by unauthenticated and unauthorized users. So to make data secure in the network, various encryption algorithms are used. Security along with confidentiality of the data is to be maintained in the network. Encryption is the technique by which data is encoded and sent to the receiver, where it gets decoded back to the plaintext. Encryption algorithm or technique should take minimum time and space for the execution. There are various encryption algorithms which are frequently used and some of them are high in measure of time and space complexity. This paper presents a more Secure and Time Efficient Encryption Method (STEEM) which is concentrating on high security of the encrypted text along with the time efficiency. This paper shows the working flows of the proposed work which would be solely responsible for achieving more efficiency on prime parameters.

Keywords Encryption · Data security · Block ciphering · Avalanche effect

1 Introduction

Cryptography is the branch of computer science which deals with the encryption and decryption of sensitive data which needs to be kept secret. Encryption is the process in cryptography which deals with the coding of data in such a way that it cannot be read or access by unauthenticated party. It does not prevent interface by itself, but not allowed the message information to be intercepted by anyone. In encryption, the

V. Raghuvanshi (✉) · P. Mewada (✉)
Technocrats Institute of Technology (Advance), Bhopal 462021, Madhya Pradesh, India
e-mail: vinodraghuvanshi99@gmail.com

P. Mewada
e-mail: pradeepmewada07@gmail.com

P. Saurabh
Technocrats Institute of Technology, Bhopal 462021, Madhya Pradesh, India
e-mail: praneetsaurabh@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_32

plaintext also refers to as text file or any document which is converted into ciphertext after using encryption algorithm. This ciphertext now cannot be read by anyone unless not cracked by any decryption algorithm. Pseudorandom key encryption is generally used for encryption key in encryption scheme. It is based on the principle where the message without decryption key can be decrypted. Large computational resources and proficiency are required for the well-designed encryption. So that only the authorized and authenticate party can access and decrypt the message received from the originator [1, 2]. The main objective of the encryption is to provide access data to the authorized person with the help of key needed for decryption of the message or plaintext file received. It will also exclude someone who is unauthorized as the key required for decrypting the data is not present [3]. Confidentiality itself can be protected by the encryption, but some other techniques are also used for the integrity and authentication of the data or information like digital signature and message authentication code (MAC).

1.1 Types of Encryption

Key generation is the most important method for encryption and decryption of the plaintext file. In cryptography, the encryption as well as the decryption can be done by the following two types which are mentioned below [4].

- Symmetric Key Encryption/Private Key and
- Asymmetric Key Encryption/Public Key.

1.2 Symmetric Key Encryption/Private Key

The encryption of the data using various kind of encryption algorithm is done with the help of key. The key is used for the encryption as well as decryption of the plaintext file. In private key encryption, the encryption and decryption of the data are done using same key only which must be kept secret and only have the access to the sender and the receiver as shown in Fig. 1. Some of the symmetric key algorithms for encryptions are [5].

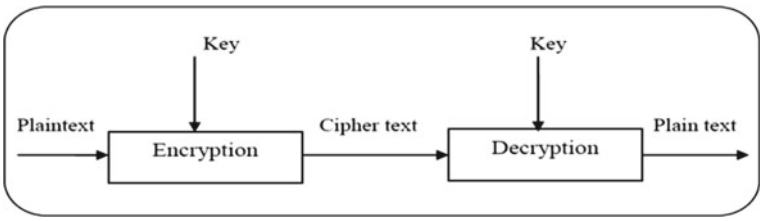


Fig. 1 Symmetric cryptosystem

1.2.1 Data Encryption Standard (DES)

It is a popular encryption technique and frequently used by many people these days. It is a block cipher encryption technique which uses same key for encryption and decryption of the plaintext file. The block size of DES is of 64 bits and whereas the key size is of 56 bits, respectively. It takes 16 rounds of mathematical operations to be performed also the permutations. This permutation at every round is added to the previous bit sent for the computation and the final key is generated.

1.2.2 Triple Data Encryption Standard (3DES/TDES)

It is an advance type of data encryption model with variable key size of 56,112, and 168 bits in which the encryption is done by following the pattern which is in form like Encrypt-Decrypt-Encrypt. In this technique, the plaintext is encrypted using encryption algorithm and produces cipher. This cipher is again encrypted and produces the next ciphertext which is sent to the receiver. In this way, it is more reliable and secure but very much time-consuming.

1.2.3 Advance Encryption Standard (AES)

It is a symmetric block cipher developed in the year 1998 [6]. It is a block cipher which is having the key length of 128, 1192, and 256 bits. In each cipher is encrypting in the block of 128 bits using the variable length key. It is also based on the design method called permutation and substitution method for the computation of the secret key. It takes 10, 12, and 14 rounds of computation according to the length of the key used.

1.2.4 Blowfish

It was built and designed by the Bruce Schneider in the year 1990. It is a symmetric key encryption block cipher. The block size used in this technique is of 64 bits, and the key size for this technique is variable ranges from 32 to 448 bits. It is the fastest technique among all the other technique. This technique usually does a key expansion and data encryption.

1.3 Asymmetric Key Encryption

In asymmetric key encryption technique, the data or the plaintext file can be encrypted by using anyone's public key and it can be decrypted only by the receiver's private key. The public and private key pair generated by receiver must be provided to the sender for the computational process of the algorithm as mentioned in Fig. 2. Following are some of the asymmetric key encryption techniques or algorithms [7].

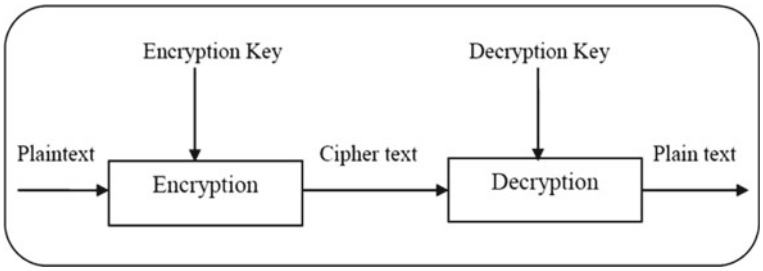


Fig. 2 Asymmetric cryptosystem

1.3.1 RSA Algorithm

This asymmetric public key encryption algorithm named after Ron Rivest, Adi Shamir, and Leonard Adleman (RSA), the founder and developer of the set of rules inside the year 1977 at MIT. It is used for statistics transmission. The set of rules uses two one of a kind keys for encryption and decryption. It uses the general public key of the receiver to encrypt the apparent text document, and receiver makes use of its personal key to decrypt the facts. On this algorithm, the private key of the receiver must be kept mystery.

1.3.2 Digital Signature

It is an asymmetric key encryption technique in which the receiver knows the sender of the message. This technique is used to ensure the receiver that the message is not repudiated by the sender. The major use of this technique is for purchasing the electronic shares and the receiver knows and ensures who has purchased. The main drawback of the digital signature is that it does not provide the confidentiality to the data. Using the private signing key, the message is being signed and sent to the receiver.

1.4 Types of Attacks

There are various types of attacks which could take place on the text data. Some of those are going to discuss hereafter:

1.4.1 Virus Assault

A virus is an executable program that when execute in the system network damages many different components and systems in the network. It can destroy hard disk and

processors. It also sometimes utilizes memory and brings down the performance of the system. So the encrypted data might get sometime affected by the virus in the system and get damaged [8].

1.4.2 Security Threats

There are many security threats in the network system by which data can be accessed by the unauthorized party so the encryption can provide security to the data up to and extend. Some of the most important security threats are distributed denial, denial of services, illegal way into the network property and data, uncontrolled Internet access, and accidental erasure of the record [9].

1.4.3 Unauthorized Access

There are many intruders which will have the unauthorized access to the plaintext file or data into the network for which the encryption mechanism is one of the techniques by which the unauthorized access to the data can be prevented. The data must be access by the authenticated user in the network [10].

1.4.4 Data Stealing/Cryptography Attack

By using the good encryption algorithm, the loss of the data or information in the network can be avoided. The encryption algorithm which uses 128 and 256 bits security encryption techniques are more secure to be used for data integrity. When the data is transferred using FTP program, if the encryption algorithm is used, it cannot be read by anyone [11]. Rest of the paper is organized in the following manner; Sect. 2 presents the related work while Sect. 3 introduces the proposed work. Section 4 explains the experimental results and analysis while Sect. 5 concludes the paper.

2 Related Work

To provide security to the data and confidentiality, use of cryptography is must. It means to transfer the data or information across the network which must be unreadable and cannot be accessed by unauthorized party. For this reason many algorithms like DES, TDES, and blowfish are used to protect sensitive data. These algorithms are then compared on the basis of avalanche effect [11]. By the method DNA-Genetic Encryption Technique (D-GET), the data can be more secured as the binaries and the digital data is converted into the DNA sequence. In this technique reshaping, crossover, mutation and then again reshaping are done. This method can be defined as a multi-protection system against different type of attacks. The new symmetric

key encryption algorithm in which the complexity is reduced is proposed and named AMEA that is AM encryption algorithm. The encryption algorithm should use less than or equal to the original file size in the disk space. The two most important parameters of any algorithm are time and space on which bases the complexity of the algorithm is measured. The algorithm should take minimum time to execute and perform the desired encryption. The space complexity of existing algorithms is very high. This algorithm is has a random key selection feature which provides better security to the data [12]. The algorithm was getting published publically for the comments after review of government undergoing for acceptability as the federal standard. It proposed a new key generation algorithm based on palm print which is used for encryption and decryption of an image. In this paper, they present a data encryption/decryption scheme based on bit XOR method [13]. The salient features of the proposed asymmetric encryption scheme can be summarized as: (a) Lossless encryption. (b) Less computational complexity. (c) Convenient realization. (d) Choosing a suitable size of matrix according to the size of data. (e) Encryption/decryption scheme uses integer arithmetic and logic operations. [14]. Recently, bio-inspired advances [15] also gained attention in realizing different goals on this domain [16–18]. To protect the data from unauthorized users, encryption and decryption algorithms along with effective and unique mechanism are required that must be capable of providing necessary safety and privacy for our online transactions and other communication. Work in this area has for the most part centered on search criteria consisting of a one keyword. The next section deals with the encryption algorithm which is proposed. This section is divided into three main parts. First part deals with architecture of the proposed work while second part deals with algorithm. Third part discusses strength of the proposed work.

3 Proposed Work

In this section, we propose a new method for effective encryption, namely more Secure and Time Efficient Encryption Method (STEEM), which is concentrating on high security of the encrypted text along with the time efficiency. Figure 3 shows the flow chart of the STEEM algorithm.

In this method, initially some random number is generated by using symmetric key, and then we will try to find out five random keys with the help of generated numbers. Calculation of message length is done through multiply it with key length. The data is then shuffled using key bit length. This shuffling is continued till all message blocks are ciphered.

3.1 Algorithm of Our Proposed STEEM Method

Step 1: Start,
Step 2: Initialize input parameters to generate random numbers (n).

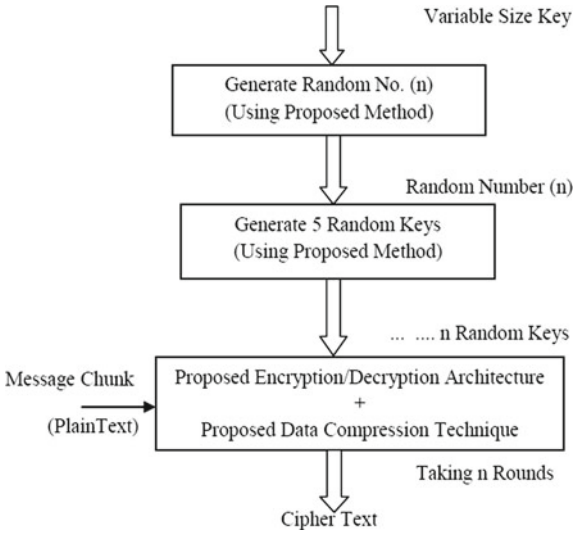


Fig. 3 Architecture of proposed STEEM method

Step 3: Now, the generated 'n' random number will be formulated to create five random keys.
Step 4: Calculate length of message and multiplying it with key length.
Step 5: Divide the whole message in key length bits of 'k' blocks.
Step 6: Shuffle the data for each key bits length by using STEEM method.
Step 7: Repeat **Step 6** for the next message blocks till all message blocks are ciphered.

3.2 Strength of Proposed Algorithm

By using the concept of random multiple keys in our proposed STEEM method, we increased the overall strength of standard encryption technique. In this technique, variable length key is used which makes this technique robust against any attack, and it can also be feasible for ad hoc type of networks because of low battery consumption and less CPU usages. In order to provide higher level of confidentiality and integrity, our proposed method provides an ensemble encryption technique which gives better performance in terms of efficiency. The below section clearly demonstrates the experimentation results along with result analysis and comparisons with exiting techniques.

4 Experimentation and Result Analysis

This section is divided into two parts: First part concentrates on the configuration of the system on which various experiments are performed along with the properties of dataset.

4.1 System Configuration

System configuration always affects, and it gives special effects on execution time while data size will affect both execution time and avalanche effect. The minimum system requirement to run our proposed method is given in Table 1.

Datasets are basically collection of data in which various types of attributes and features are included to represent some specific properties of that data. To evaluate the performance of our proposed STEEM method, we are considering three different samples of dataset along with their sizes. Table 2 and Fig. 4 show all three samples of dataset along with their sizes that are considered to calculate avalanche effect and execution time.

Table 1 System configuration

Particulars	Specifications
Processor	Core-i5
RAM	4 GB
OS	Win7-64 bit

Table 2 Size of various samples in dataset

Sample	Size (KB)
Sample-1	5
Sample-2	10
Sample-3	15

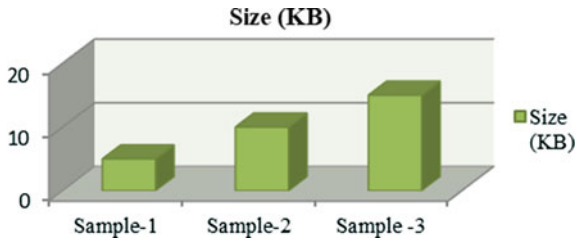


Fig. 4 Different dataset with various sizes

Table 3 Avalanche effect of base and proposed algorithm

Avalanche effect		
Sample	Existing method	Proposed STEEM method
Sample-1	26.38	49.98
Sample-2	26.38	49.50
Sample-3	26.38	49.72

Table 4 Comparative analysis between base and proposed algorithm

File size in KB	Execution time in second	
Sample (s)	Existing method encryption time	Proposed STEEM method encryption time
Sample-1	0.156	0.109
Sample-2	0.546	0.187
Sample-3	2.184	0.670

4.2 Performance Analysis

It is done on two parameters, as follows:

4.2.1 Avalanche Effect

It is quantity which shows the effectiveness of the work in terms of security. The avalanche effect is satisfied if: The output changes significantly as we perform any change in the input parameters. In "quality" block ciphers, such type of changes can be performed in either the key or the plaintext that should cause a strong change in the ciphertext. The above features can be implemented using number of iterations, and hence, every bit of the output should depend on every bit of the input before the algorithm gets closed (Table 3).

4.2.2 Execution Time

It is quantity which shows the effectiveness of the work in terms of time required to execute (Table 4).

5 Conclusion

Electronic documents which are in the form of text file or images are used with the development of information technology. But these electronic data is not secure in the network as many intruders or unauthorized parties are accessing the data which need

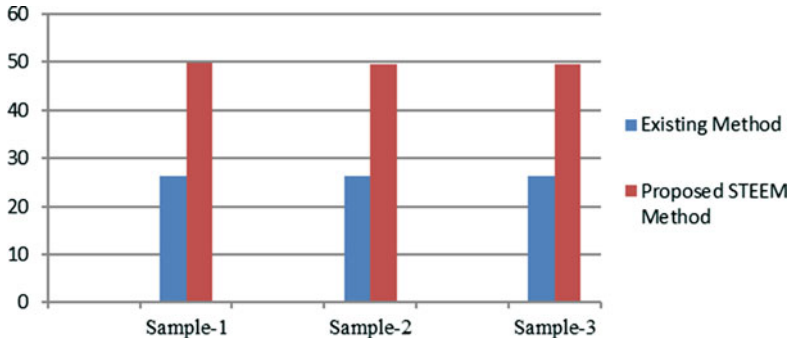


Fig. 5 Comparison of avalanche effect

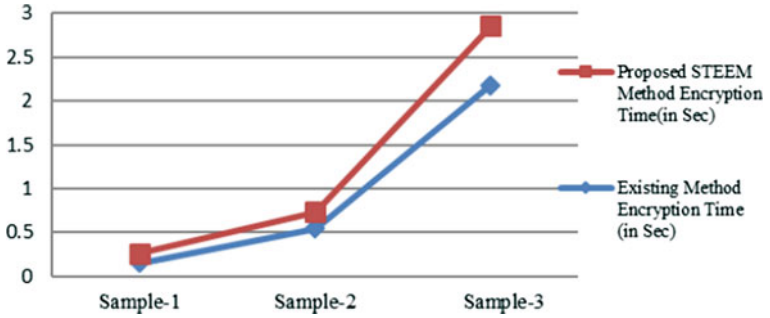


Fig. 6 Comparative analysis between existing and proposed algorithm execution time

to be kept secret and confidentiality of the data is being broken every day. In order to solve this problem, our proposed method improves the document security. This paper presents a more Secure and Time Efficient Encryption Method (STEEM) which is concentrating on high security of the encrypted text along with the time efficiency. Figures 5 and 6 along with the respective table clearly show that the efficiency of the proposed work on the scale of security and time consumption is much more than the existing work. This discussed work further can be enhanced and made for the image encryption too.

References

1. A. Odeh, R.S. Masadeh, A. Azzazi, A performance evaluation of common encryption techniques with secure watermark system (Sws). Int. J. Netw. Secur. Appl. (IJNSA) **7**(3), 31–38 (2015)
2. C. Li, Importance of digital encryption technology in internet security. Guide Bus. 296 (2011)
3. R. Dubey, A. Saxena, S. Gond, An innovative data security techniques using cryptography and steganographic techniques. Int. J. Comput. Sci. Inf. Technol. **6**(3), 2175–2182 (2015)

4. O.M.A. Al-Hazaimeh, A new approach for complex encrypting and decrypting data. Int. J. Comput. Netw. Commun. (IJCNC) **5**(2), 95−103 (2013)
5. Y.N. Goshwe, Data encryption and decryption using RSA algorithm in a network environment. IJCSNS Int. J. Comput. Sci. Netw. Secur. **13**(7), 9–13 (2013)
6. W. Mao, *Modern Cyptography Theory Practice* (Prentice Hall PTR, New Jersey, 2003)
7. S. William, *Cryptography and Network Security: Principles and Practice, Pearson Education, Inc*., 5th edn (Publishing as Prentice Hall, New Jersey, 2003)
8. G.L. Prakash, M. Prateek, I. Singh, Data encryption and decryption algorithms using key rotations for data security in cloud system. Int. J. Eng. Comput. Sci. **3**(4), 5215–5223 (2014), ISSN 2319-7242
9. J. Das, A study on modern cryptography and their security issues. Int. J. Emerg. Technol. Adv. Eng. **4**(10), 320–324 (2014) (ISSN 2250-2459, ISO 9001:2008 Certified Journal)
10. M. Koul, Authentication against man-in-the middle attack and honey encryption. Int. J. Adv. Res. Comput. Sci. Softw. Eng. **6**(8), 388–390 (2016)
11. M. Lin, An overview of session hijacking at the network and application levels, SANS Institute InfoSec Reading Room (2005)
12. H.C. Chou, H.C. Lee, C.W. Hsueh, F.P. Lai, Password cracking based on special keyboard patterns. Int. J. Innov. Comput. Inf. Control **8**(1(A)), 387–402 (2012)
13. M.A. Mobarhan, M.A. Mobarhan, A. Shahbahrami, Evaluation of security attacks on Umts authentication mechanism. Int. J. Netw. Secur. Appl. (IJNSA) **4**, 37–42 (2012)
14. J.S. Vitter, Arithmetic coding for data compression, researchgate.net (1994)
15. P. Saurabh, B. Verma, An efficient proactive artificial immune system based anomaly detection and prevention system. Expert Syst. Appl. **60**, 311–320 (2016)
16. P. Saurabh, B. Verma, Cooperative negative selection algorithm. Int. J. Comput. Appl. **95**(17), 27–32 (2014) (0975–8887)
17. P. Saurabh, B. Verma, S. Sharma, An immunity inspired anomaly detection system: a general framewor, in *7th BioInspired Computing: Theories & Applications, Springer* (2012), pp. 417–428
18. P. Saurabh, B. Verma, S. Sharma, *Biologically Inspired Computer Security System: The Way Ahead*, vol. 335 (CICS Springer, 2011), pp. 474–484

Design and Development of Cost Measurement Mechanism for Re-Engineering Project Using Function Point Analysis

Pooja Kumawat and Namarata Sharma

Abstract Software sizing is one of the necessary metrics for calculating the size, cost, duration, and effort of the project. It has a direct impact on development effort and project management. As the size of the project increases, accuracy is really important. There are a number of metrics available for size estimation, and it is very difficult to select the most suitable method for calculating the size of a software project. Function point analysis (FPA) is the most suitable method to estimate the size and complexity of the software. In this work, the FPA method is used for calculating the cost of re-engineering and reproduction applications. After applying the FPA method to the re-engineering and reproduction applications, a comparison graph is generated which compares the processing complexity, unadjusted function points (UFPs), and total function points.

Keywords Software size · Function point analysis · Lines of codes
Unadjusted function points · Processing complexity · Use case

1 Introduction

Software is important in every field such as in law, entertainment. Good software is that which permits us to get a better estimate of the time and costs of a project.

During the software development life cycle, the technique which is importantly used to ensure the quality and reliability of software is:

P. Kumawat (✉) · N. Sharma
Department of Computer Science and Engineering,
Sushila Devi Bansal College of Engineering, Indore, India
e-mail: poojakumawat.mit@gmail.com

N. Sharma
e-mail: nsharma@sdbce.ac.in

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_33

Software metrics: Metrics are measurement methodologies whose main objective is to estimate the size of software system and assist, as an indicator, the project management of software system development [1].

There are several metrics for size estimation, and it is very difficult to select the most appropriate method for the size of a software project.

Function point analysis is an efficient method for calculating the size of software. This work shows how FPA method is used for calculating the cost of re-engineering and reproduction applications.

1.1 Software Re-Engineering

New features can be added to existing systems, and then, the system is reconstructed for better use of it in future.

1.2 Reproduction

Reproduction is the development of the new application without using any existing classes or function. The cost of reproduction project is always high as compared to the re-engineering project.

2 Size Estimation Method

2.1 LOC

The LOC is typically characterized as: lines of code (LOC); this strategy checks every single line of the code, including clear lines and remarks.

Advantages

1. The LOC is constantly measured after the completion of the project.
2. LOC can be very effective in estimating the effort of the project.

Disadvantages

1. The LOC is always counted after the completion of application.
2. Contrast in Languages: The amount of effort needed to develop the application would be different for application that is built in different languages.

2.2 Use Case Points

Use Case Points (UCP) is one of the software estimation techniques which is used to measure the size of software with the help of use cases.

Advantages

1. UCP does not rely on the extent and experience of the group that executes those tasks.
2. UCP will be not difficult to utilize.

Disadvantages

1. UCP is useful when requirements are mentioned in the type of use cases.
2. There is no typical design for the use case.

2.3 Function Point Analysis

The function point concept was introduced by Alan J. Albrecht of IBM in 1979. In 1984, the International Function Point Users Group (IFPUG) was set up to clarify the rules, set standards, and promote their use and evolution [3]. FPA strategy is utilized for estimating the extent of the software. FPA measures the usefulness of the product. The premise of FPA is client's necessities and its sensible plan. FPA strategy is easy to utilize. Its initial step is the deciding client's necessities which are depicted in a product prerequisite detail (SRS).

Advantages

1. It can be calculated at the initial requirements phase.
2. Function point technique establishes effective communications between the customer and software developer.

3 Literature Survey

Let us look at the works that are already done by various researchers.

Alves et al. [1]: The authors used the function point analysis (FPA) method to estimate the size and complexity of the software system. The authors show the empirical study on two teams.

Low and Jeffery [2]: The authors show that the function point method is more accurate and consistent than lines of code method.

Behrens [3]: In this paper, the results described are consistent with Albrecht. The authors show that project size, platform, technology, and language are important parameters in determining the productivity of the system.

Standish [4]: In this paper, the author explores the software reuse. The author also shows that reusing of software components improves productivity.

Jeffery et al. [5]: In this research, the authors show the problem associated with calculating the function point count and effect of processing complexity adjustment on it.

Jacobson et al. [6]: The author shows that how an object-oriented method can be used to modernize the old system by using the re-engineering concept. The authors show the different cases of re-engineering.

4 Methodology

It was an attempt to overcome difficulties associated with lines of code as a measure of software size and to assist in developing a mechanism to predict effort associated with software development (Fig. 1).

4.1 Unadjusted Function Point

There are generally five major components of function place analysis which often capture the actual functionality in the application. These are generally:

External Inputs: Inputs are characterized as the information entering into the system.

External Outputs: Outputs are characterized as the data that is leaving the system.

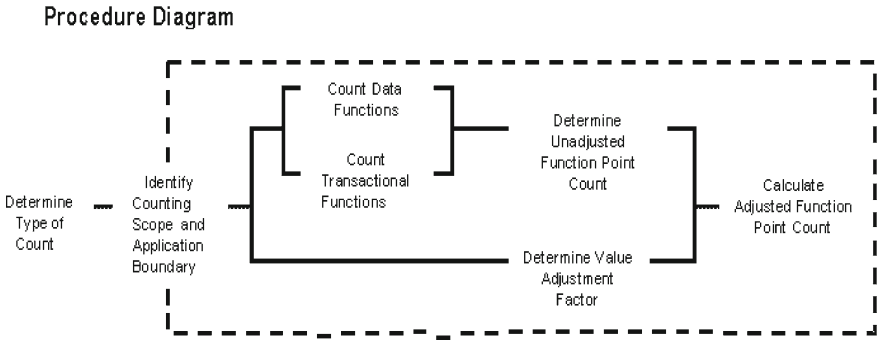


Fig. 1 Procedure diagram for function point calculator

Table 1 Function point counts

	Simple	Average	Complex
Inputs	2	4	6
Outputs	3	5	7
Files	5	10	15
Inquires	2	4	6
Interfaces	4	7	10

External Inquiries: Inquiries are requests for access to data. It brings information about recovery.

Internal Logical File (**ILF**): ILF is characterized as data held inside the system. Files are stored in the system and utilized inside the framework.

External Interface File (**EIF**): This record is outer to the framework. It is shared by the other framework. It is just utilized by the framework, not kept up by the framework.

4.2 Rating the Transactional and Data Function Types

All of the identified components are allocated a score (as minimal, average, and high). These function point counts are then weighed (multiplied) by their degree of complexity (Table 1).

4.3 General System Characteristics (GSCs)

The value adjustment aspect (VAF) is usually calculated based on 14 basic system characteristics that rate the overall functionality of the application getting counted.

0: Not present, or no influence; **1**: Incidental influence; **2**: Moderate influence
3: Average influence; **4**: Significant influence; **5**: Strong influence throughout as follows:

14 Basic System Characteristics

The basic system characteristics are data communication, distributed data processing, performance heavily used configuration, operational ease, reusability, complex processing, installation ease, end-user efficiency, multiple sites online, update online data entry, facilitate change.

Once all the GSCs have been rated, total degrees of influence (TDI) are obtained by summing up all the ratings. Now, value adjustment factor is calculated using the formula:

$$\mathbf{VAF = 0.65 + TDI/100}$$

4.4 Final FP Count

The final function position count might be calculated while using the formula (Fig. 2):

	Complexity			
Description	Low	Medium	High	Total
Inputs	___ x 3	___ x 4	___ x 6	___
Outputs	___ x 4	___ x 5	___ x 7	___
Queries	___ x 3	___ x 4	___ x 6	___
Files	___ x 7	___ x 10	___ x 15	___
Program Interfaces	___ x 5	___ x 7	___ x 10	___

Total Unadjusted Function Points (TUFP): _____

(0=no effect on processing complexity; 5=great effect on processing complexity)

	0-5
Data communications	___
Heavily use configuration	___
Transaction rate	___
End-user efficiency	___
Complex processing	___
Installation ease	___
Multiple sites	___
Performance	___
Distributed functions	___
On-line data entry	___
On-line update	___
Reusability	___
Operational ease	___
Extensibility	___

Processing Complexity (PC): _____

Adjusted Processing Complexity (PCA) = 0.65 + (0.01 * _____)

Total Adjusted Function Points (TAFP): _____ * _____ = _____

Fig. 2 Proposed method for function point analysis

FP = Unadjusted Function Point count (UFP) * Value Adjustment Factor (VAF)

The proposed method is used for the re-engineering and the reproduction applications (Fig. 3).

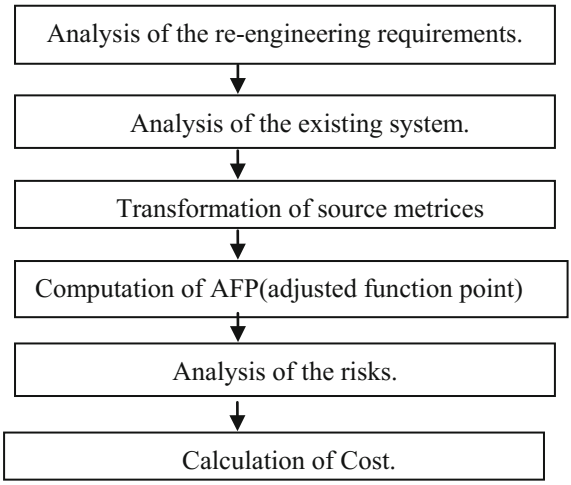


Fig. 3 Proposed method used for re-engineering application

5 Experimental Result

The proposed tool calculates the function point of three scenarios: for software engineering, re-engineering, and reproduction. The graph represents function point analysis between re-engineering and reproduction.

We calculated function point for a banking system which has different modules like login, registration, tax, loan, and salary of employee. The project is divided into three datasets. Dataset 1 consists of the following modules—Tax Investment, TaxBankAndPedi, and DoTxnHome. Dataset 2 consists of the following modules—TxnPO, TxnProerty, and TxnSalary. Dataset 3 consists of the following modules—TxnLIC, TxnLoan, and TxnProperty (Figs. 4 and 5).

The main aim of the proposed work is to compare the result of re-engineering and reproduction applications using the function point analysis method. From the graph, it is clear that the cost of re-engineering application is always less than the reproduction application.

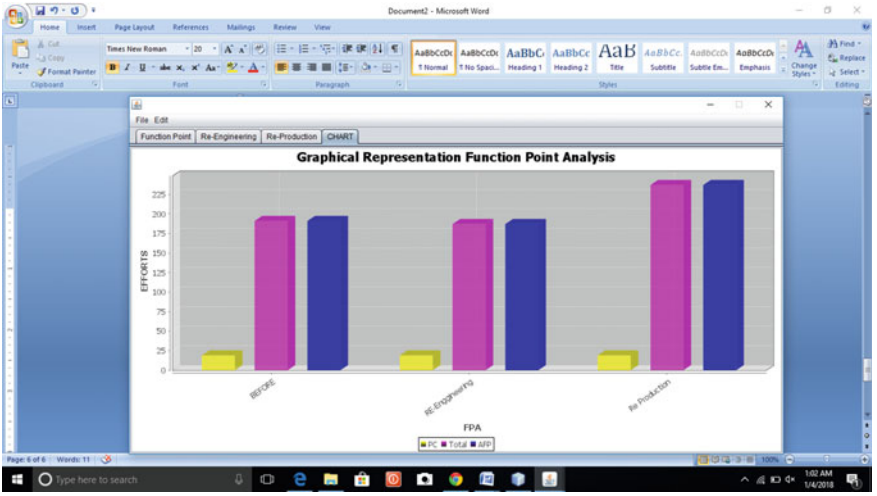


Fig. 4 Comparison graph that compares the processing complexity, value adjustment factor, adjusted function points for all the three applications

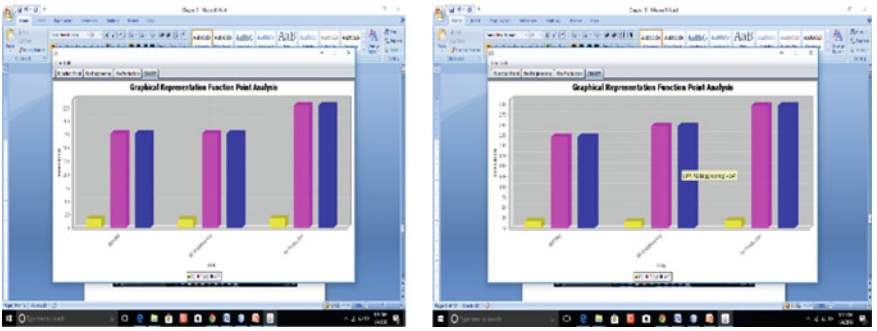


Fig. 5 Comparison graph that compares the processing complexity, value adjustment factor, adjusted function points for all the three applications

6 Conclusion

In this work, the FPA method is used for calculating lines of code, processing complexity, total function point of the re-engineering and the reproduction applications. It is directly connected to user requirements and functionality. It can be calculated at the initial requirements phase. It is better than LOC and use case-based method.

We developed a tool which is applicable to software development using any software life cycle model. The system changes, and hence, maintenance must be performed in order to correct faults, improve the design, implement enhancement, interface with other systems, adoption of environment (different hardware, software, system features, etc.), migrate legacy software by new software.

The proposed work is done in Java using NetBeans. FPA method is used for both the re-engineering and reproduction applications. After that, results are compared with both the applications.

The obtained result demonstrates that the cost of re-engineering application is less than the reproduction application. Thus, the proposed method is effective in calculating the various parameters of function point analysis.

References

1. L.M. Alves, S. Oliveira, P. Ribeiro, R.J. Machado, An empirical study on the estimation of size and complexity of software applications with function point analysis, in *14 International Conference on Computational Science and Its Applications* (2014)
2. G.C. Low, D.R. Jeffery, Function points in the estimation and evaluation of the software process. IEEE Trans. Softw. Eng. **16**(1) (1990)
3. C.A. Behrens, Measuring the productivity of computer systems development activities with function points. IEEE Trans. Softw. Eng. **6** (1983)
4. T.A. Standish, An essay on software reuse. IEEE Trans. Softw. Eng. **SE-10**(5) (1984)
5. D.R. Jeffery, G.C. Low, M. Barnes, A comparison of function point counting techniques. IEEE Trans. Softw. Eng. **19**(5) (1993)
6. I. Jacobson, F. Lindstriim, A.B. Torshamnsgatan, Re-engineering of old systems to an object-oriented architecture

Noninvasive Gluco Pulse Watch

Varun Sood, Manisha Choudhary, Aviral Malay and Rachit Patel

Abstract With the help of this watch, we are re-defining not only the way sugar levels in the body can be measured but also how technology can be leveraged to take immediate actions on that result. This innovation has not been prototype yet across the globe. Over the past decade, there have been significant deaths due to diabetes and the major reason for this is there is no simpler technique to monitor it. The doctors' guidance and medication is not provided at the required time resulting in loss of life. Noninvasive methods to monitor the pulse rate and glucose level are nowhere to be found, and this is the area in which the watch steps in.

Keywords Arduino UNO · Internet of things · pH sensing
Pulse sensing and biochemical glucose enzyme sensor
Universal asynchronous receiver and transmitter

1 Introduction

Health is and always been the primary concern of any individual, and this ever-changing lifestyle causes harm to our body internally as well as externally. Owing to this, there has been a significant rise in the number of people suffering from diabetes. Improper management of diabetes has resulted in numerous deaths all around the world.

V. Sood (✉) · M. Choudhary · A. Malay · R. Patel
ABES Institute of Technology, Ghaziabad, Uttar Pradesh, India
e-mail: varunsood65@gmail.com

M. Choudhary
e-mail: manishachoudhary6696@gmail.com

A. Malay
e-mail: aviral18@gmail.com

R. Patel
e-mail: rachit05081gece@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_34

Negligence toward diabetes and high blood pressure should be minimized, and this watch would certainly help in doing so because of its multi-featured and simpler functionality. This watch can measure pulse rate along with sugar or glucose level of a person body with sweat.

The measured values can be monitored easily by the patient and the doctor from anywhere on a constant basis with the help of the Internet of things (IoT) feature. In case of any abnormality, an alert will be sent to the linked contacts. It will indeed revolutionize the field of medical sciences and technology.

The core technical innovation is the combination of the pulse sensor and pH sensor (for sweat-based glucose monitoring) on a single PCB layout. Providing feature of IOT by which doctors can remotely monitor the measured values of the patients via the Internet.

1.1 Key Features

- To reduce the turnaround time of the patient–doctor interaction, thereby ensuring timely treatment.
- This redefined approach of treatment has the potential to save a lot of lives which otherwise are lost to diabetes in the world.
- This will introduce a painless and accurate means to measure the glucose level using sweat and pulse rate through a single module, thereby reducing the trauma.
- Measurements, prescriptions, and impact of the treatment can all be monitored and acted upon in real time irrespective of the physical location of the patient and the doctor.

1.2 Problems Faced Today by the Patients

- No noninvasive technique available to measure glucose level.
- No real-time, automatic, technology-based remedial intervention are available for the common person.
- Reduce the degree of 'Ignorance or negligence' prevailing due to the lack of technology connects between the patient and doctor. One of the leading causes of death all around the world.
- No real-time access to the measured values for doctors at remote places.

2 Description

The module is a multi-featured watch with lots of applications. The aspects which it covers include medical and electronic implications. It consists of a main controlling unit which is a microcontroller Arduino UNO, performing the specified functions

with the help of various sensors. A pulse sensor is interfaced with Arduino UNO to measure the pulse rate of an individual. The phenomenon used by it is sending of high-intensity light through our skin of which some is absorbed by the capillary tissue while some is reflected back to the photosensitive square which calculates the pulse rate giving the corresponding analog value to Arduino UNO. pH sensor is used because it has been studied in case of diabetes the pH of the sweat of the person is decreased from the normal value which ranges from 4.5 to 6.7 [1, 2]. pH sensor measures the pH of sweat and sends the corresponding value to pH meter which processes and serially transfers the value to Arduino UNO. Monitoring of pulse rate and blood glucose level can be done so easily and efficiently that to anytime and from anywhere by both the doctor and the patient. The IoT feature provides the liberty for these values to be checked from any part of the world. Such prolific combination of all the above domains has never been before.

2.1 Block Diagram

See Fig. 1.

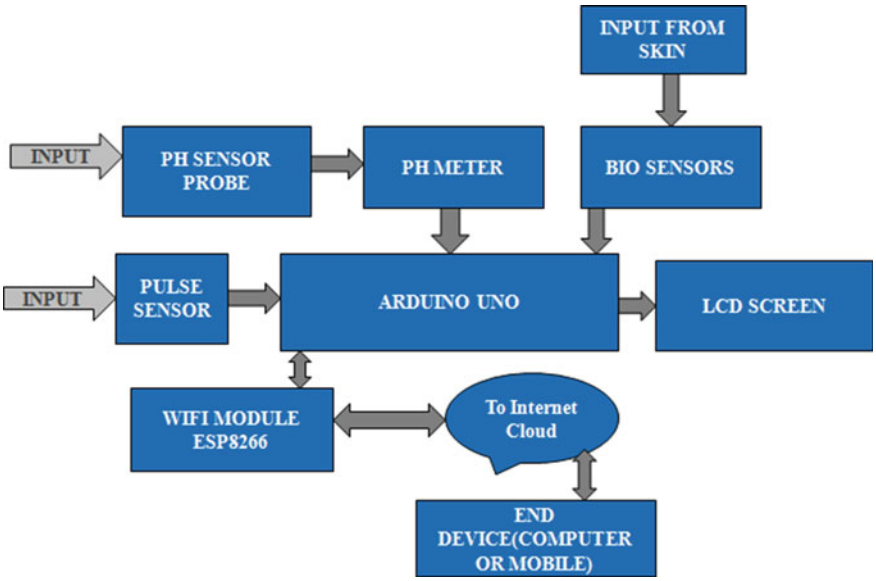


Fig. 1 Block diagram of the system

2.2 Resources Used

(a) **Microcontroller Arduino UNO ATMEGA328**: The Arduino UNO is Atmel 8-bit AVR RISC-based microcontroller. It is combined with

1. 32 kB ISP flash memory with read-while-write capabilities
2. 1 kB EEPROM, 2 kB SRAM
3. 23 general purpose I/O lines
4. 32 general purpose working registers
5. Three flexible timer/counters with compare modes
6. Internal and external interrupts
7. Serial programmable USARTa byte-oriented 2-wire serial interface
8. 6-channel 10-bit A/D converter.

(b) **pH Sensor Probe and pH Meter Kit**: **pH** means the power of hydrogen. It is used to measure the hydrogen ion concentration in the body. A pH sensor loop is made up of

1. The pH sensor
2. A measuring electrode
3. A reference electrode
4. Temperature sensor and an analyzer or transmitter.

(c) **Pulse Sensor**: The pulse sensor consists of three pins as red—5 V, black is GND, and purple is analog output of pulse sensor. This analog output of sensor is connected to A0 (analog input) pin of Arduino UNO.

(d) **Wi-Fi Module ESP8266**: This Wi-Fi module is a self-contained SOC and has integrated TCP/IP protocol stack which can give any microcontroller access to your Wi-Fi network. Every ESP8266 module comes pre-programmed with

1. An AT command set firmware.

(e) **LCD Screen (16 × 2)**: A 16 × 2 LCD display is used commonly in different circuits and devices. In 16 × 2 LCD, 16 means it can display per line 16 characters and 2 means there are 2 lines.

3 Working

The input is given to pH Sensor probe from the human sweat. In case of raised glucose level, the pH of the sweat of the person is decreased from the normal value which ranges from (4.5 to 6.7) [3, 4]. pH sensor measures the pH of sweat and sends the corresponding value to pH meter which processes and serially transfers the analog value to Arduino via UART. The pulse sensor works on the principle of photoplethysmography. It is used to measure the changing value in the volume of blood by any organ present in the body which leads to change in the light intensity through the organ (a vascular region). When the pulse rate is analyzed, the timing is more important of the pulses.

The heart pulses decide the flow of the blood volume, and as light is absorbed by blood, the signal pulses recorded are equal to the number of the heartbeat pulses. The basic heartbeat sensor consists of a light-emitting diode and one detector as of a light-detecting resistor or a photodiode. The variation in the flow of blood to different regions of the body is caused by the heartbeat pulses [5, 6].

The microcontroller further checks whether the pulse rate is in normal or abnormal range and displays the output accordingly. The pulse sensor's power supply pins are connected to the Arduino board supply pin as red is 5 V, black is GND, and purple is A0 (analog input 0). The pH meter is connected serially with Arduino UNO via UART. The measured analog value from pH meter is serially transmitted to Arduino UNO which further checks whether the value is in normal or abnormal range and displays the output accordingly. The Wi-Fi module ESP8266 is serially connected to Arduino UNO via UART. This module is used for setting the IoT feature. Through this module, the respective output of the pulse rate and glucose level from Arduino can be accessed from any part of the world by the patients and the doctors. In case of abnormality, an alert will be sent to the linked contacts. The corresponding output values of the pulse rate and glucose level measured can be seen on the LCD screen which is connected to the Arduino UNO [7–9].

4 Results

The pulse sensor has been implemented and shows accurate values for an adult to validate the results (Figs. 2 and 3).

The performance is better than conventional method.

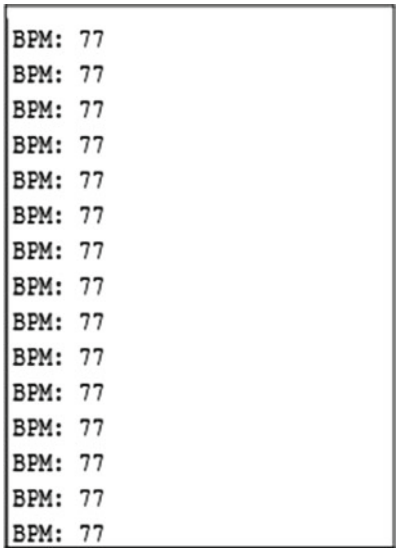


Fig. 2 Pulse sensor implementation result

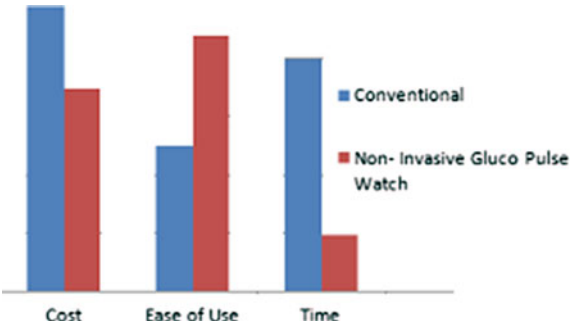


Fig. 3 Comparison

5 Conclusions and Future Scope

1. Reduced the turnaround time for the patient–doctor interaction.
2. Cost is reduced as compared to conventional methods.
3. Output is gathered within few seconds.

In future, an additional feature of measuring the body temperature can be cumulated. Along with this, power consumption and efficiency can be optimized.

References

1. M. McMillan, Blood glucose, in *Clinical Methods: The History, Physical, and Laboratory Examinations*, 3rd edn., ed. by H.K. Walker, W.D. Hall, J.W. Hurst (Emory University School of Medicine, Atlanta, Georgia, 1990)
2. H.P. Chase, Blood sugar (Glucose) testing, in *Understanding Diabetes*, 11th edn. (2006)
3. Diabetes Technol Ther **14**(5), 398–402 (2012), https://doi.org/10.1089/dia.2011.0262. Epub 2012 Feb 29
4. Science Advances **3**(3), e1601314 (2017), https://doi.org/10.1126/sciadv.1601314 Hyunjae Lee, Changyeong Song, Department of Radiology, Seoul National University College of Medicine, Seoul 03080, Republic of Korea
5. T.G. Pickering, J.E. Hall, L.J. Appel et al., Recommendations for blood pressure measurement in humans and experimental animals part 1: blood pressure measurement in humans: a statement for professionals from the Subcommittee of Professional and Public Education of the American Heart Association Council on High Blood Pressure Research. Hypertension **45**(1), 142–161 (2005)
6. X.R. Ding, Y.T. Zhang, J. Liu et al., Continuous cuffless blood pressure estimation using pulse transit time and photoplethysmogram intensity ratio (2015)

7. L.M. Strambini, A. Longo, A. Diligenti, G. Barillaro, A minimally invasive microchip for transdermal injection/sampling applications. Lab Chip **12**(18), 3370–3379 (2012)
8. C.E. Prez, Time synchronisation in Arduino-based wireless sensor networks. Latin Am. Trans. IEEE (Revista IEEE America Latina) **13**(2), 455–461 (2015)
9. B. Roy, Platform-independent customizable UART, in *2012 Third International Conference on Intelligent Systems*

Multi-input Multi-output Self-learning-Based Control System

Aashish Phatak, Deepa Panicker, Priyank Verma, Mayuri Bhadra and Vinit Hegiste

Abstract We have proposed a machine learning-based approach for automation of control room operations. The central idea is to introduce machine learning in SCADA-based control rooms, to assist the operator. We are making use of gradient boosted decision tree (GBDT) to construct a learning model for decision making. We tested the performance of the GBDT against various decision tree (DT) algorithms using the classification accuracy on an industrial plant dataset provided by the Technology Development Division of Nuclear Recycle Group of Bhabha Atomic Research Center (BARC). The predictions given by the GBDT were then sent to a SCADA operator using MODBUS communication protocol over TCP/IP. The experimental results have proved that the proposed method can be useful in real-life large-sized plants where the data to be handled is very large and there is immense work pressure on the operator.

Keywords Machine learning · Control room · Data analysis · Ensemble algorithm Gradient boosted decision tree · SCADA · MODBUS · EPICS

A. Phatak (✉) · D. Panicker · P. Verma · M. Bhadra · V. Hegiste
Bachelor of Engineering Student in Electronics and Telecommunication Department, Rajiv Gandhi Institute of Technology, Juhu-Versova Link Road, Versova, Andheri (West), Mumbai 400053, Maharashtra, India
e-mail: aashish.phatak10@gmail.com

D. Panicker
e-mail: panickerdeepa1096@gmail.com

P. Verma
e-mail: pverma7525@gmail.com

M. Bhadra
e-mail: mayuri.bhadra96@gmail.com

V. Hegiste
e-mail: vinithegiste@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_35

1 Introduction

In this paper, we introduce machine learning for control room applications. The control process of any plant in an industry uses SCADA system, and these systems are used to monitor and control various processes in the plant. The SCADA system consists of field devices, which measure the various control parameters of the processes, PLCs, data acquisition system, and HMI. The operator uses the HMI to observe the status of the plant. Within the confines of the control room, an operator sends various commands through the SCADA system to change the status of certain plant processes.

In current applications, the PLCs can only detect the off-normal conditions when a particular set point is crossed. It, however, does not take into account the similarity in trends but instead takes into account the equality of values. Thus, operator is only aware when a certain set point is reached and not how that event has occurred. For a large plant, where the number of processes are more, it is very difficult to monitor each and every abnormality that takes place in the process and take necessary actions in the plant. However, machine learning based system could play an important role to assist the operator, where the inputs coming from the field devices will be studied by the machine learning algorithm and it will continuously predict the control measure to be performed, thus acting as suggestions to the operator, as to what necessary action needs to be taken.

Our work is focused to assist the operator in his decision-making process and not to replace the operator or the existing SCADA system. It will work along with the existing SCADA system.

2 Data Analysis

The dataset [1] provided during the Smart India Hackathon 2017 consists of nine controlled input attributes labeled as N1, N2, N3, …, N8, N9 and three controlled output (target) attributes labeled as M1, M2, M3 one in each column. Data analysis must be done to accomplish several tasks such as to view data distributions, to identify skewed predictors, and to identify outliers. Some of the data visualization techniques are histograms, box plots, correlation matrix, etc.

Histogram is used to identify the skewed attributes. After observing the histograms of the inputs, it was observed that the input N2 has a plateau distribution, and the inputs N4, N7 and N8 have a right-skewed distribution, whereas N3 and N9 have left-skewed distribution.

The correlation matrix shown below in Fig. 1 is a table showing correlation between sets of variables. Each input Ni is correlated with each of the other inputs. This allowed us to see which pairs have the highest correlation. The blank space in the matrix indicates that the corresponding inputs are not at all correlated. The thin straight line indicates high correlation between the corresponding variables.

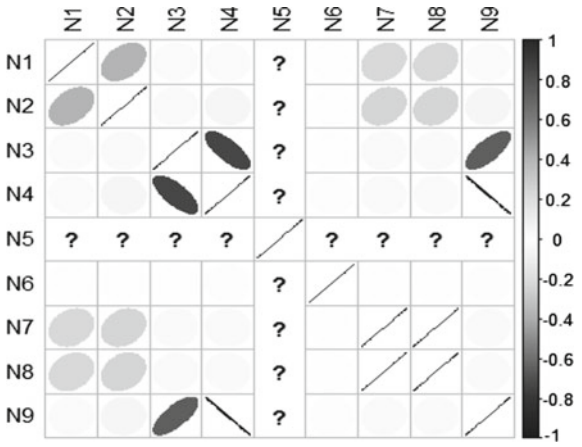


Fig. 1 Correlation matrix of the input attributes N1, N2, N3, N4, N5, N6, N7, N8, and N9

Some observations from the correlation matrix given are as follows: N5 has a constant value, and hence, a "?" is shown and it is not correlated to any other input. N4 and N9 are negatively correlated with each other.

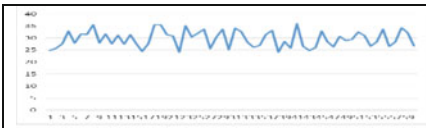
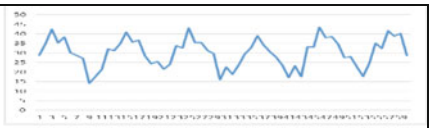
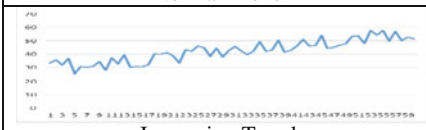
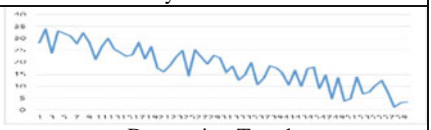
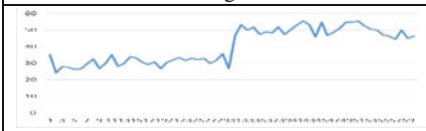
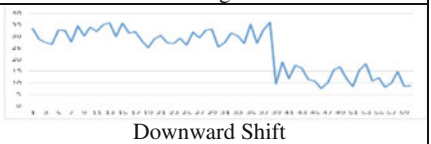
2.1 Time Series Dataset

The experimented results are also tested on time series dataset. The dataset consists of readings of various sensors (temperature, pressure, humidity, etc.) for a period of 60 s and 600 instances. Each row depicts a certain trend, e.g., normal, cyclic, increasing, decreasing, upward shift, downward shift. The first 100 values are of normal trend, next 100 values are of cyclic trend, and so on.

The first step in analysis of such dataset is the plotting of various trends. Plotting individual row gives a clear understanding about the trends. Next step is to clear the data and separate it into training and testing. For testing purposes, we need 20% of the values from each trend, and 20 readings from each trend are used for testing. Dataset is labeled as 00-normal trend, 01-cyclic trend, 02-increasing trend, 03-decreasing trend, 04-upward shift, and 05-downward shift.

As we look at Table 1, we observe that normal and cyclic trends have similar pattern, and this can cause overfitting. Moreover, this is a very small dataset (600 rows). Small datasets are very prone to overfitting. Slope, range, and standard deviation of each row is calculated and added to the dataset as 61st, 62nd, and 63rd column, respectively. Adding these extra statistical parameters is important for better functioning of the machine learning algorithm over such small dataset.

Table 1 Visualization of trends of time series dataset

 Normal Trend	 Cyclic Trend
 Increasing Trend	 Decreasing Trend
 Upward Shift	 Downward Shift

3 Machine Learning

3.1 Machine Learning Framework

The dataset given belongs to multi-label classification problem. After defining the problem, the dataset is split into training set and validation set (testing set). The splitting of data into training and testing sets is done according to the labels. Any operations that are applied on the training set must be saved and then applied to the testing set.

The variable descriptions of the dataset provided are as given as follows: (1) The input attributes N1, N2, N3, …, N9 are numerical variables. (2) The output attributes M1, M2 and M3 are categorical variables (HOLD, RAISE, LOWER, STOP).

3.2 Selecting Appropriate Algorithm

We applied some supervised learning algorithms such as decision tree, Naïve Bayes classification, SVM, CART, ensemble methods.

After practically implementing these algorithms and comparing the results, we come to the conclusion that ensemble method will work the best for the given dataset.

3.3 Comparative Study

Various ensembling algorithms are tested using the same dataset and a comparative study of the results obtained from different algorithms is done. The different ensemble algorithms which are used for testing and their various feature description along with their classification accuracy is in the table. The accuracy among the ensemble methods is highest for the gradient boosted decision tree.

3.3.1 Gradient Boosted Decision Tree

It is a powerful machine learning algorithm which is capable of doing classification, regression, and ranking. In gradient boosting, shortcomings of a model are identified by "gradients." In this algorithm, an additive model (ensemble) is fitted in a forward stage-wise manner. In each stage, a weak learner is introduced to compensate the shortcomings of the existing weak learners. It minimizes the total loss.

Different models are created F1, F2 … and each model is assigned a score for each class. The scores are then used to calculate the probabilities. The predicted label is the class that has the highest probability. After that, the loss function for each data point is calculated. The steps to calculate loss function for each data point are as follows:

1. Each label is turned into a true probability distribution.
2. The predicted probability distribution function is calculated based on the current model F1, F2 ….
3. The difference between the true probability distribution and the predicted probability distribution is calculated using KL divergence, which is nothing but a way of comparing two probability distributions. It helps to measure how much information is lost when an approximation is chosen.

4 Integration with MODBUS

The predictions coming from the machine learning algorithm are nothing but strings which are encoded to numeric values according to the preference of the operator. These values are generated by the algorithm and are stored in the holding registers present in remote device memory. As in our case, there are three outputs, and we have defined three holding registers on the server side which are updated according to the predictions coming from the algorithm. Whenever the client (SCADA operator) wants to access the predictions through a distant device, he can access these three holding registers with a suitable function code. Any error in the function code in the request PDU will result in an exception response from the server.

5 Scalability

The proposed technique is programmable and presently made with 9 input and 3 output parameters. Typically, smaller sets of datasets less than 20 inputs and less than 10 outputs are possible with one instance of the code. It is possible to have many instances of the code. This way it is scalable to any length of parameters. The precondition and assumption are that any large-scale process can be broken down into smaller stages of operations which may involve smaller datasets and control outputs; e.g., aviation I/Os for an autopilot code can be many in total, but it can be broken down into stages like runway, travel, take off. Once the process is divided into independent stages of smaller parameter sets, the proposed technique can run in parallel instances, on each stage independently.

6 Results

The proposed algorithm is developed and tested on Scientific Linux 7.0 platform. Even though it is developed on Scientific Linux, the code is compatible with other operating systems of Linux like Ubuntu 16.04 and also on different versions of Windows.

6.1 Classification Predictions

Table 2 shows the accuracy obtained for each output attribute after using various types of ensemble algorithms [2, 3] like bagging meta-estimator, Random Forest, extremely randomized tree, AdaBoost classifier, gradient boosting (GB) classifier, and gradient boosting (GB) regressor. On comparing the results of the table, the gradient boosted classifier was found out to be best suited for the given dataset, giving maximum accuracy for all the three output parameters.

Figure 2 shows the first 20 predictions of the output parameter M1 for testing dataset. Similarly, the predictions are made for the remaining two output parameters. The four controlled outputs are denoted as follows: 00-HOLD, 11-RAISE, 22-STOP, 33-LOWER.

The accuracies of the predictions for M1, M2, and M3 are 99.99% (almost 100), 92%, and 99%, respectively. The time series dataset was also executed, and the output was found to get maximum accuracy value as 97%, for max_depth to be 10.

Table 2 Experimental results

	Output attributes		
Classifiers	Accuracy for M1 (%)	Accuracy for M2 (%)	Accuracy for M3 (%)
Bagging meta-estimator	94	75.5	96
Random Forest	99.99	87.5	99
Extremely randomized tree	98.5	87.5	98
AdaBoost classifier	99.99	53	99
GB classifier	99.99	92	99
GB regressor	99.99	89	99

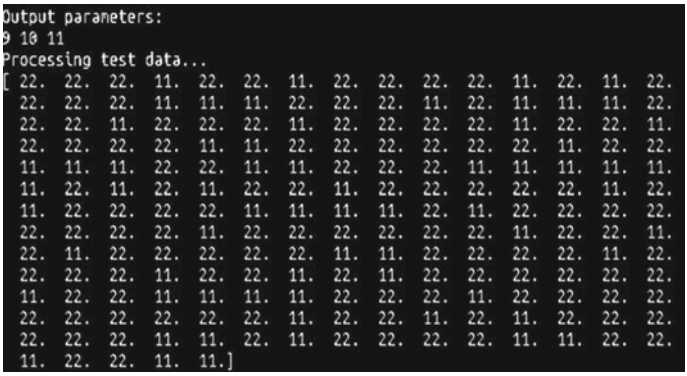


Fig. 2 Predictions using machine learning algorithm

6.2 SCADA Integration Results

The predictions obtained from the machine learning algorithm had to be sent from one device to another (the latter being a SCADA operator). Thus, a communication protocol had to be selected, which was compatible with any standard SCADA (industrial or commercial). The MODBUS communication protocol [4] was found best for communicating with SCADA systems. The MODBUS TCP/IP protocol works on client/server model, where the client requests for data and the server sends the required data. In our case, the machine learning algorithm had to be integrated inside the server in order to send the predictions when required. The client in our case is the SCADA system (industrial or commercial).

The final proposed technique which contains the machine learning predictions integrated inside the MODBUS client was tested with the commercial SCADA system as well as open-source SCADA system (EPICS [5]-based BARC SCADA system) during the summer training program which was in accordance with the Smart India Hackathon 2017, and this proposed method communicated successfully with both the systems (Fig. 3).

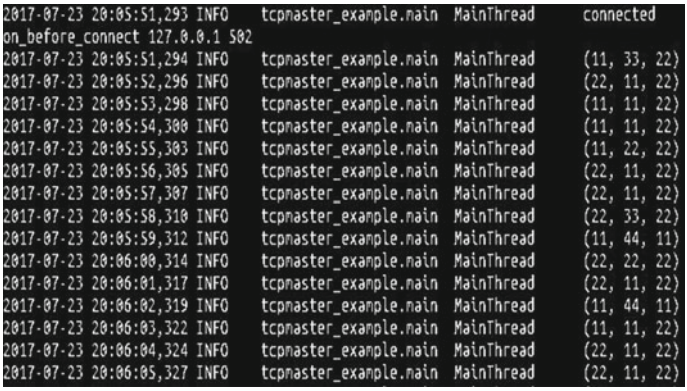


Fig. 3 Communication with standard MODBUS master code

Table 3 Abbreviations

SCADA	Supervisory Control And Data Acquisition
PLC	Programmable Logic Control
HMI	Human Machine Interface
EPICS	Experimental Physics And Industrial Control System
KL	Kullback Leibler

7 Abbreviations

See Table 3.

8 Conclusion

We conclude that the proposed method is compatible with any type of SCADA system available. The inclusion of machine learning in control room operations will not only ease the operator's work pressure but also allow him to see the trends in the outputs, with the help of which he can actually predict what will happen or what is going to happen ahead. The AI part is to assist operator as well as train new operators. It learns from human operator, learns from parameter alarm logs, and can be used to either display suggested actions, off-normal trends in SCADA or if enough confidence is generated, to actuate control devices, after operator authorization. It can generate possibility to automate difficult to model control process by learning from operator. Thus, machine learning can revolutionize the control rooms for better safety and control measures.

References

1. Machine learning based control strategy development for plant status monitoring and detection of OFF-Normal conditions, Nov 2016, https://innovate.mygov.in/wpcontent/uploads/2016/11/EXAMPLE_DATASET_HACKATHON2017.xlsx. Accessed 5 Mar 2017
2. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Duborgh, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perot, E. Duchesnay, Scikit-learn: machine learning in Python. J. Mach. Learn. Res. (2011)
3. L. Buitinck, G. Louppe, M. Blondel, F. Pedregosa, A.C. Muller, O. Grisel, V. Niculae, P. Prettenhoffer, A. Gramfort, J. Grobler, R. Layton, J. Vanderplas, A. Joly, B. Holt, G. Varoquaux, *API Design for Machine Learning Software: Experiences from the Scikit-Learn Project* (Cornell University, 2013)
4. L. Jean, GItub-ljean/modbus-tk: create Modbus app easily with Python, 6 Aug 2010, https://github.com/ljean/modbus-tk. Accessed 28 July 2017
5. EPICS—Experimental Physics and Industrial Control System. Argonne National Laboratory, http://www.aps.anl.gov/EPIC. Accessed 10 Aug 2017

Clustering and Parallel Processing on GPU to Accelerate Circuit Transient Analysis

Shital V. Jagtap and Y. S. Rao

Abstract Today, in the age of digital era, electronic circuit is the key component and its design, testing is validated through simulator. But even though use of simulator is cost-effective, large circuit simulation is quite time consuming. Also various iterations in transient analysis might make simulation slow. In this paper, we have addressed parallel computing approach using Graphics Processing Unit (GPU). Forming clusters of executable procedures are very crucial, so that it can be mapped to graphics processor for parallel processing. In every circuit nodal analysis finds current, voltage etc. at various nodes periodically. Matrix operations, linear–nonlinear equations, integration, differential equations, numerical methods are some of the very basic operations required in circuit analysis. Data-code partitioning, parallel data mapping, reductions, fast memory access, parallelizing loops are the strategies adopted for parallel processing on GPU. More than 40% speed gain is achieved on circuit having at least four components along with transient analysis for more than thousand iterations.

Keywords GPU (Graphics processing unit) · Transient analysis · Clustering LU decomposition

1 Introduction

Circuit simulators are used in almost all the electronic industries and plays very crucial role in electronic circuit design, verification and testing. All electronic designs rely truly on simulation software. In academics also, students adopts safe practices and research on simulators. DC, AC, transient, pulse or noise analysis and its graph

S. V. Jagtap (✉)
RAIT, Nerul, Navi Mumbai, India
e-mail: svjagtap@gmail.com

Y. S. Rao
SPIT, Andheri, Mumbai, India
e-mail: ysrao@spit.ac.in

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_36

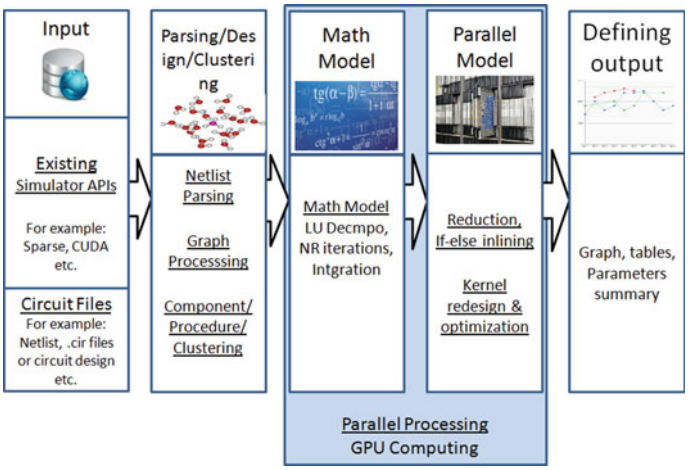


Fig. 1 System architecture

can be derived easily using simulator. But, with the density of the IC at nanoscale, the traditional simulators require more and more time to perform simulation as a whole. Transient analysis is time domain operation in which many Newton Raphson iterations are required which are very compute intensive. The simulation time of transient analysis in SPICE grows super-linearly with the number of equations that describe the circuit. For large circuit transient analysis time may extend to many hours or days. Transient analysis iteration scales as O(N1.2) where N is number of equations, whereas the double precision FLOPS of CPUs only as O(N0.96) where N is the number of transistors in the CPU [1, 2]. It means that as circuit components increases, CPU simulation time also increases exponentially.

Figure 1 shows various steps and execution of simulator in serial and parallel mode. To process circuit parameters various mathematical and load operations are required which process these parameters thousands of time and need millions of clock cycles. Some operations need common processing and common parameters. We applied clustering approach on these parameters and their operations. These clusters are then mapped on GPUs to process operations in parallel on many parameters.

This paper addresses system for GPU accelerated circuit simulation especially for transient analysis. Paper is organized as follows. Section 1 is introduction to topic and Sect. 2 gives details of previous approaches for accelerating circuit simulation. Section 3 elaborates the adopted GPU programming strategies and Sect. 4 explains clustering approach. Section 5 gives performance analysis and results.

2 Previous Work

To accelerate the simulation many approaches are proposed like parallel processing, distributed computing or specialized optimized algorithm. Transistor model is accelerated using GPU in SPICE simulator by Gulati et al. [3]. This acceleration was model-specific that is applicable to transistor model only. If-else in-lining and coalesced memory access is also proposed which helps in accelerating the operations on GPU and is useful to every GPU application. The work started by Gulati was the good start in research of parallel processing in circuit simulation. Chen and Wang used pivoting reduction technique for LU sparse solver [4], which is faster compare to the solvers like NICSLU or PARDISO. NICSLU uses a column-level dependence graph to schedule the tasks [5]. The dependence is extracted from the symbolic structure of the factors. Since the structure of the factors cannot be determined before factorization, the elimination tree (ET) is used to represent the dependence. It follows all the basic steps like numerical update, pruning with proper pivoting. These steps consume extra time for simple dense small size matrix processing. FPGA based techniques are also available for LU decomposition which are better than serial processing [6–8]. Compare to FPGA techniques GPU parallel programming is simple and easy to understand. In some approaches rather than accelerating computations, circuit or gate design is partitioned to find independent components and mapped it in parallel [9, 10]. Every circuit analysis need lots of computation, so some parallel and distributed approaches are available for mathematical operations like LU decomposition, integration [11–14]. Chen and Wang modified it for circuit simulation but for large circuit [15, 16], matrix column dependency causes slower simulation. Davis and Natarajan proposed KLU data structures and algorithm which can be used in circuit data storage and processing for sparse matrices [2]. It gives stability in matrix processing. For small circuit simulation it proves to be slow but adopted by many simulators like NGSPICE. Saol, Vuducl and Xiaoye proposed distributed approach to solve sparse matrix. This is costly approach but can be useful to extend GPU technology also. From all the available approaches, parallel processing using GPU is very cost effective approach. New faster GPUs are coming in the market. So research is still persistent to accelerate simulation on new faster GPUs. This paper focuses on utilizing computational power of GPU for heavy computations of circuit transient iterations.

3 GPU Strategies

GPU processes graphics and includes thousands of SIMD multi-threaded, multi-core processors with inbuilt memory levels having different sizes and access time. For example Kepler K40 graphics card contain 2880 cores. GPU processor do not consume enormous power, heat indulgence is adequate, so can be used with laptops or small systems. Cost of GPU is just some thousand rupees. Access to high end GPU

is available free of cost online through GPU clusters (from GPU Excellence centre). GPU is an emerging parallel processing approach for heavy computations. CUDA is the software platform available for GPU. It supports heterogeneous programming. Due to SIMD nature, sequential code is not directly executable on GPU. Redesign and optimization is needed in memory access-storage, execution configuration, instruction cycles and control flow. Following strategies are adapted to modify and redesign sequential code so that it will execute faster on GPU especially for cluster based circuit simulation.

1. Kernel formation and optimization—Avoid costly operations and replace them using less costly operations in kernel. Proper partitioning of data or operations is required which uses adequate kernel size. Two parameters are considered to decide proper kernel size: a. Maximum GFLOPS obtained from kernel and b. Maximum memory bandwidth used by that kernel.
2. Parallel reduction—Reduction is a generic operation that takes $n > 1$ values and returns a single value. Elements can be re-arranged and combined in any order. Threads need to access results produced by other threads using either shared memory or by synchronisation. Key requirements for a reduction operator $\circ$ are:

   a. Commutative $a \circ b = b \circ a$; b. Associative $a \circ (b \circ c) = (a \circ b) \circ c$

3. Minimize loops by solving data dependency.
4. If-else in-lining-Use minimum if-else statement so that optimum time is used for execution on all threads.
5. Avoid warp and thread divergence in CUDA kernel. If possible interchange the work done by warps.
6. Utilize memory hierarchy: a. Coalesced memory access utilizes optimum time to read or write data items. b. If possible copy data in shared memory or registers for fast access. c. Memory Bandwidth calculation-Choose the kernel giving highest speed up. d. Reduce/Eliminate data transfers between CPU and GPU. Combine multiple device memory allocations and transfers in one step. cudaMalloc() and cudaFree() are costly operations so minimize them by reusing the allocations. e. Use page-locked host memory for data transfers. f. Use asynchronous data copy if possible.

4 Clustering for Circuit Simulation

After compilation of netlist many procedures are required to simulate the circuit. We developed approach for circuit component and procedure partitioning and optimized for a highly-parallel GPU. Moreover, flow is structured to extract the best simulation performance from the given circuit execution. Logic is used to verify designs at the behavioral level, as well as the structural level, ensuring that a synthesized circuit's procedure cluster matches the functionality and timing of the behavioral model.

4.1 Circuit Analysis

Transient analysis is one that finds voltage (or current) versus time. Linearization of non-linear devices, operating point analysis, conductance stamping into the modified nodal analysis (MNA) matrix and linear system solution are time consuming itself. Matrix methods like LU decomposition can be used to solve linear equations. Dense matrix library proves to be time and space consumable if matrix is sparse. Various numerical methods like Newton-Raphson, Runge-Kutta or trapezoidal methods are useful to find roots of equations, derivative or integration. These methods can be parallelized to accelerate the computations.

As the execution of software functions are concerned, execution time varies. All the 'load' functions in simulator are compute intensive. It loads default parameters along with calculated parameters into the simulation matrix and its execution time sums up to approximately 54% of total analysis time. Other time consuming component is actual matrix solving to find unknown circuit node parameters which is approximately 36% [1]. In order to complete the analysis of time spent in the transient analysis—5% of time in circuit error, the truncation error calculation, 2% in NI iterations and 2% in NI integration. Then 1% is spent in rest of simulator. MNA matrix is solved using LU decomposition. Left-looking LU decomposition algorithm works columnwise and convenient to make it parallel [11]. Calculation of one column depends on values generated by previous column. Assign one column at a time to GPU and execute in parallel. Launch 'n' threads, where 'n is column size. One thread is executing one element of column. Proper synchronization is required to remove any data dependent operation. Serial algorithm worst case complexity was O(n3) whereas parallel algorithm complexity is O(n2).

4.2 Clustering

54% of simulation time is needed just in load and setup operation even if numbers of components are very few. If component count increased, it increases setup time exponentially. But if we form one load cluster of all components, load time increases by few clock cycles, at least not exponentially. So for large circuit or for transient analysis, cluster formation is better compare to manual parallel execution. Figure 2 shows approximate clusters formation. There are two modes to process complete netlist using clustering approach.

1. Component clusters—This method create clusters of components of same level. Same level means fanin of components is ready in previous level. It helps in execution of one cluster is possible at a time. Graph DFS algorithm is used select component clusters.
2. Method/Procedure clusters—All component have some common execution steps like setup or load etc. Creating clusters of these methods is method clustering.

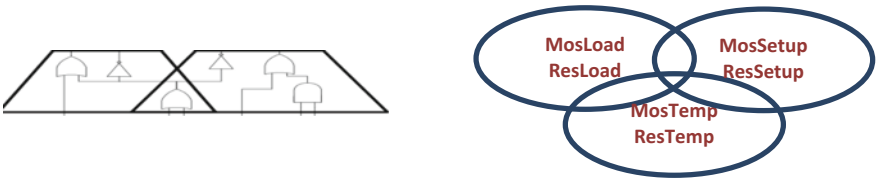


Fig. 2 Component and method clusters

Method clustering is more suitable for GPU computing as many instance of thread creation is possible due to SIMD nature of GPU.

Common operations are either with same component or different component. Same component means circuit may contain many instances of resistors or capacitors. If components are same, sub-operations are exactly same, so cluster operations are exactly same. It helps in increasing speed more compare to different components. For different components like transistor and diode, some of the sub-operations are same. So can be mapped to many threads, but thread processing time may vary. Following are some cluster creation precautions:

4.2.1 Cluster Constraints and Data Set

Data set for circuit simulation with respect to GPU processing are components with all parameters like Resistance, conductance, drain current, base current etc. In determining how to partition the netlist into clusters, we considered several factors: (i) Similarity in process with parameter data type (ii) Time required to execute functions. GPU specific factors are: (i) Minimum dependencies among parameters in various iterations. (ii) Time required to execute process should be more than a. time to load the data in GPU memory. b. overhead of selecting process from process set.

4.2.2 Partitioning

To exploit the parallelism available in the GPU, segment the simulation procedure into several logic blocks. a. Divide every logic block into sub-blocks, containing one operation. b. Compare sub-blocks of one component with other component sub-blocks. Uniform sub-blocks are given same weight. c. Apply k-means algorithm. Define number of clusters and initial mean as basic operation for every cluster. In second step we get the distance vectors. Compare distance vectors and select subblock of minimum distance as an operation in that cluster. Constraints defined above can act as threshold for distance vector and used to determine—in which cluster procedure should be added. d. Distance may vary and there may be overlap also in the operations. In the overlap, we can set flag to indicate operation is executed and don't execute it again.

4.2.3 Procedure Balancing

Cluster parameters are copied in global and shared memory. In some structures only pointers are defined and has to be converted into actual structure before copying. We map one cluster to graphics processor at a time. As fine-level granulaity, we can even divide one load method into many instances. E.g. if there are 4 resistor components, resload is repeated at least 4 times. On GPU, we can create four threads to execute load resistor function in parallel. But we need thread synchronization to manage cyclic dependency among the variables.

4.2.4 Simulation Phase

Sequence for cluster execution is based on following parameters: 1. Sequence and privilege of operation. 2. Data or input/output dependency. 3. Synchronization dependency 4. Data availability in global/shared memory. There are restrictions on GPU memory size and extra time is needed in loading, unloading all the parameters. So if cluster execution time is much more compare to serial time plus loading time, that cluster execution is suitable to make it parallel.

Sufficient memory size is also required to process all the components at a time, as for large circuit millions of components with thousands of parameters are used.

5 Performance Comparisons

Circuit netlist having basic components are tested on GeForce M980 processor with 296 cores and 2 GB graphics card memory. NGSPICE simulator is used on Ubuntu 14.04 version. NGSPICE with KLU version is considered for comparison. Execution time of thousands of transient iterations are considered. One complete iteration involves initialisation, setup, load, LU decomposition, forward- backward substitution. Setup and load involves many small mathematical operations like solving algebraic equations, integration etc. Netlist parsing time is constant for all the circuits. Clustering and parallel processing is used in parameter setup, load and mathematical operations. Speed gain is calculated using the Eq. (1)

$$\text{Percentage speed gain} = 100 * (\text{serial} - \text{parallel time})/\text{parallel time} \quad (1)$$

Average execution time is considered for every circuit execution. Table 1 gives serial and parallel execution time of five example circuits having thousands of transient iterations. Figure 3 shows relative time comparison graph among serial and parallel execution.

It shows that basic GPU strategies and clustering approach accelerate circuit processing at least by 40% and increases subsequently for more iterations.

Table 1 Serial and parallel execution of different netlist

Netlist	Circuit	No. of iterations	Serial execution in (s)	Parallel execution in (s)	Speed gain (%)
1	AC sine wave voltage	10,000	6.38	3.75	70.13
2	RC circuit	1008	0.853	0.58	47.06
3	Full wave bridge rectifier	10,000	12.54	8.23	52.36
4	Common source jfet amplifier	10,000	8.71	4.26	104.46
5	Integrator with square wave input	2520	2.115	1.4875	42.21

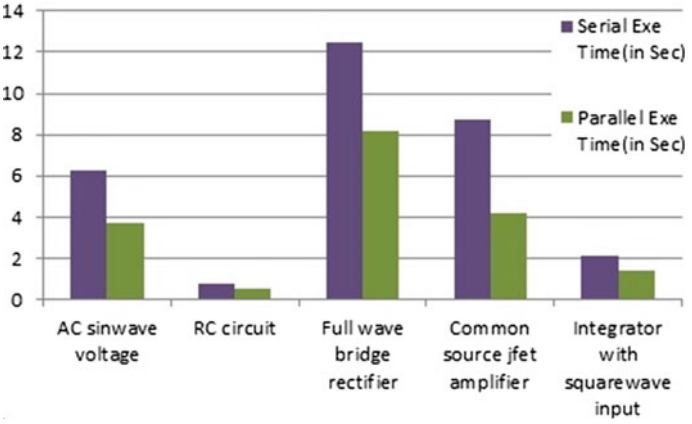


Fig. 3 Execution time comparison graph

References

1. F. Lannnutti, P. Nanzi, M. Olivieri, KLU sparse direct linear solver implementation into NGSPICE, in *19th International Conference on Mixed Design of Integrated Circuits and Systems*, Poland, 24–26 May 2012
2. T. Davis, E. Palamadai Natarajan, Algorithm 907: KLU, a direct sparse solver for circuit simulation problems. ACM Trans. Math. Softw. **37**(3), Article 36, Sept 2010
3. K. Gulati, J.F. Croix, S.P. Khatri, R. Shastr, Fast circuit simulation on graphics processing units (IEEE, 2009), pp. 403–408
4. X. Chen, Y. Wang, H. Yang, A fast parallel sparse solver for SPICE-based circuit simulators, 978-3-9815370-48/DATE15/c2015, EDA
5. X. Chen, Y. Wang, H. Yang, NICSLU: an adaptive sparse matrix solver for parallel circuit simulation. IEEE Trans. Comput. Aided Des. Integr. Circuits Syst. **32**(2), Feb 2013

6. G. Wu, Y. Dou, G.D. Peterson, Blocking LU decomposition for FPGAs, in *2010 18th IEEE Annual International Symposium on Field-Programmable Custom Computing Machines*
7. Y. Shao, L. Jiang, Q. Zhao, Y. Wang, High performance and parallel model for LU decomposition on FPGAs, 978-0-7695-3932-4/09 $26.00 © 2009 IEEE
8. M.K. Jaiswal, N. Chandrachoodan, FPGA-based high performance and scalable block LU decomposition architecture. IEEE Trans. Comput. **61**(1), Jan 2012
9. D. Chatterjee, A. Deorio, V. Bertacco, Event driven gate-level simulation with GP-GPUs (ACM, 2009), 978-1-60558497-3/09/07
10. D. Chatterjee, A. Deorio, V. Bertacco, Gate-level simulation with GPU computing. ACM Trans. Des. Autom. Electron. Syst. **V**
11. H.M.D.M. Bandara, D.N. Ranasinghe, Effective GPU strategies for LU decomposition, in *IEEE International Conference on High Performance Computing* (2011)
12. L.F. Cupertino, A.P. Singulani, C.P. da Silva, M.A. Pacheco, LU Decomposition on GPUs: the impact of memory access, 978-0-7695-4276-8/10 $26.00 © 2010 IEEE
13. T. Dong, A. Haidar, P. Luszczek, J.A. Harris, S. Tomov, J. Dongarra, LU factorization of small matrices: accelerating batched DGETRF on the GPU (2014)
14. N. Galoppo, N.K. Govindaraju, M. Henson, D. Manocha, LU-GPU: efficient algorithms for solving dense linear systems on graphics hardware (ACM, 2005), 1-59593-061-2/05/0011
15. L. Ren, X. Chen, Y. Wang, C. Zhang, H. Yang, Sparse LU factorization for parallel circuit simulation on GPU (ACM, 2012), 978-1-4503-1199-1/12/06
16. X. Chen, Y. Wang, H. Yang, An adaptive LU factorization algorithm for parallel circuit simulation, 978-1-46730772-7/12/$31.00 ©2012 IEEE

Efficient Graph Extraction-Based Clustering Technique for Social Network Analysis

Sohini Jain and Vaibhav Jain

Abstract Social networks have gained popularity recently with the advent of sites such as Flicker, Delicious, MySpace, Friendster, Twitter, Facebook. Massive quantities of data are produced daily from various social networks. The users' base for some of these popular networks is huge, e.g., billions in Facebook and still growing. In such a social network platform, large groups of people upload/download data. For analysis purpose, users can be grouped into clusters which can provide meaningful information and can be applied in applications like community detection, business intelligence, event detection. We can efficiently analyze such social network's graph by clustering techniques. In clustering analysis, we partition a social network (graph) into clusters (subgraphs) based on connectivity and interaction between users (nodes). In our work, we have implemented a clustering technique which scales well and as well as memory efficient which is based on weighted kernel k-means algorithm and uses a graph extraction technique for identifying clusters in a social network. Evaluation of our implementation shows good clustering quality.

Keywords Social networks · Clustering · K-means algorithm · Graph extraction

CCS CONCEPTS **Social networks** → graph clustering · Community detection

1 Introduction

Social networks produce big volume of data. This large data needs to be managed and reused for some analytical aspects like to recognize communities, business intelligence, event detection in social media. Massive progress in social networks

S. Jain · V. Jain (✉)
Institute of Engineering & Technology, Devi Ahilya Vishwavidyalaya, Indore, India
e-mail: vjain@ietdavv.edu.in

S. Jain
e-mail: sohinijain02@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_37

emphasizes for analysis in social network clustering. For example, recently, Facebook introduced a new feature of friend suggestion based on clustering their network.

There are different procedures to analyze these types of datasets; one of these procedures is clustering. Clustering is an important aspect of social network which not only results in driving force for business strategies but also a prominent factor in the detection of usual events and activities.

Since social networks can be expressed as a graph, the vertices represent the users while edges represent the relationship between them. We can efficiently analysis such a social network graph by clustering technique. In graph clustering, we partition a network into subgraph in such a way that minimizing the cut between clusters (i.e., finding high connectivity ratio within cluster vertices). This extraction process is easily possible with small size dataset but as soon as the size of dataset tends to increase, the complexity starts to arises, and in that large amount of dataset which is generally expected to be scattered and unstructured. Hence by using internally dense and externally sparse clusters, we can easily analyze the social networks.

Graph extraction and weighted kernel means (GEM) algorithm can be used for efficient clustering of social networks in terms of scalability and memory efficiency. In our work, we have developed a Java tool which implements the GEM algorithm. This Java tool consists of various classes which process the social network data in the form of undirected graph and generate the clusters of graph and evaluate the quality of clusters.

2 Preliminary Concepts

2.1 Clustering

Clustering is a grouping of set of objects in such a way that similar objects belong to the same group which is called a cluster. Clustering plays an important role in how people analyze and describe the world. Clustering can be used in many application areas such as fraud detection, event detection, and image processing. Figure 1 shows the result of clustering performed on a dataset where it got partitioned into different clusters.

2.2 Graph Clustering

In graph clustering, clusters are a group of vertices, in such a way that there should be higher connectivity within edges of cluster and lesser edges between clusters. In short, graph clustering is grouping of vertices of a graph into clusters. Connectivity of a graph can be represented by adjacency matrix A. As shown in Fig. 2, where Aij = 1 if an edge exists between vertex i and vertex j, and 0 otherwise.

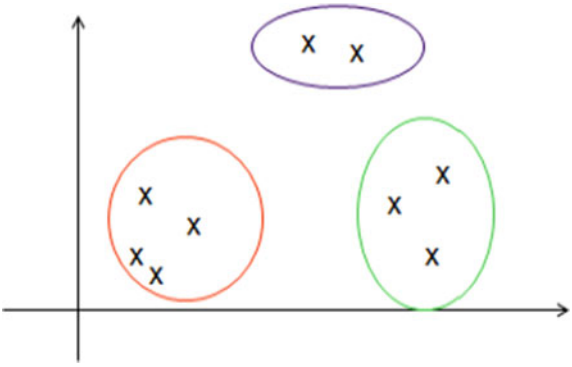


Fig. 1 Result of clustering on a dataset

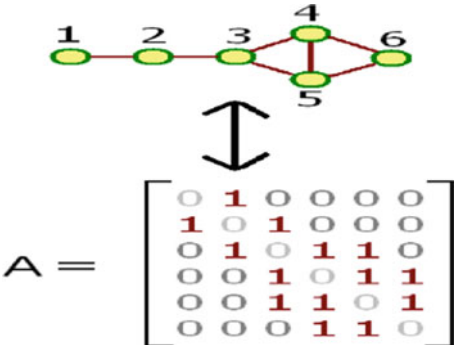


Fig. 2 Matrix representation of graph

In graph clustering k disjoint clusters V_1, …, V_k such that $V = V_1 \cup \ldots \cup V_k$. Partition a graph.

2.3 Weighted Kernel K-Means Clustering

Weighted kernel k-means clustering is a centroid-based partitioning technique. The aim of algorithm is to find k cluster $\pi 1$, $\pi 2$, …, πk that should minimize objective. The no. of clusters generated depends on the user-defined value, i.e., *k*. An objective function is used for assessing the quality of partitions so that objects within the cluster are similar to one another. In this process, we iteratively calculate the centroid for each data point and assign the data point to its nearest cluster. For each cluster, new mean value is updated using the objects assigned to the cluster. All the objects are then reassigned using means updated and new cluster center is obtained. This iteration is continuous till assignment is stable.

Using distance matrix, assign every vertex to its nearest cluster by calculating the squared distance of a vertex v with the cluster V_c as follows:

$$\text{dist}(\hat{v}, V_c) = \begin{cases} \hat{\alpha} - \frac{\sigma}{degree(V_c)} \text{if } \hat{v} \in V_c \\ \hat{\alpha} + \frac{\sigma}{degree(V_c)} \text{if } \hat{v} \in V_c \end{cases}$$

where $\hat{\alpha} = \frac{\sigma}{\hat{d}} - \frac{2links(\hat{v}, V_c)}{\hat{d} \cdot degree(V_c)} + \frac{links(V_c, V_c)}{degree(V_c)}$ and $\hat{\alpha}$ represent the degree of $\hat{v}$ and *links* $(\hat{v}, V_c)$ indicate the sum of edge between $\hat{v}$ and the vertices in V_c.

Normalized cut:
The objective of normalized cut is to minimize the cut between clusters with respect to the degree of a cluster. This objective is represented as follows:

$$\min_{v1 \ldots vk} \sum_{i=1}^{k} \frac{links\,(vi,\ v/vi)}{degree(vi)}$$

links $(\boldsymbol{v_i}, \boldsymbol{v_j})$ define as a sum of edges weights between two vertex sets v_i and v_j. That is,

$$links\left(v_{i,}, v_j\right) = \sum_{i \in vi, j \in vj} Aij$$

2.4 Down-Path Walk Algorithm

In down-path walk algorithm, we traverse a graph where vertex is considered as seed point if degree of that vertex is greater than their connected vertices in a path. Following is the pseudo-code of down-path walk algorithm:

- Follow a path $v_i \leftarrow v_j$ as a down-path if $d_i \geq d_j$.
- Traverse the number of down-paths.
- Find a seed (point) by selecting the final vertex in the path.
- Mark the seed and its neighbors.
- Repeat this procedure till we get k seeds in the graph.

3 Literature Review

Graph clustering algorithm works well for the social network which has few million vertices but the performance of algorithm rapidly decreases and consumes more time and memory when the network size increases. To deal with graphs of such scale in clustering problems, various techniques have been proposed.

Abou-Rjeili and Karypis [1] proposed coarsening strategies to resolve the graph clustering problem of power-law graphs for multilevel graph clustering framework. However, multilevel framework takes high significant time and memory.

Sui et al. [2] presented the initial implementation of a clustered low-rank approximation algorithm for large social network graphs and its usage for link prediction.

Schreiber et al. [3] proposed (α, β) clustering algorithm in social networks which find the cluster with internally dense and externally sparse in overlapped graph. They focus on only overlapped data. They use a complex and less efficient strategies.

Nussbaum et al. [4] proposed clustering social network using distance—preserving subgraph. They use subgraph strategy but not refine the cluster into original network. They use an incremental algorithm DP cluster which is order dependent.

Macropol and Singh [5] proposed an algorithm to compute the clusters in a big network. Instead of finding clusters in a graph, they only found the subset of best clusters that saves memory and also takes less time. Arthur and Vassilvitskii [6] presented the k-means++ algorithm which randomly picks a seed and then repeatedly selects seeds with respect to the distance to existing seeds. The k-means clustering algorithm and its optimization have been addressed well. Indeed, many different techniques have been proposed to initialize the cluster seed points. There is randomly selected seed may not provide a good quality of cluster and degrade the performance on getting wrong initial seed point.

Satuluri et al. [7] proposed technique to improve the existing clustering algorithms using graph sparsification, in which they tried to reduce the number of edges in the graph. They compute the clustering result by only considering the sparsified graph which might distort the structure of the original network.

Dhillon et al. [8] proposed a GEM algorithm which propagates the clustering of the representative subgraphs to the entire network and refines the clustering of the original network. They use a down-path walk algorithm to select initial seed point, which provides a better cluster quality. They use a weighted kernel k-means algorithm and process a scalable and memory-efficient approach to cluster a large social network. We have used GEM algorithm because it produces clusters of quality better than existing graph clustering algorithms, and also, it is much faster and efficient memory consumption.

4 Graph Extraction Process and Weighted Kernel K-Means

We have implemented clustering of social networks using a Java application. Our application takes an input of social network dataset and generates clusters of this dataset. The input dataset is in the form of a csv file which consists of two columns. The first column holds the vertex, and the second column holds its connected vertex. Finally, we get output as clusters of social network. The following steps of GEM algorithm are implemented by Java tool to generate clusters of social networks:

(A) **Graph Extraction**

- Calculate degree of vertex of dataset (graph).
- Extract vertices using given threshold value.
- Extract subgraph from extract vertices.
- Calculate subgraph degree.
- Calculate connected component of subgraph.

(B) **Clustering using weighted kernel k-means algorithm**

- Calculate seed point.
- Initialize the cluster.
- Evaluate final cluster.

A. **Graph Extraction**

Using graph extraction, we have extracted subgraph from original graph by selecting higher degree vertices. The input datasets vertices degree can be of higher range which leads to large amount of data to process, so we have restricted the input data with a threshold value. We have applied a parameter to store only that vertices which having degree greater than a threshold value. This filters our actual input dataset and provides a new dataset to calculate seed points. As shown in Fig. 3, the memory consumption is reduced using graph extraction technique, as increasing the threshold value the memory reducing percentage also increases.

B. **Clustering of Graph Extracted**

Once a subgraph is extracted from original graph, we apply clustering on that subgraph instead of original graph using weighted kernel k-means algorithm. On applying clustering on extracted subgraph, we first find seed points.

Seed points
The initial data points in clusters are referred to as seeds. Parameter for identifying better seeds is to place seeds in a way that they are as far as possible from each other. Note that in the weighted kernel k-means algorithm, the squared distance between two data points $\varphi(xi)$ and $\varphi(x\,j)$ is represented by:

$$||\varphi(x\,i) - \varphi(x\,j)||2 = K\,ii - 2\,K\,ij + K\,jj$$

We have calculated seed points in two ways:

1. Calculate random seed points.
2. Calculate seed point using down-path walk algorithm.

Input: $\mathcal{V}$: the vertex set, $\mathcal{V}_i$ $(i = 1, ..., k)$: initial clusters, τ_{max}: maximum number of iterations.
Output: $\mathcal{V}_i^*$ $(i = 1, ..., k)$: final clusters.

Initialize $\tau = 0$.
repeat
  for each vertex $\hat{v} \in \mathcal{V}$ **do**
    $\mathcal{V}_p \longleftarrow$ the current cluster of $\hat{v}$.
    for each cluster $\mathcal{V}_i$ **do**
      $$\hat{\alpha} = \frac{\sigma}{\hat{d}} - \frac{2links(\hat{v}, \mathcal{V}_i)}{\hat{d} \cdot degree(\mathcal{V}_i)} + \frac{links(\mathcal{V}_i, \mathcal{V}_i)}{degree(\mathcal{V}_i)^2}.$$
      if $\mathcal{V}_i = \mathcal{V}_p$ **then**
        $$\delta_i = \frac{\hat{d} \cdot degree(\mathcal{V}_i)}{degree(\mathcal{V}_i) - \hat{d}} \left\{ \hat{\alpha} - \frac{\sigma}{degree(\mathcal{V}_i)} \right\}.$$
      else
        $$\delta_i = \frac{\hat{d} \cdot degree(\mathcal{V}_i)}{degree(\mathcal{V}_i) + \hat{d}} \left\{ \hat{\alpha} + \frac{\sigma}{degree(\mathcal{V}_i)} \right\}.$$
      end if
    end for
    Find $\mathcal{V}_q$ s.t. $\delta_q \leq \delta_j$ for all j $(j = 1, ..., k)$.
    if $\mathcal{V}_p \neq \mathcal{V}_q$ **then**
      $\mathcal{V}_p = \mathcal{V}_p \setminus \{\hat{v}\}$, $\mathcal{V}_q = \mathcal{V}_q \cup \{\hat{v}\}$.
      Update $links(\mathcal{V}_p, \mathcal{V}_p), degree(\mathcal{V}_p)$, $links(\mathcal{V}_q, \mathcal{V}_q), degree(\mathcal{V}_q)$.
    end if
  end for
  $\tau = \tau + 1$.
until not converged and $\tau < \tau_{max}$
$\mathcal{V}_i^* = \mathcal{V}_i$ $(i = 1, ..., k)$.

Fig. 3 Pseudo-code for weighted k-means algorithm

In down-path walk algorithm, the parameter for selecting any seed point s is that it has the degree greater than its entire connected vertex. For example if vertex *v* is connected to vertex *p*, *q*, *r*, *s*, *t,* then we search for the vertex which has degree greater than *v* and search till the last vertex if highest degree vertex found we add it into our seed point list and now that seed will be our next vertex to find another seed this process repeated for n times to get n seeds. We have used recursive function for calculating these seed points. For random seed points, we have selected random vertex and add it into seed point list rather applying any comparison with its other connected vertices.

Weighted Kernel K-Means

Weighted kernel k-means algorithm is used to find the clusters. Following pseudo-code describes the calculation of centroid for each vertex and assigns that vertex to its nearest cluster.

Let us consider a data point $\emptyset(\hat{x})$ which currently belongs to cluster π_p. Suppose that $\emptyset(\hat{x})$ is moved to π_q, then the centroid m_q of cluster π_q changes to m_q.as follows

$$m_q = m_q + \frac{w_i \emptyset(x) - w \cdot m}{\sum_{x_i \in \pi_q} w_f + w}$$

where w is the weight of $\emptyset(\hat{x})$. The movement of $\emptyset(\hat{x})$ from π_p to π_q advantageous if the decrease of J_p greater, then increase of J_q. That is,

$$\frac{w \sum_{x_i \in \pi_p} w_i}{\sum_{x_i \in \pi_p} w_i - w} \left\| \emptyset(x') - m_p \right\|^2 > \frac{w \sum_{x_i \in \pi_q} w_i}{\sum_{x_i \in \pi_q} w_i - w} \left\| \emptyset(x') - m_q \right\|^2$$

where J_p and J_q are the effective spread of cluster π_p and π_p, respectively. After every data point is considered, the centroid of clusters is updated only when the movement of $\emptyset(\hat{x})$ from current cluster will be beneficial or not. The movement of $vertex\, \emptyset(\hat{x})$ is beneficial when greatest decrease of the objective for cluster $\frac{w \sum_{x_i \in \pi_q} w_i}{\sum_{x_i \in \pi_q} w_i - w} \left\| \emptyset(x') - m_q \right\|^2$ is minimum. This procedure is repeated until the centroid of clusters is stable or a maximum number of iterations are reached.

5 Experimental Results

We have implemented GEM algorithm using Java language in Netbeans IDE. We have used hardware component Intel Pentium 4 based PC having 4 GB of RAM and software components JDK: 1.7. Netbeans IDE: 6.5, Windows operating system in our development process.

5.1 Datasets

We have used two different real-world online social networks: Delicious and Flicker dataset.

Delicious is a provider of social bookmarking Web service for storing, sharing, and discovering Web bookmarks. This dataset is in the form of csv file in which data is stored with comma separated values. **Flicker** is an image and video hosting site, Web services suite, and online community provider. This contains the friendship network crawled and group memberships. Both datasets are in the form of csv files which consist of two columns vertex and its connected vertex or edge (Table 1).

Table 1 Summary of social network dataset

Graph	No. of vertices	No. of edges
Delicious	103,144	1,419,519
Flicker	80,513	5,899,882

5.2 Results

This section describes the result analysis approach of our implemented approach. The dataset is collection of vertices and its associated edges. It is considered as a graph having interconnected vertices through edges, which are undirected. We have partitioned this graph into different number of clusters. The actual dataset is very large in size and also complex to process, so we have further extracted this dataset by using threshold value and gain the actual output. The actual data is very large to process which required high memory consumptions and computation time, so we have restricted the input data with a threshold value. Table 2 shows the extracted dataset. The actual input nodes are 103,144 in case of Delicious which reduces to 3193 (32.50%) nodes for threshold value 100 and 2315 (44.50%) nodes for 150 threshold value. For Flicker dataset, the actual input nodes are 80513 which reduces to 2673 (30.12) for 100 threshold value and 1938 (41.54) for 150 threshold value.

Figure 4 shows memory reduction after graph extraction for Delicious dataset. As we increase threshold value, memory reduced percentage increases. For 15 thresholds, 62.70% memory reduced.

Table 2 Summary of extracted datasets generated

Dataset	Delicious	Flicker
No of nodes input	103,144	80,513
Extracted nodes (Threshold 100)	3193 Reduce 32.30%	2673 Reduce 30.12%
Extracted nodes (Threshold 150)	2315 Reduce 44.50%	1938 Reduce 41.54%

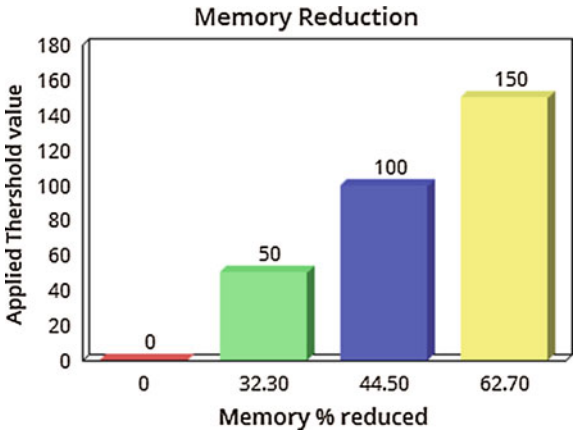


Fig. 4 Memory reduction for delicious dataset

Table 3 % of cluster within edges for datasets

	No. of clusters					
Strategy used for choosing seed points	100 Del Fli		150 Del Fli		200 Del Fli	
Using down-path walk	98.6	99.3	99.2	98	96	94.2
Random seed	98.3	98.9	98	97	94	90.7

5.3 Comparative Analysis of Down-Path Walk and Random Seed

We apply comparative analysis on Delicious and Flicker dataset. Table 3 shows the comparative representation of cluster quality for two different approaches. In first approach, we have calculated the percentage of clusters within edges by using seed points which are generated by down-path walk algorithm. For different numbers of clusters, the percentage values are different.

In second approach, we have calculated the percentage of clusters within edges by using seed points which are randomly generated. In this approach, we have taken seed points randomly rather than using down-path walk. For different numbers of clusters, the percentage values are different.

From Table 3, we can conclude that the quality of percentage of cluster within edges is more in case of with down-path walk approach (DW) than random selection of seed points of cluster. Using GEM algorithm, we get higher percentage of cluster within edges and lower edges outside the cluster. As shown in Figs. 5 and 6, the time taken is different for different no. of clusters. As increasing the number of cluster, the computation time is also increased. So, the finding no. of cluster with higher accuracy is important aspect of our result. The comparative analysis of time taken by both processes, which concludes that in case of with down-path walk time consumption, is less (Figs. 7 and 8).

6 Conclusions

We have taken different number of clusters for evaluation. For different numbers of clusters, the quality of clustering is different. As increasing the number of clusters, the computation time also increases. We have taken 100, 150, and 200 numbers of clusters, respectively. The comparative analysis of execution time taken for in case of with down-path walk is less as compared with random walk process. We have got 98% of the within-cluster edges, and a lower normalized cut value that shows the generated clusters is of better quality. Our implemented solution works efficiently up to 10 billion of nodes and up to 200 clusters. We have also used some assumptions in our implementation. The performance of our implemented

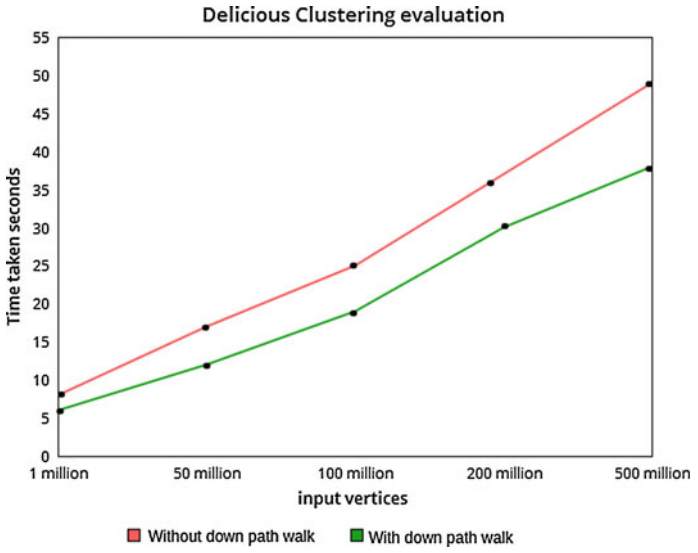


Fig. 5 Delicious clustering evaluation graph

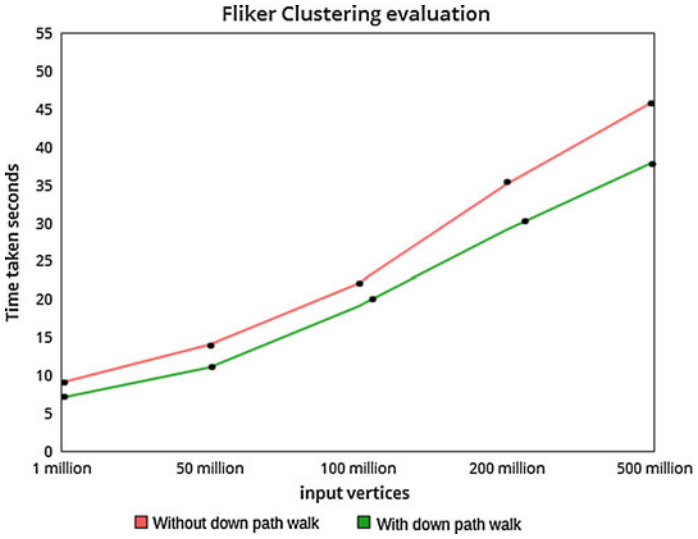


Fig. 6 Clustering evaluation on Flicker dataset

Java application works efficiently up to 10 billion nodes; however, the execution time and memory consumption increase after increasing the nodes beyond this limit.

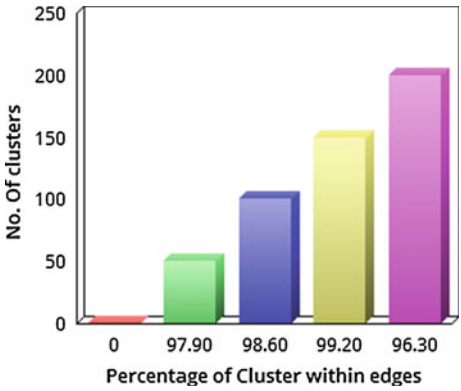


Fig. 7 Clustering using DW on Delicious

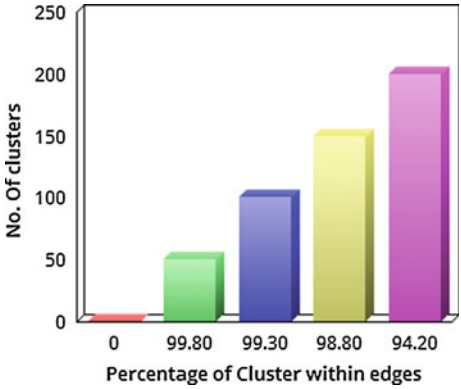


Fig. 8 Clustering using DW on Flicker

Following assumptions which we have taken are:

- The social network dataset we have used is an undirected graph.
- Implementation of these algorithms on Java platform was efficient solution up to billions of nodes. Maximum heap size we have used is 1024 MB for Java code.
- For calculating positive distance between the two vertices, we have used a positive value alpha.
- The complete implementation observes that Java applications are good for data processing.

Although computation time depends on the hardware used in the systems such as processor and RAM, it may be suggested for large data processing with good level of hardware support.

Although, better result has been observed in proposed solution in comparison with existing approach, but have some limitations. Proposed solution only considers undirected graph. As future work, our approach can be extended for directed graph

also. Though we have implemented the algorithm using Java application, parallel version of the algorithm can be implemented on Hadoop platform and which may yield more accurate results and speedup can be obtained for multiple nodes. Various dataset verifications and testing can be done further.

References

1. A. Abou-Rjeili, G. Karypis, Multilevel algorithms for partitioning power-law graphs, in *Proceedings 20th IEEE International Parallel & Distributed Processing Symposium* (2006)
2. X. Sui, T.H. Lee, J.J. Whang, Parallel clustered low-rank approximation of graphs and its, in *25th International Workshop on Languages and Compilers for Parallel Computing* (Berlin, Heidelberg, 2013)
3. N. Mishra, R. Schreiber, I. Stanton, Clustering social networks, in *5th International Workshop on Algorithms and Models for the Web-Graph* (Springer, Berlin, Heidelberg, 2007)
4. R. Nussbaum, A.H. Esfahanian, P.N. Tan, Clustering social networks using distance-preserving subgraphs, in *The Influence of Technology on Social Network Analysis and Mining* (Springer, Vienna, 2013)
5. K. Macropol, A. Singh, Scalable discovery of best clusters on large graphs, in *Proceedings VLDB Endow*, vol. 3, 1–2 Sept 2010
6. D. Arthur, S. Vassilvitskii, k-means++: the advantages of careful seeding, in *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms,* Society for Industrial and Applied Mathematics (PA, USA, 2007)
7. V. Satuluri, S. Parthasarathy, Y. Ruan, Local graph sparsification for scalable clustering, in *Proceedings of the 2011 ACM SIGMOD International Conference on Management of data SIGMOD 2011* (ACM, New York, NY, USA, 2011)
8. J.J. Whang, X. Sui, I.S. Dhillon, Scalable and memory-efficient clustering of large scale social networks, in *2012 IEEE 12th International Conference on Data Mining* (Brussels, 2012)
9. B. Mirkin, *Clustering for Data Mining: A Data Recovery Approach* (Chapman & Hall/CRC, 2005)
10. Social Computing Data Repository, http://socialcomputing.asu.edu/datasets/Delicious

Smart Home Energy Management System—A Multicore Approach

R. Ranjith, N. Krishna Prakash, D. Prasanna Vadana and Anju S. Pillai

Abstract Smart homes require energy management system (EMS) for utilizing renewable and stored energy sources. Energy efficiency can be improved by automating the connection between energy sources and loads. This is achieved by the combination of information technology and IoT to move toward the cyber-physical energy systems. Technological advancements in smart homes demand high computation capability to handle large amount of data exchange. Multicore architecture-based EMS is a cost-effective solution to make the system more robust and reliable. This paper proposes a home energy management system (HEMS) that could switch between energy sources depending on the load to be operated and communicate the encrypted energy consumption information to the data collection unit. A performance comparison between a single-core implementation and multicore implementation is carried out for HEMS in terms of various metrics, viz, execution time, processor speedup, and efficiency. The system outperforms the single-core implementation with a speedup of 1.32 using multicore architecture enabling simultaneous switching operations and communication of encrypted information.

Keywords Smart homes · Home energy management systems
Multicore architecture

R. Ranjith (✉) · N. Krishna Prakash · D. Prasanna Vadana · A. S. Pillai
Department of Electrical and Electronics Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore, India
e-mail: r_ranjith@cb.amrita.edu

N. Krishna Prakash
e-mail: n_krishnaprakash@cb.amrita.edu

D. Prasanna Vadana
e-mail: d_prasanna@cb.amrita.edu

A. S. Pillai
e-mail: s_anju@cb.amrita.edu

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_38

1 Introduction

In emerging smart grid technologies, home energy management system (HEMS) plays a vital role in improving the efficiency, cost, reliability, and energy conservation for distribution systems. There was a 64% drastic increase in the sale of smart home devices in 2016 [1]. With the advent of advanced technologies, HEMS is getting reshaped to handle renewably energized cyber-physical energy systems (CPES).

Smart homes could improve the energy efficiency by utilizing renewable energy resources like solar power and wind power along with the regular grid supply. Depending on the demand, appropriate energy source can be utilized, thus reducing the energy consumption from the grid leading to a substantial cost saving. The need for energy management system (EMS) is significant in smart homes for automating the connection between the energy sources. The EMS decides on the choice of source based on the amount of charge left in the battery which is charged by the renewable source. In the embedded system perspective, technological improvements in smart home overload the system by large amount of data and execution of several tasks at the same time. Along with the battery SoC estimation, the EMS must perform several other tasks at the same time. Implementing the system in single core increases the response time due to its sequential mode of execution. On the other hand, multicore architecture can increase the robustness and reduce the response time by means of concurrent execution. In this paper, details regarding development of HEMS and its implementation using multicore processor are presented.

The rest of the paper is organized as follows: Sect. 2 presents the state-of-the-art work in the field of EMS in homes; Sect. 3 explains the system overview; Sect. 4 elaborates the multicore implementation of the HEMS with details of software tasks. Finally, Sect. 5 describes the results and discussion about the outcome and is concluded in Sect. 6.

2 Literature Review

The EMS helps users to efficiently manage the energy consumption, by monitoring and controlling their household loads. Different strategies are employed for energy management which includes the generation of electricity using renewable sources and effective switching of load from grid to renewable-based local storage like batteries. Several techniques starting from a microcontroller-based energy monitoring to HEMS [2–4] have been proposed in the field of consumer electronics. On the other hand, with the increasing severity of energy problems, many studies on intelligent EMS with wireless sensor technologies have been conducted [5–8].

In smart grid, EMS communicates wirelessly to handle data in real time by installation of smart meters. This involves communication over a network, machine-to-machine communication, and cloud computing moving toward the development of a complete IoT-based system [9–13]. The recent researches are focused on single-core

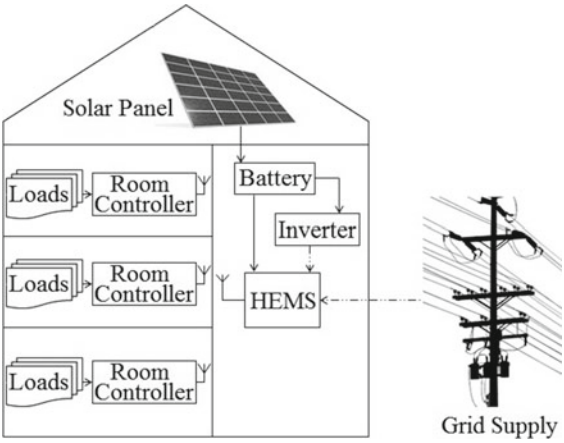
implementation of such systems. In [14], a multiprocessor architecture is used in smart home for coordinating several tasks. In order to make the system more suitable for emerging smart grid technologies and to develop a CPES, multicore approach is essential. Developments in multicore architecture-based processors have made it to emerge as a cost-effective solution that can meet the computational needs of the modern embedded systems. By splitting the computations across multiple cores saves more energy when compared to the single-core system [15, 16]. This paper proposes development of HEMS enabled with IoT to investigate the possibilities of guaranteed timely computations with reduced runtime and increased speedup using multicore processors.

3 Home Energy Management System

The HEMS monitors the status of home, which includes the availability of grid, SoC of battery, and type of load based on the power consumption, at regular intervals and manages the electrical connectivity to reduce the burden on the conventional grid (provided adequate storage is available), thereby enabling the consumer to live a "green life."

The EMS is implemented on a master controller in turn is facilitated by separate room controllers for each room as shown in Fig. 1 [17]. Grid and battery are the two power sources available, and the battery is considered to be charged from solar panel. Battery management unit (BMU) in HEMS consists of a battery-level indicator for sensing SoC of the battery and an inverter. The loads are connected to relay through room controllers which will switch the appliances to either grid or battery based on the decisions from the master controller.

Fig. 1 System overview



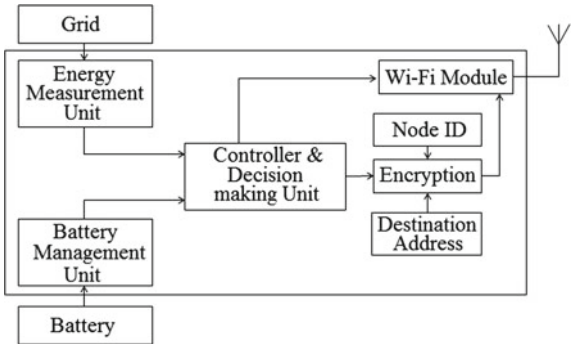


Fig. 2 Block diagram of HEMS

Loads in the house are connected to the respective room controllers. Whenever a load is turned ON, the room controller identifies the type of load, i.e., heavy load, medium load, or light load and is communicated to the master controller. Based on the status like availability of grid, solar power, battery backup, and the type of load, the master controller directs the room controller to either connect the loads to grid or to the battery. If the battery backup is sufficient, the master controller will instruct the room controller to switch the load from the grid to the local storage. Thus, the energy consumption from the grid is reduced.

HEMS also measures energy consumption from the grid and communicates the same to the nearby data collection unit. This communication is based on Wi-Fi operating in industrial, scientific, and medical radio (ISM) bands which makes the data susceptible to various forms of attacks and demands encryption of data.

The data to be communicated from the HEMS must contain the information such as the node ID, timestamp, energy consumption, and destination address. All these information are encrypted together before transmission using block cipher technique as the data size is fixed. The algorithm chosen for encryption is 128-bit Rijndael/AES [18].

4 Multicore Implementation

HEMS constantly communicates with the room controllers to keep track of the load usage. Along with the communication, the HEMS constantly monitors the battery SoC and the grid availability. The decision on the choice of source that can power the load depends on SoC of the battery and the presence of grid power. Based on the switch-over command received from HEMS, the room controllers connect the load to the grid or battery through the inverter. This process of constant communication, monitoring, and taking decisions consumes significant time when realized on a microcontroller. Additional process of encryption and communication of the energy consumption increases the burden on the microcontroller.

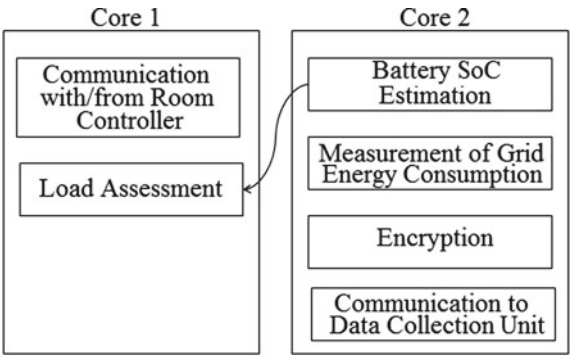


Fig. 3 Multicore task allocation

Adaption of multicore architecture for microcontroller enables parallel execution of processes as shown in Fig. 3. A dual-core microcontroller is considered for implementing HEMS in real time. The task of communicating with the room controllers to decide upon the switch-over of supply to meet the load is allocated to core 1. On the other hand, SoC estimation is performed in core 2, and the data is loaded in a common memory to both the cores. Core 1 retrieves the data from the memory before taking the decision. Suitable technique for communication using common memory is mailbox [16]. Wherein the data is stored in the memory and the sender task notifies the receiver task. Notification can be in the form of an interrupt. Along with the SoC estimation, core 2 acquires the energy data from EEPROM and encrypts before transmitting to the data collection unit as shown in Fig. 2.

4.1 Task Set for HEMS

This section deals with the tasks shown in Table 1 required for implementing HEMS for energy management and measuring unit. The room controllers determine the power requirement for the room using current transducer and voltage transducer for each room which is wirelessly communicated to the HEMS. In HEMS, the task for receiving the information from room controllers is denoted as T_{rc}. The second task (T_{soc}) retrieves the SoC of battery from the common memory. Based on the presence of grid power, SoC and load to be met at that time instant, a decision task (T_{dec}) broadcasts the command signal to the room controllers to actuate the relay in such a way that either grid or battery through the inverter supplies power to the load. The SoC of the battery is estimated using the estimation task (T_{esoc}) and stored in the common memory. The energy measurement task (T_{em}) acquires the energy consumption data from energy measurement unit. The encryption task (T_{en}) adds the required node ID and timestamp to the acquired data and encrypts them by means of block cipher. The communication task (T_c) transmits the encrypted data to the data collection unit through Wi-Fi.

Table 1 Execution time for task set

Task name	Task description	Execution time (μs)	Deadline (μs)
Trc	Room controller task	135	200
Tsoc	SoC retrieval task	86	150
Tdec	Decision-making task	234	260
Tesoc	Estimation of SoC	485	510
Tem	Energy metering task	220	300
Ten	Encryption task	640	700
Tc	Communication task	80	110

5 Results and Discussion

This paper focuses on investigating the viability of using multicore processor architectures for smart home applications. The performance of the smart HEMS with single-core and multicore processor is analyzed. The tasks in both single-core and multicore processors are tested on identical software environment based on Keil IDE and similar hardware setup based on LPC4357. The execution time of each task in single-core environment is shown in Table 1.

The task set is scheduled based on clock-driven scheduling. The execution is time triggered, initiating with reception of room controller message by HEMS. Starting from the room controller task and till the communication task, the execution happens sequentially. The timing parameters are determined, and the task trace is plotted using TORSCHE toolbox in MATLAB. The task trace for single core is shown in Fig. 4, and multicore processor is shown in Fig. 5. It is evident from the task trace that all the tasks could complete their execution before the assigned deadline as in Table 2. The time taken for completing the entire functionality once is 1.88 ms in single-core-based HEMS. The execution time can also be expressed as speedup. Speedup can be calculated as the ratio of total execution time of single-core implementation (t_s) to the total execution time of dual-core implementation (t_n). For the single-core implementation, the speedup is 1 since it acts as the reference. Table 2 demonstrates the performance advantages attained by using multicore-embedded systems.

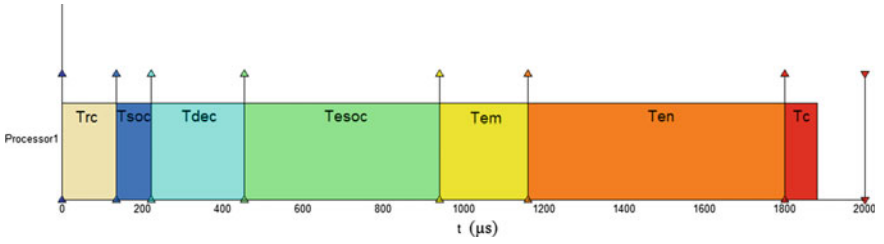


Fig. 4 Task trace of single-core-based HEMS

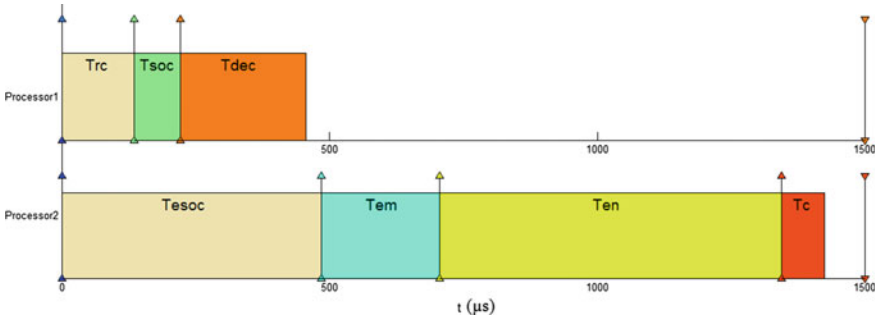


Fig. 5 Task trace of single-core-based HEMS

Table 2 Performance results

Tasks T	No. of cores n	Execution time tn (ms)	Speedup S = ts/tn	Efficiency E = S/n
7	1	1.88	1	1
7	2	1.42	1.32	0.63

The result shows the improvement in execution time of the tasks with a speedup of 1.32 in multicore processors. Another metric used to measure resource utilization is efficiency. The performance comparison reveals the advantages of using a multicore processor over a single-core processor used to implement HEMS in terms of runtime and speedup.

6 Conclusion

With the advent of new technologies, arrival of smart grid era and advancements in communication and information systems, energy storage systems based on smart home area networks would transform the electricity usage by means of efficient EMS. Present smart home with HEMS improves the energy efficiency, cost, and reliability of distribution systems. In this paper, a comprehensive view of the HEMS with its functional modules is presented, which could switch between energy sources depending on the load to be operated and communicate this encrypted energy consumption data to the data collection unit. Furthermore, the paper investigates the viability of HEMS implementation using multicore architecture, rather than a conventional single-core processor. The performance comparison between single-core and multicore implementation is carried out using various metrics, viz, task execution time, processor speed, and efficiency. The obtained results confirm the performance increase of multicore implementation with a speedup of 1.32, enabling simultaneous switching of energy sources and a timely communication of encrypted energy data. This work can be extended by dynamic scheduling of tasks in HEMS to meet the real-time demands of room controllers and different fault detection techniques can be implemented in HEMS.

References

1. IHS Markit: Rapid Expansion Projected for Smart Home Devices, http://news.ihsmarkit.com/press-release/technology/rapid-expansion-projected-smart-home-devices-ihs-markit-says
2. A. John, I.B. Santhosam, Home energy management system based on ZigBee. Int. J. Inven. Eng. Sci. **2**(4), 14–15 (2014)
3. M. Uddin, T. Nadeem, Energy sniffer: home energy monitoring system using smart phones, in *8th International Conference on Wireless Communications and Mobile Computing*, Limassol, Cyprus, pp. 159–164, Aug 2012
4. N. Dlodlo, A. Smith, L. Montsi, C. Kruger, Towards a demand-side smart domestic electrical energy management system, in *IST-Africa Conference and Exhibition*, pp. 1–12, May 2013
5. M. Inoue, T. Higuma, Y. Ito, N. Kushiro, H. Kubota, Network architecture for home energy management system. IEEE Trans. Consum. Electron. **49**(3), 606–613, Aug 2003
6. J. Han, C.S. Choi, I. Lee, More efficient home energy management system based on ZigBee communication and infrared remote controls. IEEE Trans. Consum. Electron. **57**(1), 85–89 (2011)
7. Y.S. Son, K.D. Moon, Home energy management system based on power line communication. IEEE Trans. Consum. Electron. **56**(3), 1380–1386, Aug. 2010
8. D. Prasanna Vadana, S.K. Kottayil, Energy-aware intelligent controller for dynamic energy management on smart microgrid, in *Power and Energy Systems*: *Towards Sustainable Energy* (Bangalore, 2014), pp. 1–7
9. J. Byun, I. Hong, S. Park, Intelligent cloud home energy management system using household appliance priority based scheduling based on prediction of renewable energy capability. IEEE Trans. Consum. Electron. **58**(4), 1194–1201 (2012)
10. T. Shoji, W. Hirohashi, Y. Fujimoto, Y. Amano, S.I. Tanabe, Y. Hayashi, Personalized energy management systems for home appliances based on bayesian networks. J. Int. Counc. Electr. Eng. **5**(1), 64–69 (2015)
11. N. Krishna Prakash, B. Surjith, Wireless sensor network based remote monitoring system in smart grids. Int. J. Control Theor. Appl. **9**(14), 6639–6646 (2017)
12. D. Yogavani, N. Krishna Prakash, *Implementation of Wireless Sensor Network based Multi-core Embedded System for Smart City*, vol. 10, no. 2 (2017), pp. 119–123
13. V.R. Arvind, R.R. Raj, R.R. Raj, N. Krishna Prakash, Industrial automation using wireless sensor networks. Indian J. Sci. Technol. **9**(11), 1–8 (2016)
14. R. Woo, S. Lee, E.J. Yang, D.W. Seo, Smart home system architecture for real-time and low standby power, in *Proceedings of IEEE 5th International Conference on Consumer Electronics* (Berlin, 2015), pp. 441–442
15. A. Munir, A. Gordon-Ross, S. Ranka, Multi-Core embedded wireless sensor networks: architecture and applications. IEEE Trans. Parallel Distrib. Syst. **25**(6), 1553–1562 (2014)
16. R. Ranjith, R. Shanmughasundaram, Simulation of safety critical applications for automotive using multicore scheduling, in *2015 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT)* (Kumaracoil, 2015), pp. 12–16
17. N. Krishna Prakash, D. Prasanna Vadana, Machine learning based residential energy management system, in *Proceedings of International conference on Computational Intelligence and Computing Research* (2017), pp. 684–687
18. W. Wang, J. Chen, F. Xu, An implementation of AES algorithm Based on FPGA. In: *2012 9th International Conference on Fuzzy Systems and Knowledge Discovery* (Sichuan, 2012), pp. 1615–1617

Implementation of Hindi to English Idiom Translation System

Himani Mishra, Rajesh Kumar Chakrawarti and Pratosh Bansal

Abstract Idiom plays a significant role in a language literature (Sinha in A system for identification of idioms in Hindi. IEEE, 2014 [1]). It enhances the glory of the language which contains it. As idiom points to a figurative meaning (Anastasiou in Idiom treatment experiments in machine translation, 2010 [2]), i.e., the meaning different from the word set present in that idiom, it becomes difficult for the machines to translate them. This paper proposes an achievable implementation way of Hindi to English idiom translation using a hybrid approach which consists of interlingual-based approach combined with the transfer-based approach. This paper also provides details on system requirements, algorithm, database connection, and result analysis.

Keywords Idiom · Language · Machine · Idiom translation

1 Introduction

India, being a multilingual country, has multiple languages for communication. But Indians emphasizes on Hindi for communication. All over the country, people widely speak Hindi, contributing to a huge figure of around 400 million speakers. This

H. Mishra (✉) · R. K. Chakrawarti
Shri Vaishnav Institute of Technology and Science, Indore, Madhya Pradesh, India
e-mail: himanimishra.hm21@gmail.com

R. K. Chakrawarti
e-mail: rajesh_kr_chakra@yahoo.com

P. Bansal
Devi Ahilya Vishwavidyalaya, Indore, Madhya Pradesh, India
e-mail: pratosh@hotmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_39

scenario arises a need for Hindi to English translators for connecting India with the world. Hindi is a morphologically rich language comprising of huge collection of poems, phrases, and idioms, which are difficult to translate. A group of words provide a different meaning when taken together as compared to when taken individually is called idioms [1–3]. Idioms furnish a language with its beauty, expressiveness, and elegance but also tenfold the difficulty of machine translation (MT); because they point to a different meaning than that of what their word set express [2]. This implementation paper provides a Web system to translate Hindi idioms to its equivalent English idioms using a hybrid approach.

2 Literature Survey

Today, we are in a digital world where every data is available digitally. It would be a mistake if we use manual translations rather than machine translation system (MTS) for translation of that zillion of information. For machine translation, we have surveyed some approaches, noteworthy contributions and available systems.

2.1 Approaches

To make a machine understand the language and meaning of the text and then to translate it, we are equipped with many MT approaches. These approaches provide the method to ease MT. Some machine translation approaches are *Corpus-Based MT*, *Rule-Based MT, and Hybrid MT* [4–6].

Rule-Based Machine Translation (**RBMT**). *RBMT* analyzes the input and creates a mediated representation of that input text. This intermediate representation is translated into the target language. It is classified as direct machine translation, transfer-based machine translation (TBMT), and interlingual-based machine translation (IBMT). It is easy to build an initial rule-based system and is effective for core phenomena. RBMT systems are difficult to extend. Anusaaraka systems (1995), Punjabi to Hindi MTS (2007–2008), Web-based Hindi-to-Punjabi MTS (2010) are examples of Direct MT. MANTRA Systems (1997), MAT System (2002), Shakti (2003) belongs to TBMT and ANGLABHARTI (2001), UNL-Based English-Hindi MT System (2001), Angla Hindi (2003) are some IBMT SYSTEMS [1, 4–9].

Table 1 Noteworthy contributions [2, 7]

System	Description
Schenk-1986	Rosetta—"Isomorphic grammar" was coined for idiom translation system
Santos-1990	PORTUGA—A parser that studies and produces the Multi Word Expression (MWE) due to lexical transfer was utilized
Wehrli-1998	ITS2—Here idiom was parsed, then subjected to lexical constraints and if satisfied, are retrieved
Ryu et al.-1999	An idiomatic expression recognizer—from to K/E was the new idea used in this idiom translation system
Krenn-2000	Collocations, identification, and representations of idioms using some tools were explained here
Franz et al.-2000	HARMONY—an idiom MTS based on EBMT approach was given
Poibeau-2001	Involved a bi-directional finite state automata (BFSA) for parsing the idioms in the system which was quoted in this research work
Fellbaum-2002	Work was dedicated to show information like status and representation related to the type of VP idioms

Hybrid Approach. Hybrid machine translation is the combination of two or more approaches, resulting in better translation by extracting the strengths of both the approaches and dropping out the weakness. ANUBHARTI-II (2004), Bengali to Hindi MT System (2009) are based on Hybrid approach [4–7, 9].

2.2 Noteworthy Contributions

Many researchers from around the world have discussed various ideas for developing MTS for idioms. Some of their remarkable concepts are listed (Table 1).

2.3 Online Translators

There are various types of machine translation systems available in multiple language pairs. They provide translation services with high accuracy. But when comes to translation of idioms alone or idioms in a sentence, even many renowned systems do not provide satisfactory result [6]. Some of the online machine translation systems are listed (Table 2).

Table 2 Online translators [6]

System	URL
India typing [10]	http://indiatyping.com/index.php/translations/hindi-to-english-translation
Soft112 [11]	http://english-to-hindi-and-hindi-to-english-converter-software.soft112.com/
SAMPARK [12]	http://ilmt.tdil-dc.gov.in/sampark/web/index.php/content
Language translators [13]	http://www.stars21.com/translator/english_to_hindi.html
Google translator [14]	https://translate.google.co.in/
Dictionary.com [15]	http://translate.reference.com/

3 Implementation

Idiom translation system contains two phases—comparison phase and translation phase. To understand the working of the system, idiom classification is one of the significant aspects to be considered.

3.1 Idiom Classification

Idioms are used globally in all languages, so the categorization came from wide range of scholars across the globe. Some of the major classifications are: *Grammatical idiom—Exragrammatical idiom*, *Idiom with—without Pragmatic point*, *Encoding idiom—Decoding idiom by Makkai* (*1972*), *Lexical Filled idiom—Lexically open Idiom by Fillmore* (*1988*) [2, 7]. Another classification that can be done on the basis of "*on the way the idioms can be translated*" [3, 7]:

Case I: *Idioms which have similar meaning and similar form in both the languages* (*source and target*): In this case, the idiom provides literal meaning in both languages. *For Instance*: Unity is strength— एकता ही बल है ।[16, 17].

Case II: *Idioms which have similar meaning but dissimilar form in both languages*: Consider an example To add fuel to fire— आग में घी डालना ।, [16, 17] as we can notice that the Hindi idiom is a normal translation of the word "vegetable oil" in English can be replaced with higher word classification, i.e., "fuel."

Case III: *Idioms with completely different meaning and different form in both the languages*: The Hindi and English idioms are completely different and do not share any similarity directly. *For Instance*: A drop in the ocean— ऊठ के मुँह में जीरा ।[16].

3.2 System Background

The major part of idiom falls in case III, which is the unresolved issue considered in this system design. There exists no direct relation between Hindi and English equivalent idioms. But they do share a relation or a bond which can be considered for translation. The "*meaning*" of any idiom is always the same in any language. This is the basis of design of this idiom MTS [7].

3.3 System Architecture

The implementation of the translation system is proposed utilizing a RBMT approach, by applying a combination of TBMT [1, 4, 7] and IBMT [1, 7]. There are two phases: comparison phase followed by translation phase [7] (Table 3).

Comparison Phase. As the user provides Hindi idiom, the system calls for a comparison algorithm, which searches for the input in a Hindi database which contains Hindi idioms and its meaning in Hindi. If the input is present that means the input belongs to case III and is forwarded to interlingual-based module; if the input is not present, it belongs to case I or case II and is sent to transfer-based module [4, 7, 8] (Fig. 1).

Table 3 System requirements

Component	Specification
RAM	2 GB
Operating system	Windows XP or higher
Front end	Microsoft visual studio 2008 or higher
Back end	Microsoft SQL server version-09.00.3042

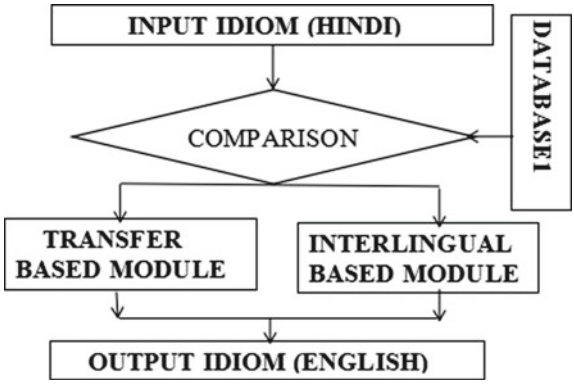


Fig. 1 Overall system architecture [1, 3, 7]

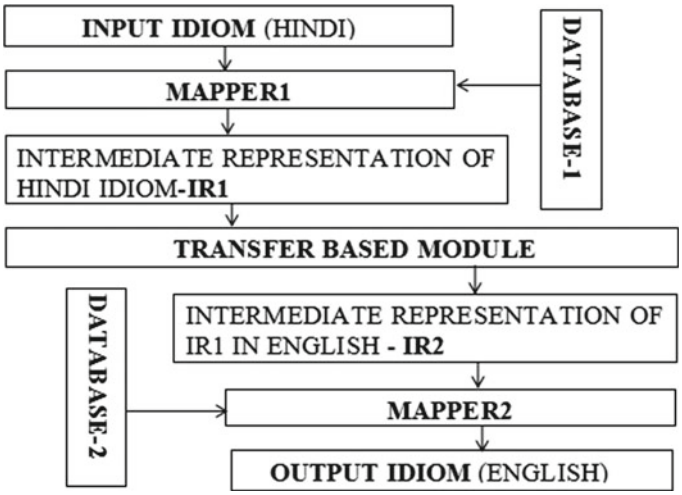


Fig. 2 Interlingual module [1, 3, 7]

Translation Phase. It starts with the call of either of the transfer-based [1, 7, 8] or interlingual-based module [7]. In either of these two modules, the Hindi idiom is presented to the specified algorithm and English equivalent idiom is generated [7].

Transfer-Based Module. It is based on TBMT. This module translates Hindi meaning of Hindi idiom to English meaning and handles Case I and Case II category idioms. It comprises of *Tokenizer*, *Parser and POS Tagger*, *Declension Tagger*. An input and output module for accepting user Hindi idiom input and displaying equivalent English idiom output will be present. An improvised dictionary will be needed during translation phase for translation of II category of idioms [1, 7, 8].

Interlingual-Based Module. It is based on IBMT. This is designed to handle idioms of Case III. Components of this module include *Input*, *Mapper*, *Database*, *Transfer-Based Module, and Output.* In interlingual-based module, the transfer-based module is used for generating English translation of Hindi idiom's meaning (Fig. 2).

3.4 Algorithm

The algorithm is used for the comparison phase and the translation phase.

```
String t1, Result, Output, Str1, Str2, Str3, Str4, Str5;
Hindi Database=H1; English Database=E1;
Step1:- User provides the input in textbox1 and is
stored in t1;
Step2:- Comparison() is called
Comparison(t1)
{If (H1 contains t1)
Result=Call interlingual();
Print Result;
Else
Result=Call string transfer(t1);
Print Result;}
Step3:- Either transfer() function or interlingual()
function is called.
(a) If the comparison phase calls Interlingual()
function
Str1=select h_mean from hindi where h_idiom ==t1;
Str2=Call string transfer(str);
Output=select e_idiom from english where e_mean=Str2;
Return Output;  OR
(b)if the comparison phase calls transfer() function
Str1=Parser(t1)
Str2=Declension(Str1);
Str3=Reordering(Str2);
Str4=Translator(Str3);
Str5=Morphological(Str4);
Return (Str5);
Step4:-English Idiom equivalent to input Hindi Idiom
is displayed.
```

3.5 Database

Database Connection. Before connecting to a database, we have to create and define necessary elements of the database schema. Following this, we connect to the database through our code.

```
SqlConnection con = new SqlConnection(@"Data
Source=.\SQLEXPRESS;AttachDbFilename=F:\IDIOM_TRANSLATI
ON\App_Data\Hindi_Database.mdf;Integrated
Security=True;User Instance=True");
DataSet ds = new DataSet();
SqlDataAdapter ad = new SqlDataAdapter("insert into
hindi(h_idiom,h_mean)
values(N'"+TextBox2.Text+"',N'"+TextBox3.Text+"')",con;
ad.Fill(ds);
```

Database Search. To search any idiom from the database, following code can be used:

```
SqlDataAdapter ad = new SqlDataAdapter("select h_mean
from hindi where h_idiom=N'"+TextBox1.Text+"'",con);
ad.Fill(ds);
```

4 Result Analysis and Discussion

In this implemented Hindi to English idiom translation system, the accuracy of the system depends on the richness of the database of interlingual module (case III of idioms) and efficiency of transfer-based module. To analyze the output of the system, we have divided the user idiom input into below three cases:

Case 1: **Input is an Idiom and available in the database** (Fig. 3). The input belongs to case III, and interlingual module will be called. The system will provide an accurate result.

Case 2: **Input is an Idiom but not available in the database1**. This case can be divided into two scenarios.

A. Input Idiom belongs to case I and case II of Idiom Categorization: The Hindi idiom will be sent to transfer-based module and will provide accurate results, i.e., English equivalent output of the given Hindi idiom input.

B. Input belongs to Case III of Idioms classification: These kinds of idioms should be accepted by interlingual-based module. But due to the absence of idioms in the database 1, it will be sent to a transfer-based module which will translate the idiom as an ordinary translation, not considering it as an idiom. This does not guarantee the proper translation of these cases of idioms.

Case 3: **Given Hindi input is not an idiom** (Fig. 4). The input does not fit in with the translation system. But it will be sent to transfer-based module which will provide the translation as of normal sentence. As the system is not designed for translation of ordinary sentence, it does not ensure the proper translation of these sentences.

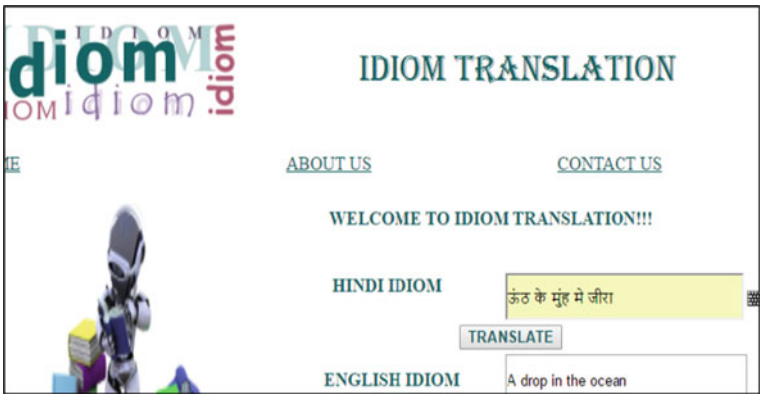


Fig. 3 Interlingual-based module result

Fig. 4 Input is not an idiom

Furthermore, if the input sentence contains idiom (as part of it) then too, it cannot ensure proper translation, as this system is not designed for extracting the idiom from a sentence and then translate it.

5 Conclusion and Future Work

This research work presents a proposal for effective Hybrid machine translation approach for translation of Hindi to English idioms using "interlingual approach" combined with "transfer-based approach." The system utilizes a very important aspect of idioms, i.e., their "meaning remains same in all languages" to translate from Hindi to English. This type of approach (interlingual based + transfer based) has not been used in any existing machine translation system to date.

This translation system is for Hindi to English idiom translation, but can be extended for different language sets, for instance—Hindi–Bengali, English–Gujarati, Assamese–Hindi [4]. The system can be embedded with other MTS to improve their efficiency in relation to idiom translation. Moreover, we can extend the idiom database for increasing the number of idioms (of case III) that can be translated using this translation system [7, 8].

References

1. R.M.K. Sinha, A system for identification of idioms in Hindi (IEEE, 2014)
2. D. Anastasiou, *Idiom Treatment Experiments In Machine Translation* (The University of Saarland, Germany, 2010)
3. M. Gaule, G.S. Josan, Machine translation of idioms from english to Hindi. IJCER **2**(6)
4. J. Nair, A. Krishnan, R. Deetha, An efficient English to Hindi machine translation system using hybrid mechanism, in ICANCCI (IEEE, 2016)
5. G.V. Garje, G.K Kharate, Survey of machine translation systems in India. IJNLC, **2**(4) (2013)
6. R.K. Chakrawarti, H. Mishra, P. Bansal, Review of machine translation techniques for idea of Hindi to English idiom translation. IJCIR **13**(5) (2017)
7. H. Mishra, R.K. Chakrawarti, P. Bansal, A new approach for Hindi to English idiom translation. IJCSE **9**(7) (2017)
8. A. Ghelot, V. Sharma, S. Singh, A. Kumar, Hindi to English transfer based machine translation system. IJACR **5**(19) (2015)
9. L.R. Nair, P.S. David, Machine translation systems for Indian languages. Int. J. Comput. Appl. **39**(1) (2012)
10. India Typing, indiatyping.com/index.php/translations/Hindi-to-English-translation
11. Soft112, English-to-Hindi-and-Hindi-to-English-converter-software.soft112.com/
12. SAMPARK (TDIL), ilmt.tdil-dc.gov.in/sampark/web/index.php/content
13. Hindi to English translator, translation2.paralink.com/Hindi-English-Translator
14. Googletranslator, translate.google.co.in/
15. Dictionary.com, translate.reference.com/
16. R.C. Phatak, A few English idioms with their hindustani equivanlents, in *Bhargava's Standard Illustrared Dictionary*, vol. 10
17. N.K. Aggarwala, *Essentials of English Grammar and Composition* (New Delhi, Goyal Brothers Prakashan, 2003)

Design and Development of Compact Super-Wideband Antenna with Integrated Bluetooth Band

Renu Jinger and Navneet Agrawal

Abstract A novel compact monopole antenna for super-wideband (SWB) applications with Bluetooth band is presented. The antenna is composed with a modified triangular-shaped patch, tapered feed and an L-shaped ground plane. For the Bluetooth band (2.4–2.48 GHz) by attaching an L-shaped strip to the radiator, an additional resonance is excited. We compare the proposed antenna with existing antenna (Torres et al in Microw Opt Technol Lett 59:1148–1153, 2017 [3]). The proposed antenna is having wider frequency band and smaller size as compared to the existing antenna (Torres et al in Microw Opt Technol Lett 59:1148–1153, 2017 [3]). The size of the antenna is only $13 \times 24 \times 1.58$ mm^3, and frequency band is from 3.1 to 39.5 GHz. The proposed antenna is useful for modern wireless communication applications.

Keywords Sierpinski structure · SWB antenna · Tapered microstrip feed line UWB antenna

1 Introduction

The antennas are required for modern wireless-enabled devices which should be small in size because the space available in such devices is limited. There are many techniques which have been applied to reduce the size and make them compact wideband antennas such as using diversities of reactive loaded technique [1]. On the other hand, the requirement of wireless wideband communications is continuously increasing because wideband communication supports more users and also provides more information and data rates. So, in this paper we develop a single antenna which covers many communication services.

R. Jinger (✉) · N. Agrawal
Department of Electronics and Communication Engineering,
College of Technology and Engineering, MPUAT, Udaipur, Rajasthan, India
e-mail: riyajinger@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_40

Table 1 Proposed SWB antenna and previously designed SWB antennas comparison

Description	Bandwidth (GHz) at S11 < −10 dB	Size W × L (mm^2)
Antenna based on reactive loaded technique [1]	2.43–32.39	25 × 26
Fractal antenna [2]	3–35	20 × 33.4
Antenna based on Sierpinski structure [3]	1.68–26	62 × 64
CPW-fed antenna [5]	2.4–24.3	30 × 41
Hut-shaped printed antenna [7]	0.9–22.35	25 × 40
Semicircular-shaped SWB antenna [8]	1.30–20	42 × 52.25
Proposed antenna	3.1–39.5	13 × 24

Super-wideband (SWB) technology has higher channel capacity and provides higher video, voice and data transmission. Super-wideband antennas must have a bandwidth ratio larger than 10:1 for impedance bandwidth −10 dB return loss.

There are many designs which have been studied by a number of researchers to enhance the value of impedance bandwidth such as using fractal antenna [2], Sierpinski structure [3], different stub shapes [4] and by using feed modification such as using CPW-fed antenna [5], tapered microstrip feed line [6]. In Table 1, the comparison between some designs used for SWB antenna configuration with proposed antenna is given.

In this paper, we have designed a simple and small triangular-shaped antenna with a greatly improved bandwidth and smaller size which is better than those in previous studies [1–8] and integrated Bluetooth band is demonstrated. The SWB performance and compact size of the present design are realized by a simple triangular-shaped patch fed by tapered shaped microstrip feed line and L-shaped ground plane. Further, an L-shaped strip is added to the radiator to excite the Bluetooth band. The antenna is having frequency band from 3.1 to 39.5 GHz.

It supports many existing wireless bands, such as ISM (2.4–2.48 GHz), Wi-fi (2.4 GHz), GPS (2.4 GHz), Bluetooth (2.4–2.48 GHz), WLAN (2.4–2.48 GHz), UWB (3.1–10.6 GHz), WiMAX (3.3–3.7 GHz), C band (3.8–4.2 GHz), WLAN (5.15–5.85 GHz), K and X bands (8–12 GHz) and UWB vehicle radar (22–29 GHz).

2 Antenna Design and Geometry

2.1 Antenna Configuration

Figure 1a displays the top view, while Fig. 1b shows the bottom view of the proposed antenna. The proposed antenna with a compact size of 13×24 mm^2 is fabricated on epoxy FR4 with the thickness, dielectric constant and loss tangent values being 1.58 mm, $\in r = 4.3$ and $\tan \delta = 0.025$, respectively. The other parameters are L = 24 mm, W = 13 mm, F = 3 mm, l1 = 14 mm, l2 = 4.61 mm, l3 = 2.30 mm, l4 = 5.60 mm, w1 = 11 mm, w2 = 2 mm, f = 0.90 mm, G = 6.80 mm, g1 = 7.60 mm, g2 = 5.74 mm, g3 = 19.69 mm, g4 = 5.26 mm, g5 = 1.50 mm as shown in Fig. 1.

2.2 Different Geometries Used in Evolution of the Final Designs

Different geometries used in the evolution of the final design are shown in Fig. 2a, b, c. In Fig. 2a, the antenna is having a linear tapered feed line with a simple triangular-shaped radiator on the one side of the substrate while on the other side of the substrate has partial ground plane. The base of the triangular patch is W, and perpendicular is l1 as shown in Fig. 1a.

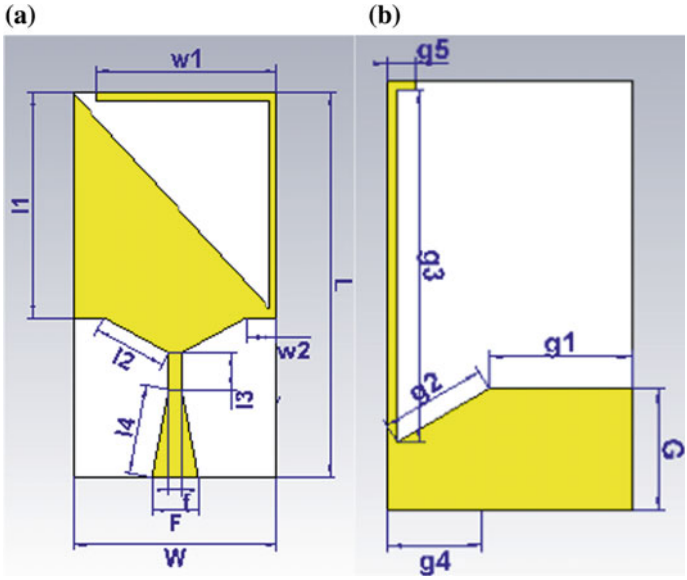


Fig. 1 Configuration of the proposed antenna (top view and bottom view)

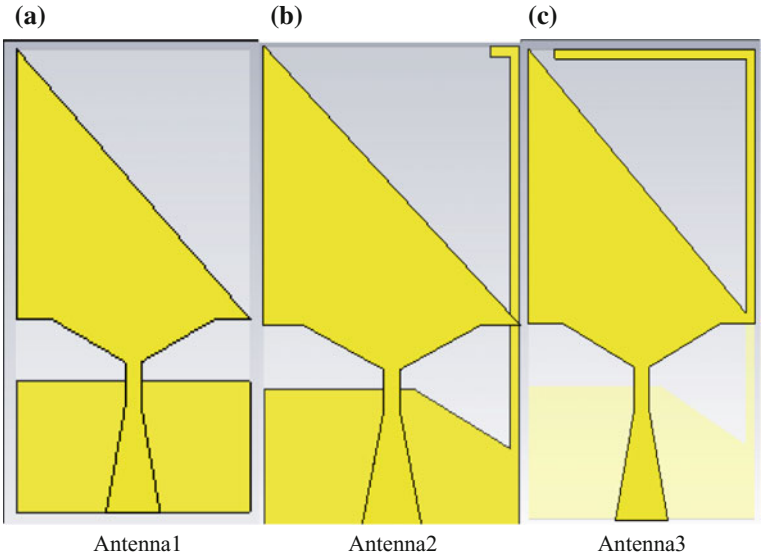


Fig. 2 Different geometries used in the evolution of the final design

In Fig. 2b modified the ground plane by using L-shaped of the ground plane is to get the improvement in impedance bandwidth and for SWB performance. After achieving the SWB performance, the antenna is further modified to integrate the Bluetooth band. In Fig. 2c, a simple L-shaped strip is attached to the radiator to excite the Bluetooth band (2.4–2.48 GHz). The width of L-shaped strip is added to both the radiators, and the ground is 0.5 mm. Figure 1 shows the parameter values of the proposed antenna.

3 Results and Discussion

3.1 Comparison of S Parameter for Different Geometries Used in Final Design

In Fig. 3, the simulated S11 parameter plot is shown for all the geometries used in the evolution of the final design. As displayed in Fig. 3, the plot of return loss in Antenna 1, Antenna 2 and Antenna 3 is for Fig. 2a, 2b and 2c, respectively. In Antenna 2, the ground plane is modified compared to Antenna 1 so we get the improved result, and as shown in the plot of Antenna 3, the resonance for Bluetooth band is achieved through an additional L-shaped strip that is attached to the triangular radiating patch. The resonating length includes the hypotenuse of the radiating patch

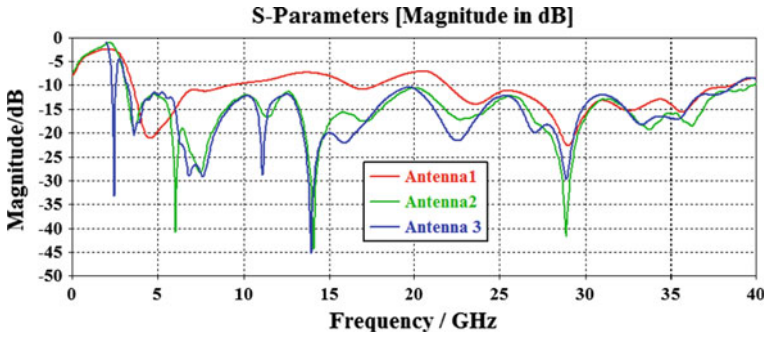


Fig. 3 Return loss (S11) plot for all the different geometries used in the evolution of the final design

and the strip attached. At the resonant frequency, this length should be 1/2 of the guided wavelength for the Bluetooth frequency. Therefore, the resonating length can be written as

$$L_b = \sqrt{l1^2 + w1^2} + l1 + w1 \tag{1}$$

where the values of l1 and w1 are given in the antenna configuration.

The resonant frequency f_b may be empirically approximated by

$$f_b \approx \frac{c}{2L_b\sqrt{\in_{reff}}} \tag{2}$$

where c is the speed of light and $\in_{reff}$ is the dielectric constant (effective). The effective dielectric constant, due to the lack of ground plane, is found to be half of the dielectric constant of the substrate. Hence, the effective dielectric constant is approximated as 2.15. Thus, for the Bluetooth band, the calculated length L_b is 42.8 mm. So by changing the value of $w1$, we can get the different resonance frequencies. By taking the value of $w1 = 11$ mm, we can get resonance at Bluetooth band. The proposed antenna shows an operation frequency range from 3.10 to 39.5 GHz, which represents a bandwidth ratio larger than 12:1 and also gets integrated Bluetooth band from 2.4 to 2.48 GHz.

3.2 Comparison of Existing Antenna with Proposed Antenna

The comparison of the return loss of existing antenna [3] and the proposed antenna is shown in Fig. 4. The existing antenna [3] is a fractal antenna based on the Sierpinski structure.

We can see that the existing antenna [3] is having frequency band from 1.68 to 26 GHz but the proposed antenna is having frequency range from 3.10 to 39.5 GHz.

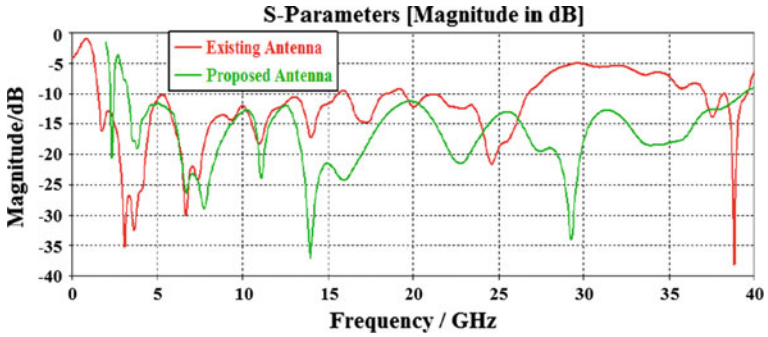


Fig. 4 Return loss comparison between existing antenna and proposed antenna

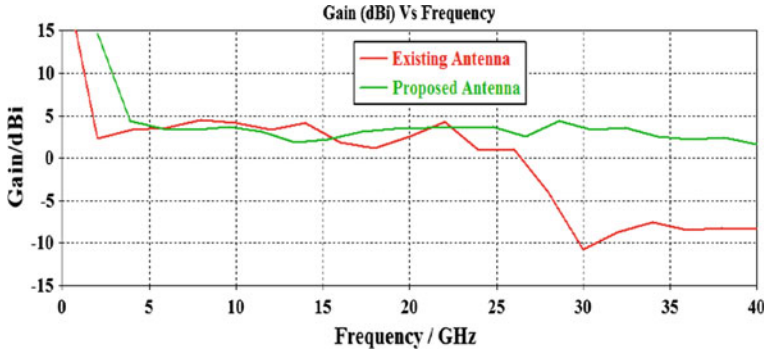


Fig. 5 Gain comparison between existing antenna and proposed antenna

Size of the existing antenna [3] and proposed antenna is 13×24 mm^2 and 62×64 mm^2, respectively. So we get simple structure and reduced size with higher bandwidth as compared to existing design [3].

Figure 5 shows the existing and proposed antenna gains. It is found that the gain of the antenna varies from 2 to 5 dB over the Bluetooth and SWB frequency range.

Figure 6 shows the VSWR of the existing and proposed antennas. The VSWR of the proposed antenna is less than 2:1 over the entire SWB frequency range.

4 Conclusion

A small size, triangular-shaped SWB antenna (monopole) with integrated Bluetooth band is proposed and implemented. The Bluetooth band was realized by adding an L-shaped strip to the radiating patch. The antenna has a wide bandwidth, and the impedance bandwidth ratio is found to be 12.74:1, covering frequency from 3.1 to 39.5 GHz with integrated Bluetooth band. The proposed antenna is compared

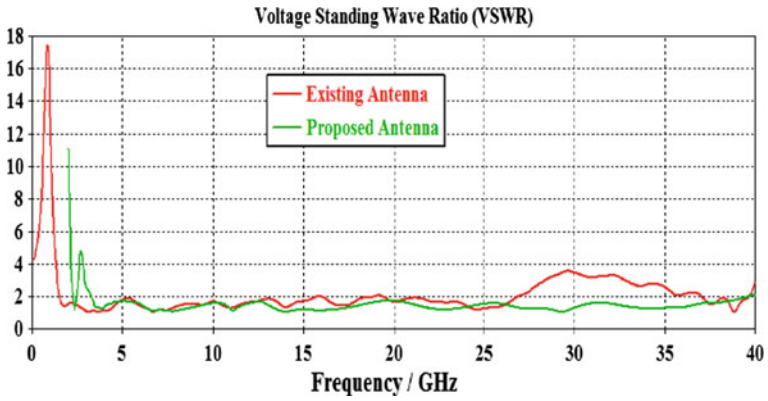


Fig. 6 VSWR comparison between existing antenna and proposed antenna

with an existing antenna [3] in terms of size, frequency band, return loss, gain and VSWR. We found that the proposed antenna is far better than the existing antenna [3]. The proposed antenna can support many wireless services and making possible its integration into mobile phones and other portable devices, due to the compact dimensions of the antenna (13×24 mm^2) and also finding its application in wideband and multi-band wireless devices.

Acknowledgements The author would like to extend their sincere gratitude to the mentor along with Head of ECE department, CTAE, MPUAT, Udaipur, Rajasthan, for supporting this work under M.Tech. programme of the Department of ECE.

References

1. S.Z. Aziz, M.F. Jamlos, Compact super wideband patch antenna design using diversities of reactive loaded technique. Microw. Opt. Technol. Lett. **58**, 2811–2814 (2016)
2. B.L. Shahu, S. Pal, N. Chattoraj, Design of super-wideband hexagonal-shaped fractal antenna with triangular slot. Microw. Opt. Technol. Lett. **57**, 1659–1662 (2015)
3. C.A.F. Torres, J.L.M. Monroy, H.L. Morales, R.A.C. Perez, A.C. Tellez, A novel fractal antenna based on the sierpinski structure for super wide-band applications. Microw. Opt. Technol. Lett. **59**, 1148–1153 (2017)
4. S. Barbarino, F. Consoli, Study on UWB and SWB planar slot antenna with different stub shapes. Microw. Opt. Technol. Lett. **53**, 1528–1532 (2010)
5. C. Deng, Y.J. Xie, P. Li, CPW-fed planar printed monopole antenna with impedance bandwidth enhanced. IEEE Antennas Wirel. Propag. Lett. **8**, 1394–1397 (2009)
6. S. Yadav, A.K. Gautam, Design of band-rejected UWB planer antenna with integrated bluetooth band. IET Microw., Antenna Propag. **10**, 1528–1533 (2016)
7. F.A. Tahir, A.H. Naqvi, A compact hut-shaped printed antenna for super-wideband applications. Microw. Opt. Technol. Lett. **57**, 2645–2649 (2015)
8. M. Samsuzzaman, M.T. Islam, A semicircular shaped super wideband patch antenna with high bandwidth dimension ratio. Microw. Opt. Technol. Lett. **57**, 445–452 (2015)

Customizing Lineage for Different Embedded Devices

Alok Sharma and Sunil Nimawat

Abstract Android is a rampant success in the mobile market creating various opportunities. Since evolution adds up various user experience and efficiency for devices and integration of this OS in various devices has also expanded, because of the promising trust and have other features like better UI, better efficiency, better device performance and consistent APIs [1], but gradual decrease in the performance of the devices can been seen due to various applications. This research tries to sway the user's opinion about various custom ROMs with the possibility of increasing the performance as well as efficiency of crippled devices by using various modification of the system source before compiling it for a particular device. We presented the modification of hardware abstraction layers (HALs) which will limit the use of a particular component, i.e., CPU, so that the maximum computing power is used on other important applications and proposed a method to improving the battery life of a particular device by proposing new I/O scheduler and CPU governor for improving scheduling by on-demand application as in result reducing the power usage.

Keywords Lineage · CPU governor · I/O scheduler · Stock ROM
TWRP recovery · Root access · AOSP

1 Introduction

Android was introduced on September 23, 2008, by Google for various devices like smartphones, setup boxes, locomotives, various enterprise use, various machines used in industry which are based on Linux architecture and ARM-based boards [2]. Due to the similarity between Linux and Android that they are free and open source, it is becoming the most widely used mobile OS these days which encourages OEMs to base their mobiles for the Android devices [3].

A. Sharma (✉) · S. Nimawat
IES IPS Academy, Indore, Madhya Pradesh, India
e-mail: chip@doyl.in

S. Nimawat
e-mail: sunilnimawat@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_41

According to the verge android has covered more than 80% of total shares in smartphone market. Various applications (e.g., Google Play Services, gaming, entertainment, and various communications) can be installed by the user so as to simplify the daily ongoing life purposes and basic life necessity. While the application is being implemented, constant use can arise some vulnerability. To be precise, memory needed by the application when not available causes subtle vulnerability like memory leaks [4] that may result in application crash. It happens when an allocated object lives longer than the expected lifetime inside an activity which creates reference of unused memory objects (e.g., redrawable framework layouts) thus denying request for new memory allocation which are caused by various vendor optimization with all the preloaded application [5] thus by improving the RAM utilization by Android application [6] by LMK [7] and OOMK [8] and also by disabling memory swapping so as to reduce the sluggish response so create low application cache for the utilization [9]. Android has a built-in memory manager regularly checking for garbage collection; regardless of this, the adequacy of managing the resource allocation table is not fully optimized. Smartphones are computers for general use with a phone attached to it, but it has its uses like storing data and communicating to other user [10, 11]. Android along with iOS smartphones allows the user to install the third-party application as the user seems fit.

Android has a compiler which helps the system apps to open faster. JavaScript and Android browser are also faster with the added support of Adobe Flash with performance boost in later versions by creating the VFS which retrieves the files faster, thus boosting the devices up to 66% faster [12].

Source of Android is open to all the users and thus available freely for improving and developing [13]. Since Android has friendly commercial license, one can do whatever changes and improvement as he seems fit, without the intention of blaming Google, if the devices are bricked or damaged but with one exception, i.e., the kernel based on Linux architecture, which falls under the GNU Public License thus the manufacturers had to release their device's Linux kernel source code and OEM code after the product is shipped. Various types of operating system based on Androids like AOSP [2] and CyanogenMod can also be modified for various embedded devices and various purposes like anti-forensic [14] and ported to other Linux-based machine.

2 Problem Identification

Android is being widely used in mobile phones such as smartphones and various other ARM devices and with every version of android we get more patches and powerful versions which support new functionality, more multimedia processing, and more storage thus by increasing more number of problem day by day some of which are:

- Battery usage due to lots of application and their large size.
- I/O and CPU governor scheduling due to large number of simultaneous application running in background.

3 Proposed Solution

Better Usage by Reducing Wakelock—To avoid battery drain while a device is left in idle condition, it quickly falls asleep, but sometime application needs to wake up the CPU or screen to complete the remaining work. Thus by removing the new wakelock reference counter variable the loop will be eliminated and class toggle in which three variable will be defined i.e. held acquire or release so in absence of reference counter variable the default setting for this wakelock system is to sleep only when system is idle for long duration of time.

Brightness—The brightness of various screens of devices lies between various range depending on the max and min values provided in OS thus changing these values in the integrated source code by which brightness adjustment factor i.e. allowing change of specific led brightness between a permitted range will change the battery drain.

Power Whitelist—Some application which is generally used by a user can be added to the whitelist so the system could keep the application awake to minimize the wakelock trigger count which can minimize the waste memory lock [15] in the primary memory. So by adding an application to the default whitelist to enable a particular receiver instead of loading the whole application into the memory. This will reduce the wakelocks and thus reducing the battery drain [16].

Idletimer—The idletimer is a function in which the OS cannot find the specific time in which the OS will put every application, broadcast receiver, various services, etc., on hold till the next wakelock arrives. So by adding the idletimer to the whole system will go to deep sleep while some receivers would be active because they are defined in whitelist.

Limiting Step-Up Frequency—The step-up frequency of an operating system is not defined so when system calls a wakelock or an application is triggered by the user, the CPU diverts all the power for maximizing the CPU frequency [17] for the user resulting in battery drain, thus limiting the step-up frequency for reducing the consumption of power and enabling all cores even for a single small application.

PROPOSED ALGORITHM-1	PROPOSED ALGORITHM-2
ENABLE IDLETIMER DEFINE BLINKRATE DEFINE LED ADJUSTMENT (R, G, B) DEFINE BRIGHTNESS FACTOR (PERMITTED RANGE 0-255) DEFINE WHITELIST INCLUDE PACKAGE IF WAKELOCK (IS HELD) RELEASE () PROCESS () DESTROY () ENDIF	IF CPU IDLE CPU RELAX () CHECK FOR TASK () IF ONLINE RETURN ACTIVITYMANAGER () DEFINE FREQUENCY TABLE GET CLKRATE AND RESUME WAKELOCK () SWITCH PROFILE POWER SAVER BALANCED HIGH PERFORMANCE ACQUIRE NODES & DATA PROCESS NODES & DATA RELEASE NODES & DATA FLUSH NODES & DATA

4 Results

4.1 Test Environment

In this research, a Sony Xperia Z5—E6653 has used to implement the proposed method. The devices have the following configuration:

CPU—Octa-core (4 × 1.5 GHz Cortex-A53 and 4 × 2.0 GHz Cortex-A57)
RAM—3 GB
Inbuilt Storage—32 GB
Android Version—Marshmallow 6.0.

4.2 Test Results

Before the compilation of OS, various changes have been made on source and it is time to test various improvement alongside with various effects of these improvement.

To illustrate various improvements, the snapshot at various instances as well between stock ROM, custom ROM, and optimized custom ROM is given below.

Wakelock—The figure below shows the wakelock counts in number/hr by Google services on Better BatteryStats pro. After implementing the above algorithm, wakelock in stock ROM, custom ROM, and optimized-custom ROM can be seen in Fig. 1.

Battery—The figure below shows the battery loss comparison in %/hr in stock ROM, custom ROM, and optimized-custom ROM. After implementing the algorithm, the battery loss in stock ROM was 5.6, 0.5%/hr in custom ROM and 0.1%/hr in optimized-custom ROM. The application we used for performing test is BetterBatteryStats Pro (Fig. 2).

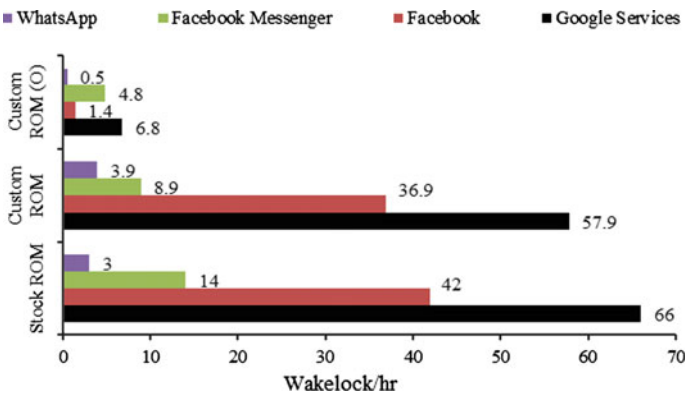


Fig. 1 Wakelock comparisons

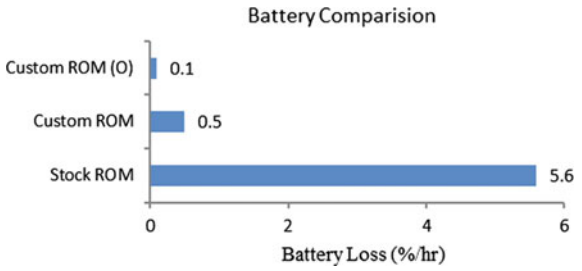


Fig. 2 Battery comparisons

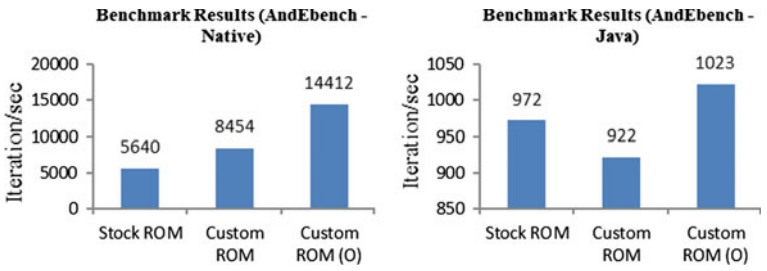


Fig. 3 Benchmark comparison

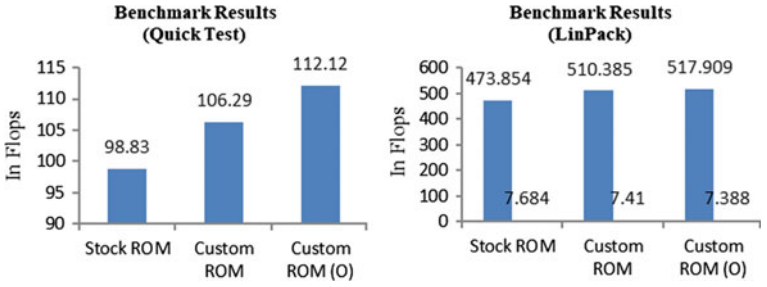


Fig. 4 Benchmark comparison

Benchmark Results—On implementing the proposed algorithm, increase has been seen in the performance of the device. Figures 3 and 4 illustrate the comparison of performance by And Ebench, Quicktest and Linpack in stock, custom and optimized-custom ROM.

RAM Comparison—On implementing the proposed algorithm, the significant increase in the performance of the device can be seen. The amount of ram free in stock custom and optimized-custom ROM can be seen in Fig. 5.

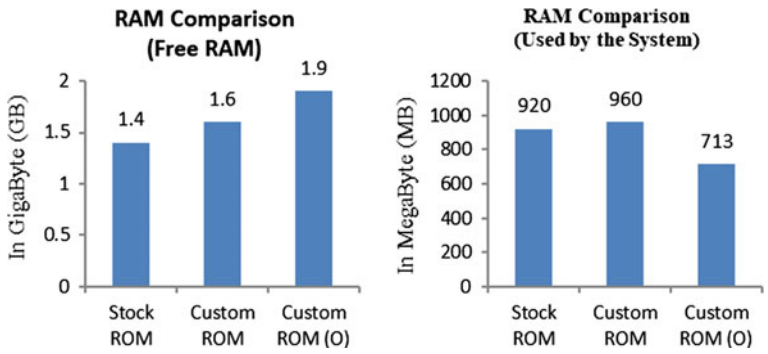


Fig. 5 RAM comparison

5 Conclusion and Future Scope

In the above testing on various parameters in various applications, it can be seen that custom OS is better than stock ROM but an optimized custom ROM is better than simple custom ROM; thus, improvement in various factors can be seen. However, there may be any unwanted results or bugs that can occur at the times and may even cause application crash because the optimized ROM has not been extensively tested in various other fields not known to us.

Performing the changes in basic structure of the OS is quite challenging and is a very complex method. This research demonstrates that all the methods implemented to improve various features will reduce the battery drain, improve CPU performance, and will provide a better user experience; however, it may also cause instability in the system and cause the hardware crash or failure and is not perfect in some respects. These methods would not cause an imminent failure but may have some effects on the user's experience. The research empirically supports that a user must not make the changes not knowing the results. Thus, in future, these researches must focus on improving I/O schedulers like noop, deadline, and various governors like pegusq, on-demand, interactive and to improve stability of the OS.

References

1. S. Manjrekar, R. Bhati, Custom ROM—a prominent aspects of android. Int. J. Adv. Res. Comput. Eng. Technol. (IJARCET) **5**(5), 1590–1593 (2016)
2. K.V. Charan, S.P. Sharmila, A.S. Manjunath, Customizing AOSP for different embedded devices, in *2014 International Conference on Computing for Sustainable Global Development (INDIACom)* (2014), pp. 259–264
3. A. Shanker, S. Lal, Android porting concepts, in *2011 3rd International Conference on Electronics Computer Technology (ICECT)*, vol. 5 (2011), pp. 129–133

4. H. Shahriar, S. North, E. Mawangi, Testing of memory leak in android applications on high-assurance systems engineering (HASE), in *2014 IEEE 15th International Symposium* (2014), pp. 176–183
5. C. Stach, B. Mitschang, Privacy management for mobile platforms—a review of concepts and approaches, in *2013 IEEE 14th International Conference on Mobile Data Management* (2013), pp. 305–313
6. D. Kayand, U. Shrawankar, Performance analysis for improved RAM utilization for Android applications, in *2012 CSI Sixth International Conference on Software Engineering (CONSEG)* (2012), pp. 1–6
7. H.J. Yoo, S.J. Kim, M.S. Jung, Study of garbage collection performance on Dalvik VM heap considering real-time response, in *2013 International Conference on IT Convergence and Security (ICITCS)* (2013), pp. 1–3
8. G. Lim, C. Min, Y.I. Eom, Enhancing application performance by memory partitioning in Android platforms, in *2013 IEEE International Conference on Consumer Electronics (ICCE)* (2013), pp. 649–650
9. M. Ju, H. Kim, M. Kang, S. Kim, Efficient memory reclaiming for mitigating sluggish response in mobile devices, in *2015 IEEE 5th International Conference on Consumer Electronics—Berlin (ICCE-Berlin)* (2015), pp. 232–236
10. H. Hu, J. Song, Integration and optimization of Android applications based on service-oriented architecture, in *2016 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)* (2016), pp. 2098–2103
11. B. Henne, C. Kater, M. Smith, M. Brenner, Selective cloaking: need-to-know for location-based apps, in *2013 Eleventh Annual Conference on Privacy, Security and Trust (PST)* (2013), pp. 19–26
12. Z. Wenxuan, L. Hang, Y. Xia, Z. Fangjie, Fastboot and fast shutdown of android on the embedded system, in *2013 IEEE 11th International Conference on Electronic Measurement & Instruments (ICEMI)*, vol. 2 (2013), pp. 1003–1008
13. T.S. Fernandes, E. Cota, A.F. Moreira, Performance evaluation of android applications: a case study, in *2014 Brazilian Symposium on Computing Systems Engineering (SBESC)* (2014), pp. 79–84
14. K.J. Karlsson, W.B. Glisson, Android anti-forensics: modifying cyanogenmod system sciences (HICSS), in *2014 47th Hawaii International Conference* (2014), pp. 4828–4837
15. B.G. Joo, S.M. Kim, *A User's Experience in Optimizing Smartphone Performance Using Overclocking and Memory Cleaning Techniques* (2012), pp. 127–138
16. K. Kapetanakis, S. Panagiotakis, Efficient energy consumption's measurement on android devices, in *2012 16th Panhellenic Conference on Informatics* (2012), pp. 351–356
17. K. Vimal, A. Trivedi, A memory management scheme for enhancing performance of applications on Android, in *2015 IEEE Recent Advances in Intelligent Computational Systems (RAICS)* (2015), pp. 162–166

Structural Coverage Analysis with DO-178B Standards

Parnasi Patel, Chintan Bhatt and Darshan Talati

Abstract Software testing is one of the most important ways to protect civil aviation safety and reliability of software for airborne equipments. D-178B/C standards are used to assure safety of avionics software and control systems and provide certificates according to safety criteria. This paper describes two different phases to achieve structural coverage analysis using DO-178B/C standards. Analysis of structural coverage can be done using to capture the amount of code which is covered of the airborne software. The first phase which contains the instrumentation procedure which instrument the source code at execution time and second phase is generating a report which specifies that which portion of source code is executed and which one is not in the form of percentage. Implementation is done for first metric which is statement coverage.

Keywords Avionics software testing · Code coverage · Decision/branch coverage
DO-178B/C · HTML report · Instrumentation probes · MC/DC coverage
SCA tool · Statement coverage

1 Introduction

DO-178B/C provides guidance and standards to software development, verification, configuration management and the interface to approval authorities. It prescribes a procedure which is being followed in the software development of airborne systems by developers [1]. Based on a system safety determination, some criteria

P. Patel (✉) · C. Bhatt
U & P U Patel, Department of Computer Engineering, CSPIT, CHARUSAT, Changa, India
e-mail: parnasipatel@gmail.com

C. Bhatt
e-mail: chintanbhatt.ce@charusat.ac.in

D. Talati
EInfochips Ltd., Ahmedabad, India
e-mail: darshan.talati@einfochips.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_42

Table 1 Failure conditions for structural coverage criteria

Level	Failure conditions	Structural coverage criteria
A	Software resulting in catastrophic failure condition or cascading system failure	MC/DC coverage Branch/decision coverage Statement coverage
B	Software resulting in hazardous failure condition	Branch/decision coverage Statement coverage
C	Major system failure	Statement coverage
D	Minor system failure	None required
E	No effect on system	None required

Table 2 Software quality assurance levels for structural coverage

Level	Coverage criteria	Measures
A, B, C	Statement coverage	Each statement of source code is evaluated
A, B	Branch/decision coverage	Each branch condition/decision is evaluated
A	Modified condition/Decision coverage (MC/DC)	Every entry and exit points are evaluated Each branch and each condition in decision are evaluated

are prescribed according to the failure conditions upon different situations. Consider Table 1 which describes the failure condition categories associate with structural coverage criteria.

Structural coverage determines that which structure of the software or which portion is not exercised. Structural coverage analysis for any test code is one of the approaches to measure the quality of any test suit. Depending upon the RTCA standards, DO-178B and DO-178C define different software quality assurance levels for structural coverage which indicated into Table 2.

There are three main different metrics/criteria considered within DO-178B which are considered as statement coverage, decision coverage and modified condition/decision coverage which are described as below.

1.1 Statement Coverage

Statement coverage is a criterion which involves execution of all exercised statements at execution time of test suit at least once. Machine instructions are considered as statement for any test suit which is written in assembly language.

1.2 Decision/Branch Coverage

Decision coverage is considered as each and every entry point, and exit point of test suite has been exercised at least once ate execution time. Every decision in the source code has considered all possibility at least once.

1.3 Modified Condition/Decision Coverage (MC/DC)

MC/DC can be defined as every entry and exit point of test suit and has been exercised at least once at execution time, every condition within a decision has been taken all possible outcomes at least once, every decision in test suit has been taken all possible outcomes at least once and each condition within a decision has been shown independently in decision's outcomes.

Calculate Structural Coverage

Structural coverage can be calculated from given formula,

$$Structural\ Coverage = \frac{Number\ of\ executed\ Statements\ *\ 100}{Total\ number\ of\ Statements}$$

2 Literature Survey

DO-178B provides guidance to developers for getting certification for airborne systems and equipment with software requirement suit. A number of documents are available for structural coverage. The primary objective of structural coverage analysis verifies all requirements and determines which portion of test suit is exercised and which is not. DO-178B provides different criticality levels that must be achieved by any test software.

Requirement-based testing (RBT) is one of the structural coverage techniques. RBT is used to achieve structural coverage testing by combination of requirement-based testing outcomes and their coverage analysis. The code, which is not covered, is analysed using a number of other criteria. This combination of testing results and analysis provides the actual structural coverage [2].

According to Rapita Systems Ltd., who publishes a white paper: 'Seven Roadblocks to 100% Structural Coverage (and how to avoid them)', specify a number of reasons that why it may not be possible to achieve 100% structural coverage. In practice, these usually conspire to make it very rare to achieve full coverage. To reduced effort for certification activities, different tools offer different file format like CSV, text, XML, html files [3].

According to the FAA's final report for Software Verification Tools Assessment Study, it gives a description of test cases for evaluating a tool correctly which determine statement coverage for Level C or not and check conditions [4] which are described below:

- Statement coverage for test case is indicated correctly.
- Test cases are correctly detected.
- Test cases in subprograms are correctly detected.
- Test cases which contain statement sequences in test suits are properly managed and detected.
- Compound statements are correctly managed.
- Conditional statements are properly managed.
- **if** statements are managed, including its other variants such as **if**-**else** statements and **if**-**else if**-**else** statements.
- Case statements are correctly managed.
- Coverage report is represented in the presence of described mechanisms and display correctly in its format.
- Test cases that do not execute due to **break** statements or **exit** statements are managed properly.

According to the CarFast's approach, they measure the speed with statement coverage, is achieved both in the number of test runs of the application under test the AUT test with multiple test case inputs such as multiple time execute same application with different input test cases and measures their execution time. While the passed time provides the absolute value of the time, it takes to reach a certain level of coverage. Measuring the number of iterations which essentially means that achieving coverage goals with number of test cases [5].

According to Zhu's software test units for coverage, they have two main program groups. These two program groups are based on the structural criteria of different test adequacy. First is the control flow criteria and second is data flow criteria. These two kinds of criteria are combined and extended to give another dependent coverage criteria. Most sufficient criteria for both of these group are flow graph model of program structure. However, a few control flow criteria are defined as test requirements in terms of code rather than using any other software abstract model [6].

According to Namin's, they provide a solution in which they determine how the size of code and coverage both are individually given their impact to the test suit for overall software. They generate different test suits of some fixed size and measure the relation between coverage and fixed size of test suits. Increasing test suit's size which directs cause to achieve higher effectiveness and higher coverage outcomes [7].

3 Framework Overview

Our framework is designed to be flexible and implement the statement coverage test cases. Table 3 describes the basic infrastructure for the procedure.

Figure 1 shows the actual flow of the procedure. It contains two main parts from which one is instrumentation procedure and other is to generate a report for given test suit. Structural coverage contains main two phases from first is instrumentation of source code and another is report generation. Here assume that tester gives source code and test cases to coverage tool, and instrumentation procedure is done and generated instrumented code folder along with another log files. After the instrumentation, execution of generated instrumented code is done. After the execution, tester will get the actual data or code which gives the executed portion of test suit. Another main phase is report generation in which original source code along with coverage data is given as input to report generation mechanism to generate coverage report. Coverage report can be generated for each file of the folder.

Table 3 Infrastructure for the process

Phase	Task
Instrumentation	Object level instrumentation is done Generate instrumented folders along with log files
Run test	Run instrumented folder Generate executed probes file
Report	Generate analysis reports

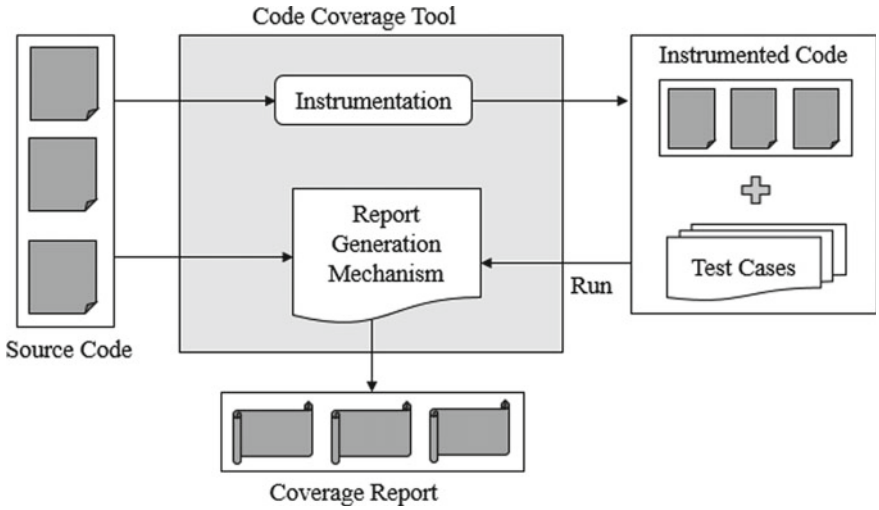


Fig. 1 Framework for structural coverage analysis

4 Procedure

4.1 Probe Instrumentation

Here we discuss the procedure that how actually tool performs instrumentation procedure. Instrumentation is the process to add additional test suit at the execution time to keep the track of executed portion of the test suit. Instrumentation can be done at source-level instrumentation or at intermediate language-level instrumentation.

- ***Source-Level Instrumentation***: In this, instrumentation code is inserted by the coverage tool to the input test suit. This will be compiled into the input code at execution time.
- ***Object-Level Instrumentation***: It is a post-compilation of test suit in which executable lines are collected to obtain coverage data, and they are injected into the object code. Test suit is inserted with instrumented probes without any modification in its original code or program. At the execution time, test cases trigger these probes. These probes give the actual location of executed portion of test code.

4.1.1 Probe Instrumentation Issues

Discussed points are considered as instrumentation issues which are based on probe instrumentation of executable specifications.

- ***Type of Coverage***: Probes are generally considered as counters which are used in program or test suit. Generally, these probes give the actual line count with its appropriate file. It is easier to measure the coverage in terms of line count from which we can determine the executed potion.
- ***Optimization***: To minimize the performance and behavioural impact of the instrumentation, the number of probes should be kept to a minimum, and the probes need to be inserted at the most appropriate locations in the specification or in the test suit.

4.1.2 Probe Instrumentation Technique

Instrumentation is done at particular location into a test suite. Instrumentation is done at the beginning of any function definition/condition declaration. Here we used one function as instrumentation probe/potion which gives the actual executed locations of lines of test suite at run-time. Consider the given Fig. 2 which describes the instrumentation procedure for our approach for statement and branch coverage. In this procedure, tester gives a C folder which contains multiple ‘C’ files as input to the mechanism described for statement and branch coverage in framework section. At the

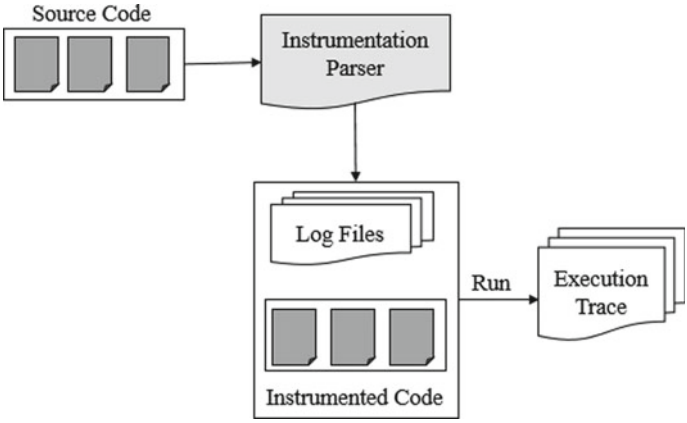


Fig. 2 Process of instrumentation

time of program compilation, a parser checks the location of function definition and condition declaration to insert an instrumented function after the next of function definition and condition declaration statements. Here the instrumentation is done at the starting of the function definition and condition declaration to get the exact portion of that particular function or condition's criteria of line of code. Here we apply a simple mechanism in which parser just check for function definition and condition declaration and if it found then add the instrumentation function to next of the line and if not then just pass for the report generation part. Instrumentation function contains the logic of gathering a line count of executed probes. Using that probe count, we get the actual portion which is executed. At the run-time, there are three other log files are generated from which one file contains instrumented line counts, second file contains function definition and condition declaration line counts and third file contains brace match counts which are used to determine executed portion of test suit. After the execution of instrumented folder, we get the coverage data which is further used for report generation phase.

Table 4 illustrates this concept with a simple C program. Here two programs such as program (a) and program (b) are given which described the actual difference of original instrumented 'C' code and instrumented 'C' code. Here two programs from which program (a) is the original 'C' code and program (b) is the instrumented code in which Ins_Probe(); is the instrumented function which is inserted at the execution time of the code. It gives the actual count number of executed line of the test suit.

4.2 Report Generation

Report generation is the graphical representation of test suit which differentiates the executed code and unexecuted code. Instrumented file with other log files is used

Table 4 Example of probe instrumentation for C code

(a) Original C code	**(b)** Instrumented C code

`If (condition)`
`begin`
`Proc1;`
`end proc 1`
`else`
`begin`
`Proc2;`
`end pr2c 1`
`end if)`
	`If (condition)`
	`begin`
	`Ins_Probe();`
	`Proc 1;`
	`end proc 1`
	`else`
	`begin`
	`Ins_Probe();`
	`Proc 2;`
	`end proc 2`
	`end if)`

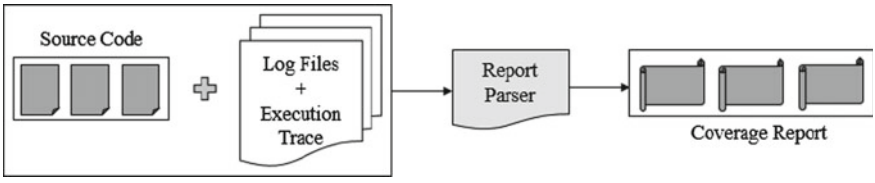


Fig. 3 Process of generating a report

to analyse the code with original source code and gives the actual output. Consider the Fig. 3 which describes the process for generating a report. In this phase, tester has the one original C code folder, coverage data and other three log files from which one contains the instrumented line counts which specify the instrumented line numbers which are further compared with executed instrumented counts, second file contains the function definition/condition declaration line counts and third file contains executed probes line counts which gives actual executed probes at the time of all program execution of input folder. This third log file is generated at the time of execution of instrumented folder.

After the instrumentation phase, execution of that particular instrumented file is by giving some user inputs. At that time, another log file is generated which contains actual line counts of executed lines. Coverage data, instrumented line count file and function definition/condition declaration line count file are parsing by build system which maps the counts of these files with original C code files. After the execution, all reports in '.html' file format in which executed portion is displayed in green colour notation and unexecuted portion is displayed in red colour notation. Unexecuted branches are differentiated using yellow colour notation. In HTML report, format percentages are also shown which specifies that how much lines are executed in form of percentage at execution time.

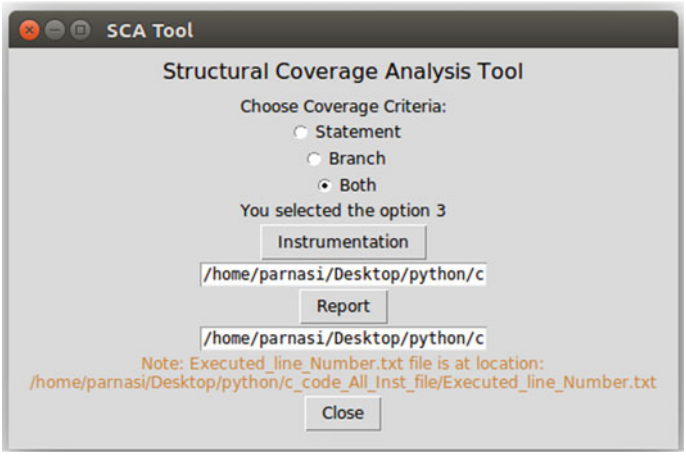


Fig. 4 Basic GUI

5 Result

Implementation and expected results of this structural coverage tool are done according to the procedure which we discussed in earlier sections. Consider the given Fig. 4 which shows the basic GUI of SCA tool.

Here we get the report in form of HTML file. Figure 5 shows the outcome of one of the generated reports. Here we get coverage for both statement and branch in the form of percentage.

6 Conclusion

The discussed procedure ensures that all the covered lines of test suit are given by statement coverage analysis and branches using branch coverage analysis and generate a coverage report. All the specified test cases described in testing section are tested properly and get appropriate outcomes for statement and branch coverage. The structural coverage analysis provides a sensible approach that balances the DO-178B requirements for structural coverage. Statement testing forces the developer to reason carefully before complete implementation of the design specification. It reveals an error in the hidden code but does not reveal missing code or statement. Statement coverage does not reveal the missing statement or code. It specifies an error in hidden code. It cannot provide the solution for branching conditions, but these can be retrieved using branch coverage. Coverage is expensive in terms of both money and time required to perform testing in avionics softwares.

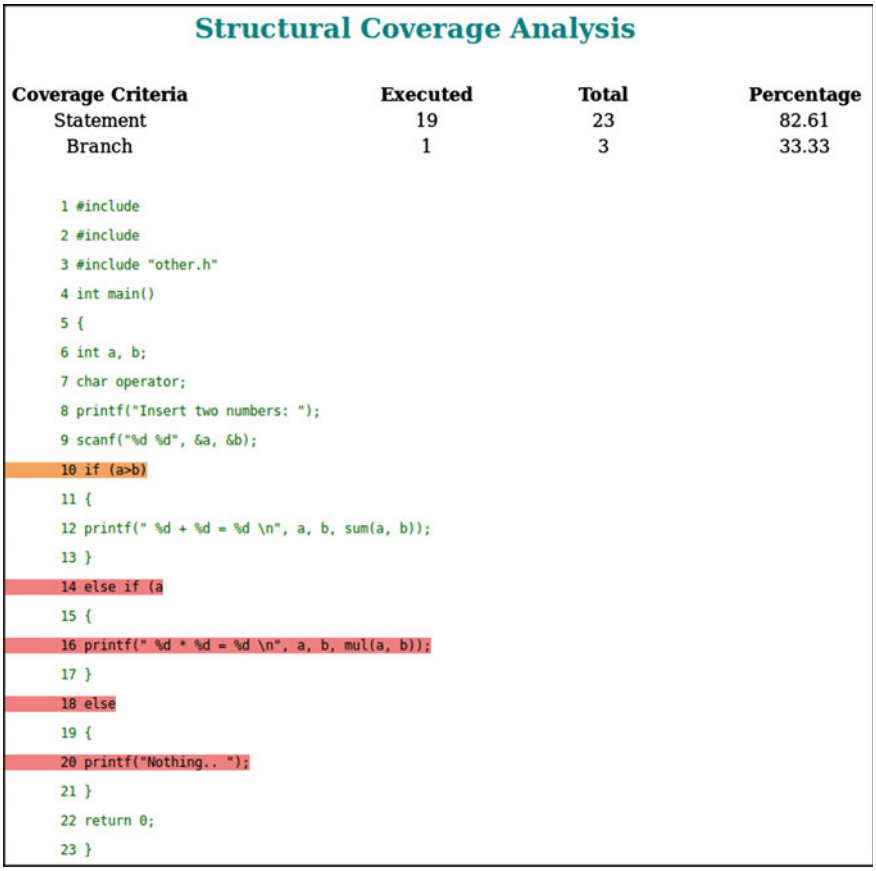


Fig. 5 HTML format of source code

7 Future Directions

Develop MC/DC coverage with their appropriate functionality and with combination of these statement and branch coverage analysis. Develop a mechanism to fulfil all the requirements for other two criteria such as decision coverage and MC/DC coverage and complete tasks which are listed below:

- Deactivated code;
- Impossible combinations of events;
- Compiler-introduced errors.

References

1. Air Traffic Organization, *Software Verification Tools Assessment Study*, Air Traffic Organization Operations Planning Office of Aviation Research and Development Washington, DC 20591, DOT/FAA/AR-06/54, June 2007
2. W. Gifford, Structural coverage analysis method, in *15th AIAA/IEEE* (1996)
3. Seven Roadblocks to 100% Structural Coverage (and how to avoid them), White Paper, Rapita System ltd
4. Software Verification Tools Assessment Study, *Air Traffic Organization Operations Planning Office of Aviation Research and Development* (Washington, DC 20591, DOT/FAA/AR-06/54, June 2007)
5. S. Park, I. Hussain, K. Taneja, B.M. Mainul Hossain, M. Grechanik, C. Fu, Q. Xie, CarFast: achieving higher statement coverage faster, 11–16 Nov 2012
6. H. Zhu, P.A.V. Hall, J.H.R. May, Software unit test coverage and adequacy. ACM Comput. Surv. **29**(4), 366–427 (1997)
7. S. Namin, S. Kakarla, The use of mutation in testing experiments and its sensitivity to external threats, in *ISSTA* (2011), pp. 342–352

Performance Analysis of Hard and Soft Thresholding Techniques for Speech Denoising Using FRFT

Prafulla Kumar and Sarita Kansal

Abstract The fractional Fourier transform is an extended version of the classical Fourier transform which is a time–frequency distribution. FRFT finds a wide range of applications in the areas of signal processing, specifically in noise removal and signal restoration. This paper gives the application of fractional Fourier transform for speech denoising using hard and soft thresholding techniques. The additive white Gaussian noise (AWGN) is considered as affecting noise to the speech signal. The performance analysis is done based on soft and hard thresholding techniques. The performance is evaluated using MATLAB.

Keywords FRFT · AWGN · Soft thresholding · Hard thresholding · MAE · PSNR

1 Introduction

A basic step in signal processing is the step of removing different noises from the received speech signals. The white Gaussian noise gets added to the speech signal; therefore, denoising the speech signal is an essential process. There are several existing approaches to denoise the speech signal. The purpose of this paper is to denoise the speech signal using a thresholding process. A good signal denoising model must remove noise completely as far as possible with preserving edges. The denoising technique mainly depends on the type of the noise and signal. The soft thresholding function can be applied for denoising the signal from its transform-domain representation, provided the transform yields a sparse representation of the signal. Soft thresholding not only smoothens the time series but also moves it toward zero.

P. Kumar (✉) · S. Kansal
Department of Electronics and Communication Engineering,
Medi-Caps Institute of Technology and Management, Indore, India
e-mail: prafulla93ethane@gmail.com

S. Kansal
e-mail: sarita_1411@yahoo.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_43

One of the researches presents the review of noise models and corresponding work in the field of denoising. Noise in the signal (speech or image) occurs mostly due to storage, transmission, and acquisition of the image. The author in the paper discusses various denoising approaches and models in detail. There exist several methods to denoise image. The important characteristic of a good image denoising model is that it should completely remove noise as far as possible as well as preserve edges [1]. The paper gives the implementation details of the discrete fractional Fourier transform. It considered the finite register length for the implementation of discrete fractional Fourier transform matrix. The algorithm is applied is hoped that this implementation and fixed-point error analysis of the discrete fractional Fourier transform which helps for further study in the signal processing community [2].

The paper is organized as follows: Sect. 2 introduces in brief about the FRFT. Section 3 explains denoising. Section 4 describes the thresholding and types of thresholding. Section 5 discusses the condition for hard and soft thresholding selection criteria and methodology used for denoising. Section 6 represents the simulation setup and result analysis. The last section concludes.

2 Fractional Fourier Transform

Fractional Fourier transform is an extended form of Fourier transform [1]. It is based on the rotation angle denoted by 'α' whose period is between 0 and 2π. The rotation angle α is calculated as $\alpha = (a\pi/2)$ where $a \in R$. If time and frequency axis (t and ω) is rotated by an angle α in counterclockwise, then the rotated variables can be represented by u and v in the form of a matrix as shown below:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \cos\alpha & \sin\alpha \\ -\sin\alpha & \cos\alpha \end{bmatrix} \begin{bmatrix} t \\ w \end{bmatrix}$$

where u and v should always orthogonal to each other.

If the signal in the time domain is a rectangular pulse, then it is realized as a sinc function in the frequency domain. Then, fractional Fourier transform converts the rectangular pulse to be in the domain between time and frequency shown in Fig. 1.

3 Denoising

Scientists and researchers in several fields of planetary science and molecular spectroscopy are a concern to recover an original and authentic signal from noisy, indirect, or incomplete data received. But fractional Fourier transform has emerged as the solution to recover original signal from the noisy data, and the technique is known as denoising technique. The denoising techniques in Fourier transform are known as shrinkage or thresholding techniques.

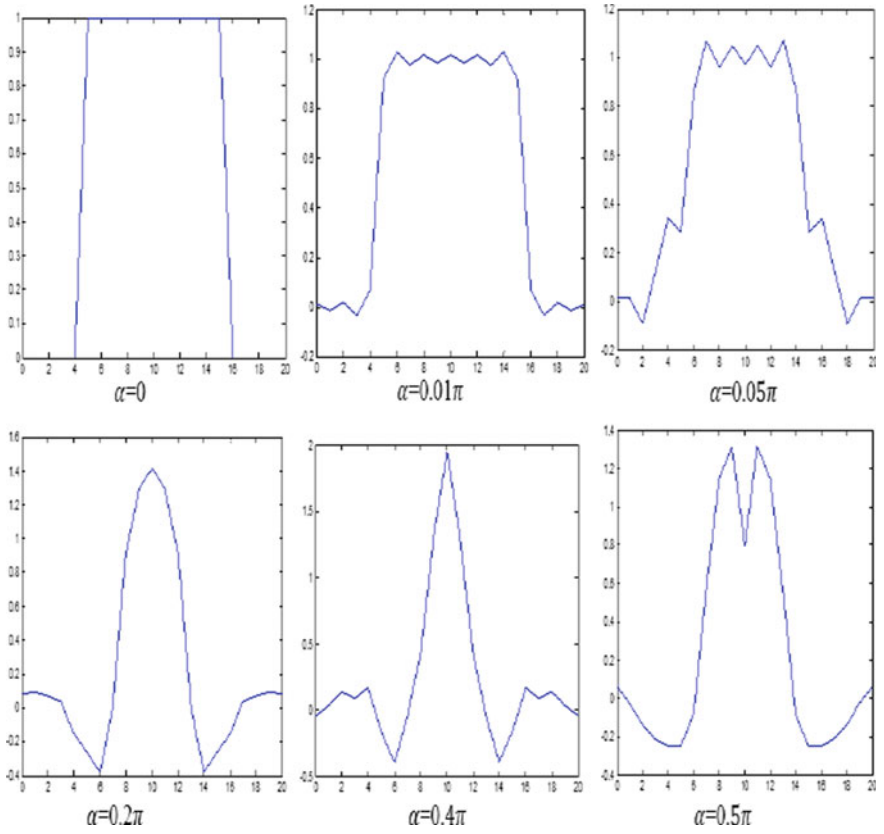


Fig. 1 Waveforms of the FRFT

4 Thresholding

Generally, the coefficients of fractional Fourier transform of any signal are either equal or close to zero. Thus, thresholding is used to modify the coefficients such that the coefficients sequence may contain long strings of zeros. Therefore, different threshold schemes can be employed.

There are two popular versions of thresholding for denoising the signal:

Hard Thresholding: In hard threshold scheme, a fixed tolerance is selected. The wavelet coefficient whose value lies below the tolerance is set to zero aiming to append as many zeroes as possible. Threshold value should be properly selected as larger threshold may result in errors. In other words, hard thresholding is based on keep and kill rule and is more instinctively appealing by introducing artifacts in the recovered signals. This sets any coefficient less than or equal to the threshold to zero.

Soft Thresholding: In soft threshold, the coefficients are compared with the selected tolerance. For the coefficients having value less than threshold selected are set to zero

while the remaining coefficients are replaced with the average and difference of the coefficients in the matrix. As hard thresholding sets any coefficient which is less than or equal to the threshold to zero, then the threshold is subtracted from any coefficient which is greater than the threshold. This takes the time series toward zero. Soft thresholding works on shrink and kill rule, as it shrinks the coefficients above the threshold in absolute value.

For a speech signal, soft thresholding rule can be applied by using σ_n and σ. Here, σ_n is the standard deviation of the noise and σ is the standard deviation of the noise-free transform coefficients. For the known value of σ_n, unknown σ has to be calculated. The experiments on the transform coefficients show that σ varies scale to scale. Therefore, σ is different for each subband.

5 Threshold Selection Criteria

Finding an optimal threshold value is not that much easy task. For speech signals denoising applications, the selected threshold value should maximize peak signal-to-noise ratio (PSNR). A lower value of threshold will allow all the noisy coefficients to pass, and therefore the resultant speech signal will be noisy, while larger threshold would change more coefficients to zero. As a result, resultant signal may lose some signal information. Thus, an optimal value should be selected as threshold value.

5.1 Condition for Thresholding

```
For any signal i, the Fourier coefficient is coef[i].
Then condition for hard thresholding is:
  if (coef[i] <= thresh)
    coef[i] = 0.0;
  Similarly, condition for soft thresholding:
  if (coef[i] <= thresh)
    coef[i] = 0.0;
  else
   coef[i] = coef[i] - thresh;
```

The block diagram of the denoising process is shown in Fig. 2. The input speech signal is mixed with Gaussian noise. The soft and hard thresholding is applied on the spectrum received after FRFT. The inverse FRFT is applied after the shrinkage of coefficients.

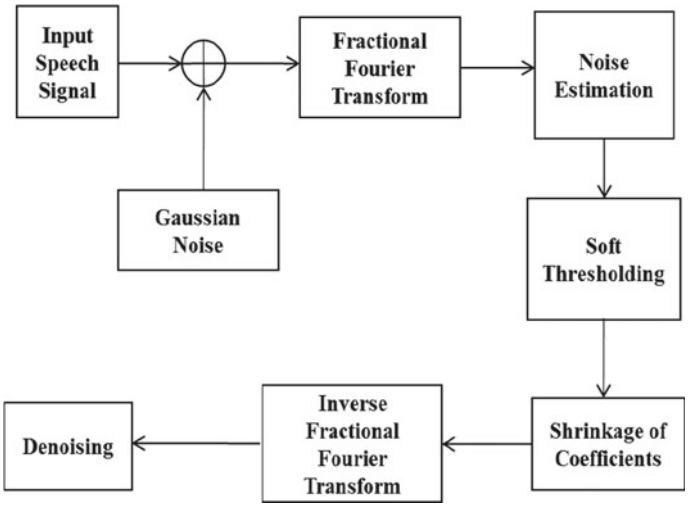


Fig. 2 Block diagram of denoising process

6 Simulation and Results

The denoising of the speech signal has been performed using the hard thresholding/soft thresholding on FRFT signal for different signal-to-noise ratios. The parameters used for the simulation are given in Table 1.

Figure 3 shows the soft thresholding technique is applied to FRFT tool for denoising the speech signal with different values of SNR. The performance is shown in Fig. 3. The increase in SNR decreases the MAE value with different thresholds.

The hard thresholding is applied to FRFT tool for denoising the speech signal with different values of SNR. The performance is shown in Fig. 4. The increase in SNR decreases the MAE value with different thresholds.

The hard thresholding gives better performance than soft thresholding at high threshold value. At high threshold value, hard thresholding gives low MAE than soft thresholding with increasing SNR value by comparing Figs. 3 and 4.

Table 1 Simulation parameters

S. no	Parameters	Values
1	Sampling rate	44100 Hz
2	Speech duration	18.46 s
3	Fraction exponent	0.1–0.9
4	Threshold range	0.005 to 20% of max signal
5	SNR range	0–30 dB

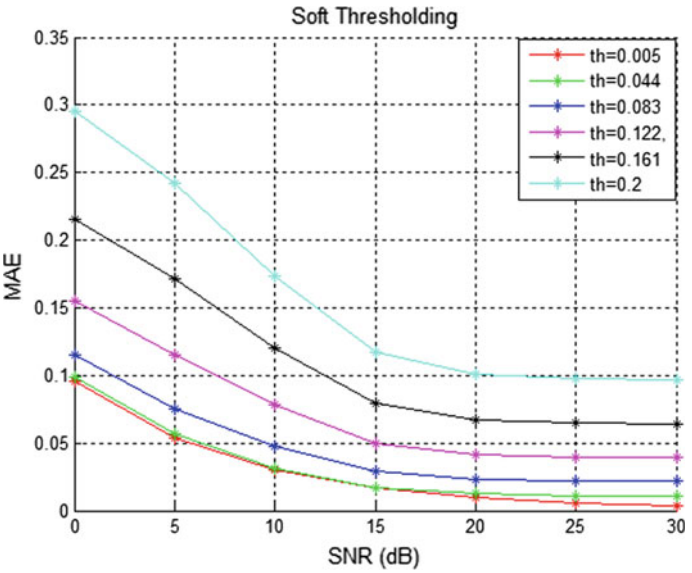


Fig. 3 Performance of variation of MAE with SNR

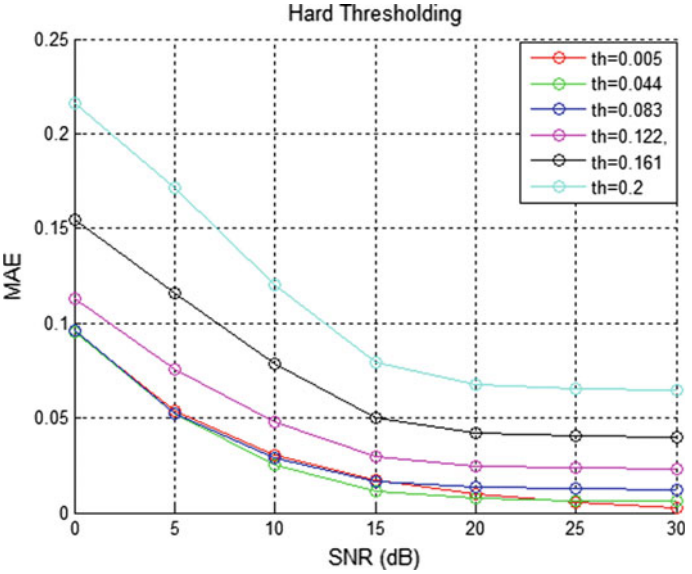


Fig. 4 The performance of variation of MAE with SNR

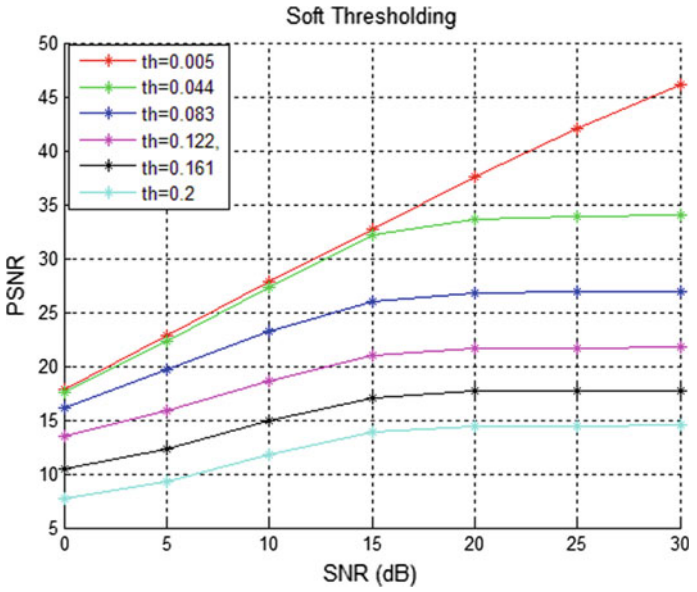


Fig. 5 Performance of PSNR with SNR

The soft thresholding is applied to FRFT tool for denoising the speech signal with different values of SNR, and its performance is shown in Fig. 5. The increase in SNR increases the value of PSNR at different threshold values.

The hard thresholding is applied to FRFT tool for denoising the speech signal with different values of SNR to obtain PSNR, and its performance is shown in Fig. 6.

The hard thresholding obtains better performance than soft thresholding. The hard thresholding gives good PSNR than soft thresholding by comprising Figs. 5 and 6.

The soft thresholding technique is applied to FRFT tool at different fractional powers to denoise the speech signal. The soft thresholding gives low MAE with increase in the value of SNR shown in Fig. 7.

The hard thresholding technique is applied to FRFT tool at different fractional powers, and its performance is shown in Fig. 8.

The hard thresholding technique gives better performance than soft thresholding because of low MAE in hard thresholding than soft thresholding at different fractional powers.

The performance of soft thresholding is shown in Fig. 9. The soft thresholding is applied to FRFT tool to denoise the speech signal with different values of SNR at different fractional powers.

The hard thresholding is applied to FRFT tool to denoise the speech signal at different fraction powers, and its performance is shown in Fig. 10.

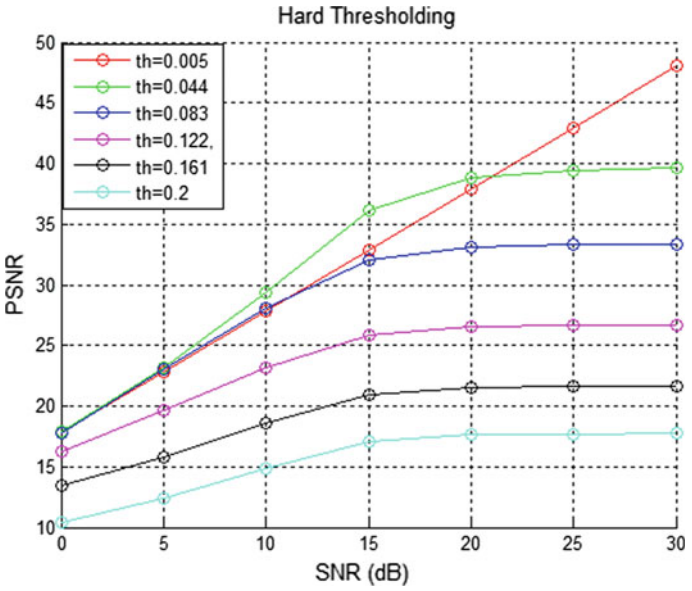


Fig. 6 The performance of PSNR with SNR

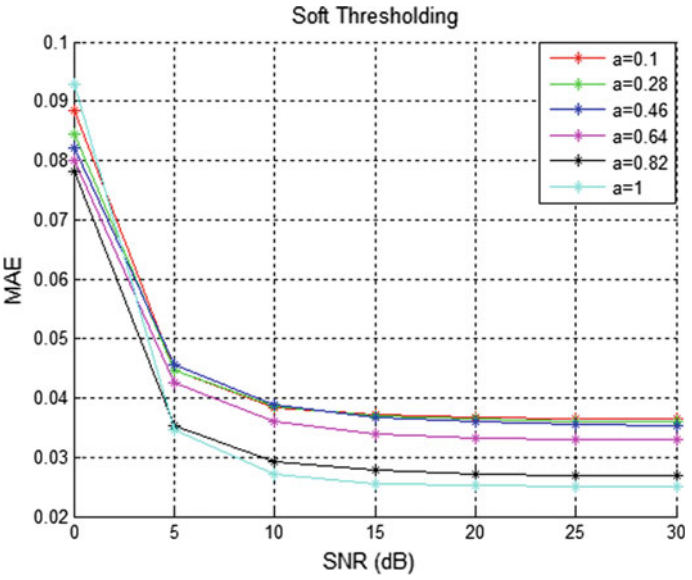


Fig. 7 The performance of variation in MAE with SNR

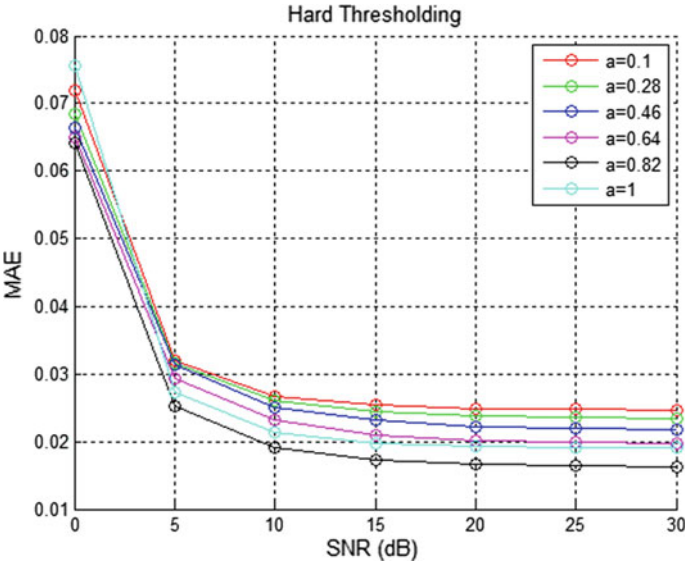


Fig. 8 The performance of MAE with SNR

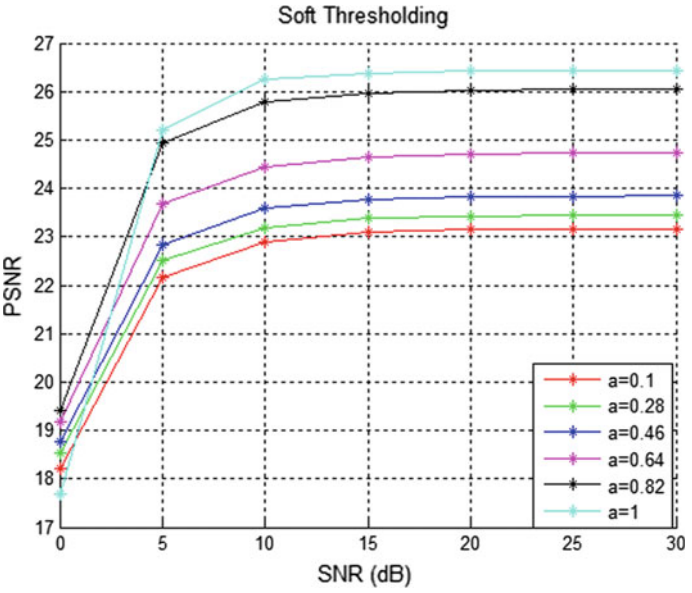


Fig. 9 The performance of PSNR with SNR

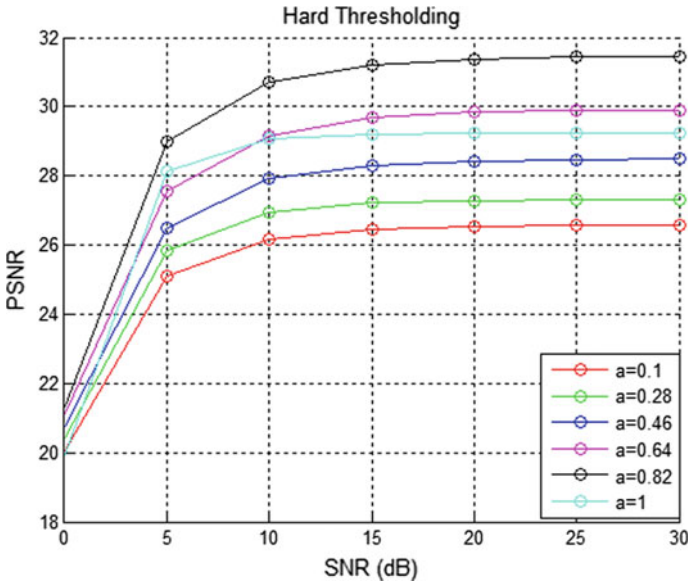


Fig. 10 The performance of variation of PSNR with SNR

7 Conclusion

The speech signal denoising is performed using FRFT and the thresholding (soft/hard) to remove the noise part from the spectrum. The FRFT spectrums of clean speech signal, the upper spectral band has almost no magnitude. This is a motivating thought to perform the thresholding on this spectrum band to remove the noise level in noisy speech signal. The simulation result is for each case at a different SNR, threshold value, and different fractional coefficients. The hard thresholding is better than soft thresholding because it gives good PSNR and low MAE.

References

1. P.B. Alisha, K. Gnana Sheela, Image denoising techniques-an overview. IOSR J. Electr. Commun. Eng. (2016)
2. V. Ashok Narayanana, K.M.M. Prabhu, The fractional Fourier transform: theory, implementation and error analysis (2003)
3. A. Rutwik, B.T. Krishna, Speech enhancement using fractional order spectral subtraction method. IJCTA (2015)
4. J. Wang, Speech enhancement based on fractional Fourier transform. WSEAS Trans. Signal Process. **10** (2014)
5. M.F. Edren, M.A. Kutay, H.M. Ozaktas, Repeated filtering in consecutive fractional Fourier domains and its applications to signal restoration. IEEE Trans. Signal Process. (1999)

6. V. Namias, The fractional order Fourier transform and its application to quantum mechanics. J. Inst. Math. Appl. **25**, 241–265 (1980)
7. E.U. Condon, Immersion of the Fourier transform in a continuous group of functional transformations. Proc. Natl. Acad. Sci. (1937)
8. R. Patel, Review paper on fractional Fourier transform. Int. J. Adv. Res. Sci. Eng. **5** (2016)
9. P.-Y. Lin, The fractional Fourier transform and its applications. IJCTA (2010)
10. H.M. Ozaktas, O. Arikan, Digital computation of the fractional Fourier transform. IEEE Trans. Signal Process. **44**(9) (1996)

A New Algorithm to Implement Priority Queue with Index Sequential Chaining

Fakhruddin Amjherawala and Ummulbanin Amjherawala

Abstract Priority queue is an abstract data type which is used to insert or remove an element according to priority. It works as assigning a priority to the process or thread to execute accordingly. There are two ways to implement priority queue dynamically: using linked list and using heap. Inserting an element in a priority queue using linked list requires visiting the list from start to end to determine the proper location to set an element, as maintain the list according to precedence. On the other hand inserting and deleting key element using heap, requires preserving the framework and its arrangement, the result must be a heap. In this paper, we eliminate the task of breaking the link to place an element according to its priority, as well as do not maintain the structural and ordering properties by exploring the indexing mechanism as priority to implement priority queue efficiently.

Keywords Heap · Index · Linked list · Priority

1 Introduction

In computer science, a priority queue [1] which is like a regular queue data structure, in which each element has a "priority" associated with it. For example, an element with high priority is served before an element having low priority. In multithreading, each thread is assigning a priority so that it executes as accordingly. So there are concept of linked list and heap to implement priority queue. In this paper, we show the index sequential chaining to implement it.

F. Amjherawala (✉) · U. Amjherawala
School of Computers, IPS Academy, Indore, Madhya Pradesh, India
e-mail: ips_fakhruddin@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_44

2 Problem

The concepts of priority queue show that how element in queue arrange so that it can be added and access according to the priority. One way to define a priority queue dynamically using linked list [2], while putting an element in such a priority queue requires visiting the list from start to end to determine the proper location to set an element according to its precedence, in order to keep the list sorted. Another way is by using the concept of heap [3] (a binary tree), while inserting and deleting an element from a heap must be careful to preserve the framework and its arrangement. The result must be a heap. "Move up" the element towards the root, replacing with element of ancestor, until it is in a node whose ancestor has a maximum or minimum element (or it is in the root). While deleting the element from a heap, "Move down" the element of the root towards the leaves, replacing with element in the maximum or minimum element child, until it is in an intermediate node whose children have minimum or maximum element, or it is in a last level node.

3 Background Knowledge

3.1 Defining a Priority Queue Dynamically Using a Linked List

One way to define a queue using linked list requires:

- Pointers must be at the start of the list and the end of the list.
- Put an element at the end; remove an element from the starting position.

Maintain the list either in ascending or descending order, with the "Maximum priority" element first. Putting an element in such a priority queue requires visiting the list from start to end to determine the proper location to set an element, as maintain the list according to precedence. While removing the highest priority element in such a priority queue just requires breaking the starting element link from the list.

3.2 Defining Heap as a Priority Queue Using Binary Tree

Creating heap using binary tree with the following characteristics:

- "Framework": It follows the "Almost complete" binary tree characteristics, which is filled from left-to-right. Heaps are "complete" binary trees and balanced: of height H with complexity O (log N).

- "Arrangement": Element in the root must be greater than or equal to its child element. This is sometimes called a max-heap, for min-heaps, change greater than with less than.

So heap follows the framework for defining queue data structure with arrangement according to its precedence are known to be "Priority Queue".

3.3 Algorithm to Insert an Element into a Binary Tree Using Heap

While putting a key element into a binary tree using heap, must be maintaining its framework and arrangement. The result must be a heap! As shown in Fig. 1.

1. Arranging level one by one from left-to-right by creating a new node.
2. Element to be inserted in this new node.
3. "Move up" the element towards the root, replacing element with its parent element, until it is in a node whose predecessor has a maximum element or it is the root.

Insert in a heap with N nodes has time cost O (log N).

3.4 Algorithm to Remove the Maximum Element from Binary Tree Using Heap

While removing the maximum element from a binary tree using heap must be maintaining its framework and arrangement. The result must be a heap!! As shown in Fig. 2.

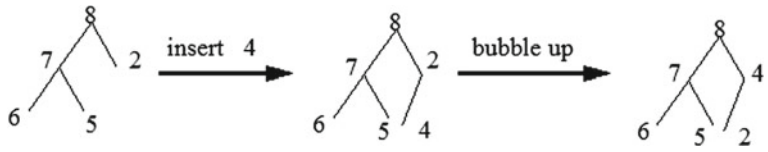


Fig. 1 Insert a key into a heap

1. To return, initially retrieve the maximum element from the root for swapping.
2. Determine the node to be removed, arrange leaf level from right-to-left.
3. Move the element to the node which is to be removed from the root and remove that node.
4. "Move down" the element from the root towards the leaves, replacing with maximum element of child, until it is in a node whose successor have smaller elements, or it is in a leaf.

In a binary heap, insertion and deletion can be implemented in O (Log N) time.

4 Research Objective and Approach

To implement priority queue abstract data type in such a way that priority set as an index so that it maintains the order as well as sequentially insert and delete list of elements. Used of this approach removes the constraint of linked list and heap.

In this approach, set size of index (list) as per minimum and maximum priority. For each priority, index insert list of elements without shifting any element with their respective position. There is a rear pointer which shows the end of each of the priority queues. There is a front pointer which deletes an element from the queues sequentially and according to the priority. Duplicate priority elements are also placed at the same index position sequentially.

5 Design of the Algorithm

In this approach, we use the concept of index sequential chaining to insert and delete the elements according to the priority. Initially, create an indexing to arrange the elements according to the priority. According to queue operation, each index contain queue as a linked list of same priority element. Each queue contain rear pointer which shows the last element of the particular index queue. It contains a front pointer which deletes element sequentially.

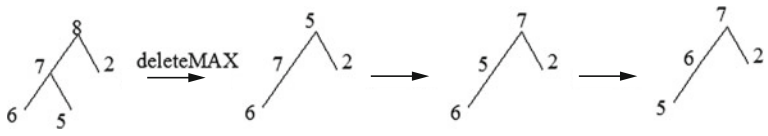


Fig. 2 Delete a key from heap

```
Create Index:
structure  index
begin
 int info, priority;
structure index *next;
 end Index;
Initially set Count -> 0 and max_ priority -> 0

Insertion:
procedure Insert (Index in, Index Front, Index Rear, pri-
ority)
begin
        Allocate a memory to the new node
         node ->info=element;        // Set element
         node ->priority=priority;   // Set the priority
         node ->next=NULL;
               if in[priority] = NULL
                   in[p]=Rear[priority]=node;
               else
                   Rear[priority]->next=node;
                   Rear[priority]=node;
               end if;

       Count=Count+1;
       // count shows how many element in priority queue
  If priority > max _priority
      max _priority = priority;
      Front=Rear[priority];
  end if;
 end procedure;

Deletion:
procedure Delete (Index in, Index Front, Index Rear)
begin
Index temp;
//Create a temporary node for delete and free the memory
//space

if count = 0
print "empty"  return;
  end if;
  temp = front;
  Front = Front-> next;
```

Index	Pointer	Rear position	
0	NULL	Rear [0]	NULL
1	NULL	Rear [1]	NULL
2	NULL	Rear [2]	NULL
3	NULL	Rear [3]	NULL
4	NULL	Rear [4]	NULL
5	NULL	Rear [5]	NULL

Fig. 3 Creation of temporary node for deleting and freeing the memory

```
   in [max_priority]= Front;
   free (temp);
   temp = NULL;
// Check if front is empty so move it the next priority
index queue.
   if  Front = NULL
               Rear [max _priority]=NULL;
                max _priority = max _priority -1;
// Move to next index position till it is not empty then
//set the front.
            while in [max _priority] = NULL do
            max_priority =  max_priority -1
                        Front= in [max_priority];
   end while;
        end if;
    Count = Count-1;
end procedure;
```

A. The Insertion Operation:

Creating a priority list indexing with maximum size 6 having min_priority set with 0 and max_priority as maximum size given by user.

Initially, each index is with empty as given in Fig. 3.

Here, max_priority index is 5.

Set Front → NULL.

Insert 25 with priority 3, 43 with priority 2, 34 with priority 0 as given in Fig. 4.

If same priority element occurs then insert element without shifting as given in Fig. 5.

Figure 6 shows insertion of 61 with priority 2 and Insert 50 with priority 3.

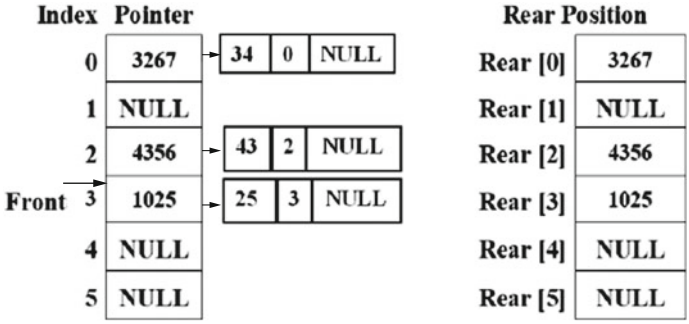


Fig. 4 Insertion of 25 with priority 3, 43 with priority 2, 34 with priority 0

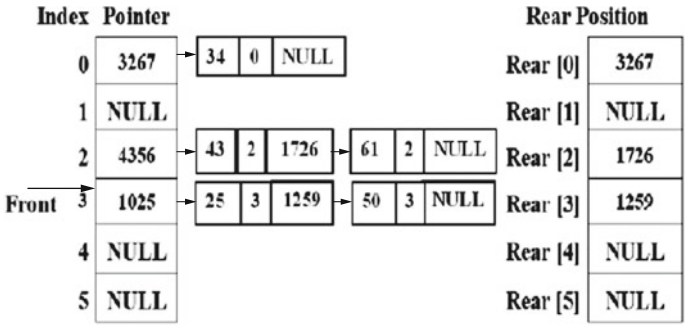


Fig. 5 Same priority elements then insertion without shifting

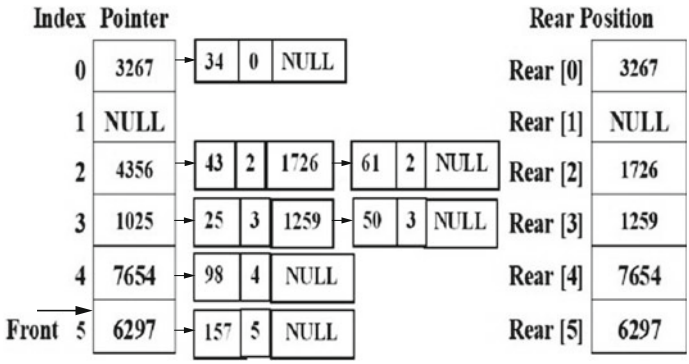


Fig. 6 Insertion of 61 with priority 2 and Insert 50 with priority 3

If higher priority element is inserted, then shifting of element does not occur in as shown in Fig. 6.

Insert 157 with priority 5 and 98 with priority 4.

B. The Deletion Operation:
Delete an element according to the front pointer one by one without any shifting.

6 Implementation Result

We implemented the priority queue with index sequential chaining using 64-bit operating system and x64-based processor along with 4 GB DDR3 RAM. The execution time need for an insertion and deletion operation is O (1) time.

An advantage of this algorithm is to save time on swapping as well as on splitting the link while insertion and deletion of node occur. Length of same priority node is independent of the actual length of queue.

7 Conclusion

In this paper, we presented scalability according to time and space as well as removing task of exchanging of element according to the precedence of element in a queue by using index sequencing. This algorithm minimizes the operation of heap and linked list by exploring the index mechanism as priority to implement priority queue efficiently.

References

1. M. Tenenbaum, Y. Langsam, M.J. Augenstein, *Data Structure Using C* (Pearson Education, India, 1990)
2. J.P. Tremblay, P.G. Sorenson, *An Introduction to Data Structures with Applications* (Tata McGraw-Hill, New York, 1976)
3. S. Chattopadhyay, D.G. Dastidar, M. Chattopadhyay, *Data Structure Through C Language* (BPB, India, 2001)

An Efficient Parallel Implementation of CPU Scheduling Algorithms Using Data Parallel Algorithms

Suvigya Agrawal, Aishwarya Yadav, Disha Parwani and Veena Mayya

Abstract Modern graphics processors provide high processing power, and furthermore, frameworks like CUDA increase their usability as high-performance co-processors for general-purpose computing. The Graphical Processing Units (GPUs) can be easily programmed using CUDA. This paper presents an efficient parallel implementation of CPU scheduling algorithms on modern The Graphical Processing Units (GPUs). The proposed method achieves high speed by efficiently exploiting the data parallelism computing of the The Graphical Processing Units (GPUs).

1 Introduction

The modern GPU is a many-core processor which supports execution of thousands of threads concurrently. GPU comprises of a series of streaming processors with hundreds of core aligned in a particular way which facilitate single instruction multiple threads (SIMTs) programming model.

General-purpose computing on GPU is a graphical processing unit which is very efficient at computer graphics manipulation and image processing. The highly parallel structure of GPU makes it easy to use to perform the general-purpose computation and accelerate traditional CPU-based computational tasks. Recently, general-purpose

S. Agrawal (✉) · A. Yadav · D. Parwani · V. Mayya
Department of Information and Communication Technology, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal 576104, India
e-mail: suvigya1st@gmail.com

A. Yadav
e-mail: aishwaryayadav91@yahoo.com

D. Parwani
e-mail: disha.parwani9927@gmail.com

V. Mayya
e-mail: veena.mayya@manipal.edu

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_45

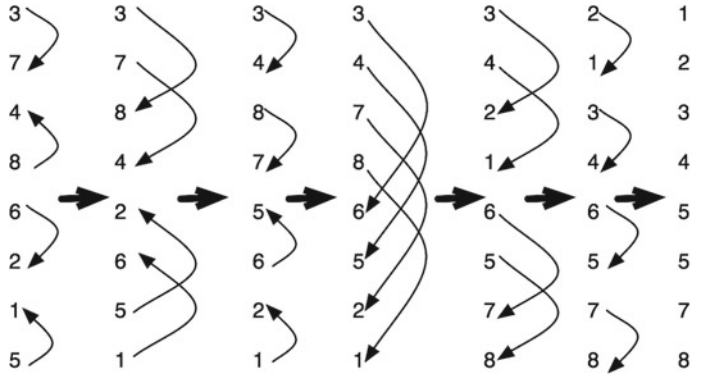


Fig. 1 Example of bitonic parallel sort algorithm

computation on graphics processing units (GPGPU) has been adopted in accelerating many algorithms such as sorting [11], graph algorithms [12], data encryption algorithms [5], and other generic algorithms [13]. Several libraries and programming environment are available that allow programmers to perform GPGPU and exploit computational power of GPU. Compute Unified Device Architecture (CUDA) [8] framework is one of the programming environments provided by NVIDIA that allows to exploit data parallelism of GPU using C-style programming language.

Scheduling is a way by which work specified by some means is assigned to resources that complete the work. Job/process scheduling is the process of arranging, controlling, and optimizing the allocation of system resources to threads, processes, and data flows for maximum utilization. The operating system schedules the processes for execution using several scheduling algorithms based on various scheduling criteria such as CPU utilization, throughput, turnaround time, waiting time, and priority. System resources can be equally and effectively utilized by properly scheduling the processes and hence achieve a target quality of service. The major scheduling algorithms types of CPU/job scheduling algorithms include as follows: First Come First Serve (FCFS), Shortest Job First (SJF), Round Robin (RR), and Priority-Based Scheduling (PBS). Scheduling is required to perform multitasking and multiplexing. Scheduling is a complex job requiring extensive processing which is better performed on a parallel platform.

In this paper, a novel parallel approach to perform scheduling using CUDA technology to enhance the performance of process scheduling algorithms is proposed. A combination of well-established data parallel algorithms and parallel sorting techniques has been adopted to achieve drastic performance increase in execution time of scheduling algorithms (Fig. 1).

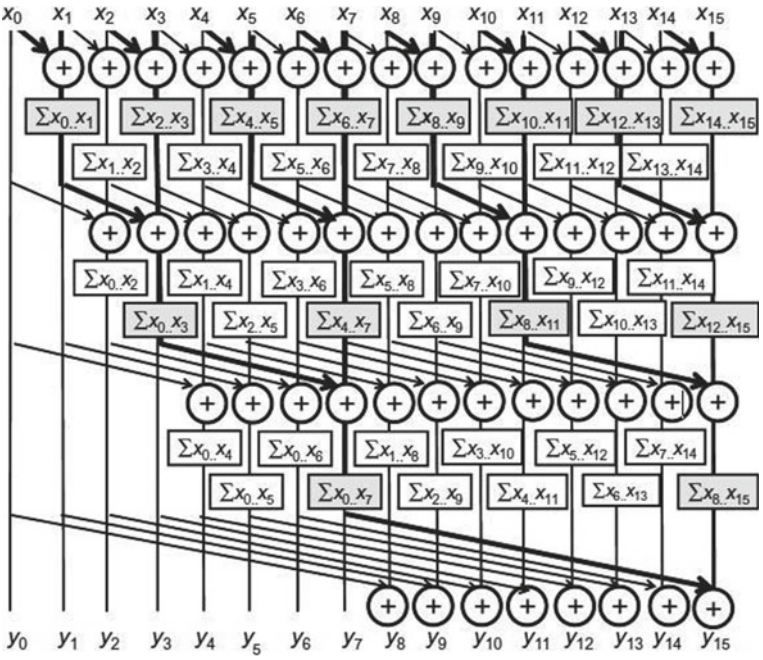


Fig. 2 Prefix sum data parallel algorithm

1.1 Data Parallel Algorithms

In data parallel algorithms, parallelism is involved in simultaneous operations across large sets of data, rather than from multiple threads of control [4]. Prefix sum or scan is one among the most common data parallelism algorithms. The prefix sum, cumulative sum, inclusive scan, or simply scan of a sequence of numbers $x_0, x_1, x_2, \ldots x_{n-1}$ are the second sequence of numbers $y_0, y_1, y_2, \ldots y_{n-1}$, the sums of prefixes (running totals) of the input sequence where $y_0 = x_0; y_1 = x_0 \oplus x_1; y_1 = x_0 \oplus x_1 \oplus x_2 \ldots$, as shown in Fig. 2. Mathematically $y_i = x_0 \oplus x_1 \oplus x_2 \oplus \cdots \oplus x_i \; \forall_i \in 0 \ldots n-1$. Figure 2 depicts the pictorial representation for the scan data parallel algorithm [6].

1.2 Bitonic Sort

There are multiple sorting algorithms available such as Merge sort, Quick sort, Radix sort, and Heap sort. Bitonic sort is one among the sorting algorithms that can make use of GPU computing power and thus is efficient in terms of both space and time complexities [7].

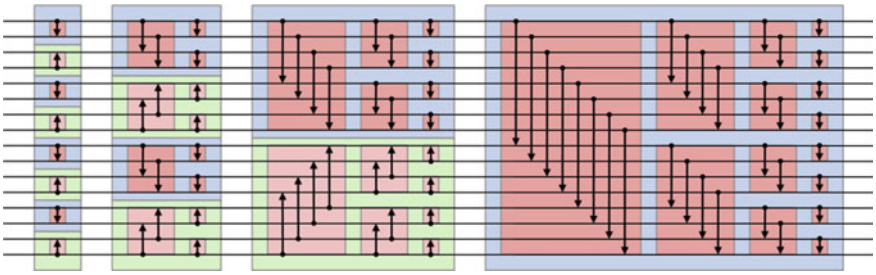


Fig. 3 Representation of bitonic sort

Bitonic Sequence: A sequence is called Bitonic if it is first increasing, then decreasing. In other words, an array arr[0..n − 1] is Bitonic if there exists an index i where $0 \leq i \leq n-1$ such that,

$$a_0 \leq a_1 \leq \cdots \leq a_{n/2-1} \, and \, a_{n/2} \geq a_{n/2+1} \geq \cdots \geq a_{n-1}$$

The Bitonic Sort Algorithm (as illustrated in Fig. 1):

(1) Let s = $\langle a_0, a_1, a_0, \ldots, a_{n-1} \rangle$ be a bitonic sequence such that

   (a) $a_0 \leq a_1 \leq \ldots \leq a_{n/2-1}$, and
   (b) $a_{n/2} \geq a_{n/2+1} \geq \ldots \geq a_{n-1}$

(2) Consider the following subsequences of s

   (a) $s_1 = \langle min(a_0, a_{n/2}), min(a_1, a_{n/2+1}), \ldots, min(a_{n/2-1}, a_{n-1}) \rangle$
   (b) $s_2 = \langle max(a_0, a_{n/2}), max(a_1, a_{n/2+1}), \ldots, min(a_{n/2-1}, a_{n-1}) \rangle$

(3) Sequence properties

   (a) $s_1 \, and \, s_2 \, are \, both \, bitonic$
   (b) $\forall x \forall y \, x \in s_1, y \in s_2, x < y$

(4) Apply recursively on $s_1 \, and \, s_2$ to produce a sorted sequence
(5) Works for any bitonic sequence, even if $|s_1| \neq |s_2|$.

Given an unordered sequence of size $2n$, exactly $log_2 2n$ stages of merging are required to produce a completely ordered list (Fig. 3).

1.3 Prefix Sum (Scan)

Prefix sum of a sequence of numbers $x_0, x_1, x_2, \ldots, x_n$ is another sequence of numbers $y_0, y_1, y_2, \ldots, y_n$ given by:

$$y_0 = x_0; y_1 = x_0 \oplus x_1; y_1 = x_0 \oplus x_1 \oplus x_2$$

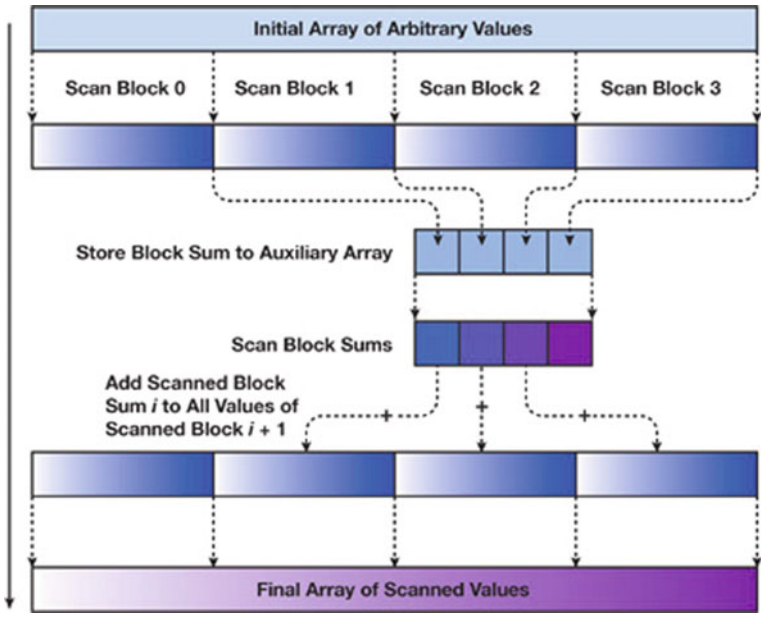


Fig. 4 Block-wise data parallel scan operation

The general mathematical representation of inclusive prefix sum for computing the output value in a sequential order is:

$$y_i = x_i; \quad i = 0$$
$$y_i = y_{i-1} \oplus x_i; \quad i > 0$$

The given data set is divided into blocks of fixed size. The scan is performed individually on each block. The result obtained is not the final result. It is a temporary array, as second block does not counter for first block elements and similarly henceforth. Each block sum is the last element of that block. The concept used first involves extracting the last element of the block that will give the individual block sum and storing these in a separate array let's say Y in the corresponding positions, i.e., block 0 sum stored at zeroth index and block 1 sum stored in first index. The final result is obtained by adding the elements in array Y to the corresponding temporary array elements by using indices of block and the thread. For instance, the block 0 contains final result so we do not have to perform any computation. But for block 1, Y[0] (which is nothing but the block sum of block 0) is to be added to each and every element of block 1 present in the temporary array. For block 2, Y[0] + Y[1] is added to every element, which is nothing but the block sum of zeroth and first block, respectively, as shown in Fig. 4, similarly for further blocks. Finally, the resultant array is obtained.

2 Related Work

Job/process scheduling is one of the important tasks performed by the operating system (OS). The performance of the OS depends on the CPU scheduling algorithms. Recently, several improved CPU scheduling algorithms such as [2, 9, 10] have been introduced for improving the system performance. Scheduling needs to be carried out very frequently by the operating system and is a complex job which may require repetitive computation. State-of-the-art performance is achieved by implementing many generic algorithms [1, 3] on GPU. This motivates to implement scheduling algorithms on GPU and analyze the performance.

3 Implementation

Algorithm 1 provides the steps carried out to implement the parallel non-preemptive scheduling algorithm.

Algorithm 1 Algorithm for parallel scheduling using data parallel algorithm

```
1: procedure MainModule
2:     Allocate memory for the input using cudaMallocManaged
3:     Read and store the input priority, burst time, arrival
       time in the above allocated variables
4:     Launch the SORT kernel as depicted in Algorithm 2.
5:     Perform block  SCAN kernel  as depicted in Algorithm 4
       with required parameters.
6:     Launch the Reduction kernel with required parameters
       to find the average waiting time and turnaround time
```

Algorithm 2 Algorithm for Bitonic Sort kernel launch

```
1: procedure SortingFirst (int *pr, int *bt)
2:     dim3 blocks(BLOCKS,1);
3:     dim3 threads(THREADS,1);
4:     for (int k = 2; k <= NUM; k <= 1) do
5:         for (int j = k >>1; j >0; j = j >>1) do
6:             BitonicSortStep <<<blocks,threads>>>(pr,j,k);
7:     //The kernel is shown in Algorithm 3.
```

Algorithm 3 Main Algorithm of Bitonic Sort

```
1: procedure BitonicSortStep(float *values, int j, int k)
2:     unsigned int i, ixj;
3:     i = threadIdx.x + blockDim.x * blockIdx.x;
4:     ixj = i ^ j;
5:     if ((ixj)>i) then
6:        if ((i & k )==0) then
7:           if (values[i] > values[ixj]) then
8:              float temp = values[i];
9:              values[i] = values[ixj];
10:             values[ixj] = temp;
11:       if ((i & k)!=0) then
12:          if (values[i] < values[ixj]) then
13:             float temp = values[i];
14:             values[i] = values[ixj];
15:             values[ixj] = temp;
```

To form a bitonic sequence from a random input, we start by forming four-element bitonic sequences from consecutive two-element sequence. Consider four-element in sequence x_0, x_1, x_2, x_3. We sort x_0 and x_1 in ascending order and x_2 and x_3 in descending order. We then concatenate the two pairs to form a four-element bitonic sequence. Next, we take two four-element bitonic sequences, sorting one in ascending order, the other in descending order (using the bitonic sort which we will discuss below), and so on, until we obtain the bitonic sequence as shown in Fig. 3.

Algorithm 4 Algorithm for Scan kernel

```
1: procedure BlockSum(x)
2:     n = length(x)
3:     y = fill(x[1],n)
4:     for i = 2 : n do
5:        xy[i] = xy[i-1] + x[i]
6:     Wait for all threads in a block to finish
7:     Copy xy into y
8:     Wait for all threads of all block to finish
9:     if (threadIdx.x <blockIdx.x) then
10:       Extract last element of every block and put in xy.
11:    Wait for all the threads to finish.
12:    for i = 0 : blockIdx.x do
13:       Add the block sums to all the elements of that
          block from xy.
```

The experiment done was on analyzing the performance of the Priority-Based Scheduling (PBS) algorithm. A large number of processes are taken which include parameters such as burst time, arrival time, and priority. The processes are sorted based on the priority first, and then, the tie is broken between the processes having same priority depending upon their burst time. The scheduling is performed placing the processes with the highest priority and lowest burst time first (the highest priority corresponds to lowest number). The input elements are sorted using bitonic sort. Further, the block-wise work-efficient scan operation is performed to obtain the waiting time and turnaround time as shown in Fig. 4. Data parallel reduction algorithm is applied to sum waiting time and turnaround time and thus compute average waiting time and turnaround time. The scan algorithm iterates $log(n)$ time.

4 Results

The proposed method uses parallel bitonic sort, and the computation of this sorting has a complexity of $O((\log N)^2)$ that makes it n times faster than its serial complexity $O(N\ (\log N)^2)$. Param Shavak with Kepler GTX GPU card is used to analyze the proposed method. Screenshots of execution are shown in Figs. 5 and 6 that depict the time taken to execute serial and parallel scheduling code, respectively. Figure 7 shows the graphical representation for the same. It can be seen that execution speed of parallel code is almost 10–15 times more than that of serial code.

```
Serial Computation Elapsed time: 0.000263s for 256 process
Serial Computation Elapsed time: 0.000921s for 512 process
Serial Computation Elapsed time: 0.003582s for 1024 process
Serial Computation Elapsed time: 0.013594s for 2048 process
Serial Computation Elapsed time: 0.044259s for 4096 process
Serial Computation Elapsed time: 0.121259s for 8192 process
```

Fig. 5 Screenshot of serial execution of the PBS

```
Parallel Computation Elapsed time: 0.000489s for 256 process
Parallel Computation Elapsed time: 0.000557s for 512 process
Parallel Computation Elapsed time: 0.000615s for 1024 process
Parallel Computation Elapsed time: 0.000730s for 2048 process
Parallel Computation Elapsed time: 0.000890s for 4096 process
Parallel Computation Elapsed time: 0.001073s for 8192 process
```

Fig. 6 Screenshot of parallel execution of the proposed method

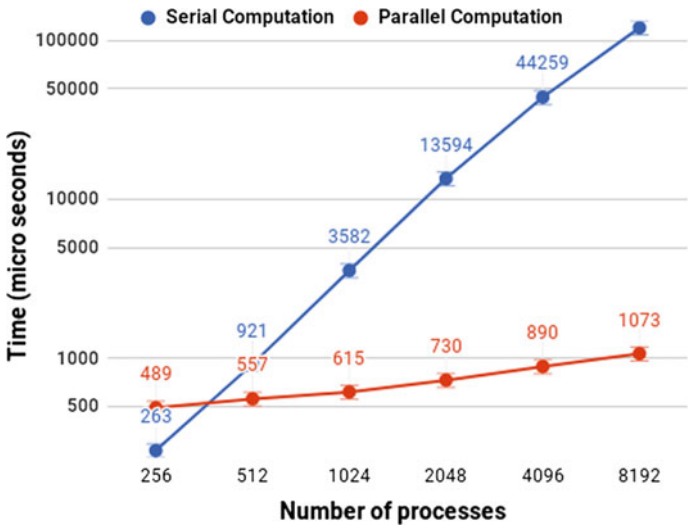


Fig. 7 Graphical representation of the comparison of serial and parallel scheduling algorithms

5 Conclusion

The paper focuses on solving a priority-based algorithm using bitonic sort and the prefix scan, where all the codes are implemented in parallel and executed on GPU, to enable faster execution of the problem.

The entire task is broadly classified into two sub-tasks that are as follows:

1. Sorting the processes based on priority first and then on burst time (execution time) using the parallel code for the bitonic sort by launching multiple kernels.
2. Priority-Based Scheduling is implemented by using the parallel code for prefix scan which again has two stages:

    a. Data that are divided into blocks are first put through initial prefix scan which results in scanned results, but block-wise.
    b. The second stage involves the different blocks to encounter for all the elements that are present in the block previous to it.

The prefix scan is again done by launching multiple kernels.

It is observed that when the number of processes to be scheduled increases, the amount of the time taken for the CPU to schedule these processes also increases drastically. But, however, in the case, when there is an increase in the number of processes that are to be scheduled on the GPU, the amount of the time taken by it to schedule these processes increases but by a relatively very less value.

In other words, the proposed parallel implementation is 10–15 times faster than the serial implementation. Future work involves analyzing the proposed method for non-preemptive scheduling algorithms.

References

1. J.P. Arun, M. Mishra, S.V. Subramaniam, Parallel implementation of MOPSO on GPU using OpenCL and CUDA, in *2011 18th International Conference on High Performance Computing (HiPC)*, pp. 1–10
2. N. Goel, R.B. Garg, Simulation of an optimum multilevel dynamic round robin scheduling algorithm (2013), http://arxiv.org/abs/1309.3096
3. P. Harish, P.J. Narayanan, Accelerating large graph algorithms on the GPU using CUDA, in *Proceedings of the 14th International Conference on High Performance Computing*, p. 197, Heidelberg (2007), http://dl.acm.org/citation.cfm?id=1782174.1782200
4. W.D. Hillis, G.L. Steele Jr., Data parallel algorithms. Commun. ACM **29**(12), 1170–1183 (1986), http://doi.acm.org/10.1145/7902.7903
5. H. Jo, S.T. Hong, J.W. Chang, D.H. Choi, Data encryption on GPU for high-performance database systems. Procedia Comput. Sci. **19**(Supplement C), 147–154 (2013), http://www.sciencedirect.com/science/article/pii/S1877050913006327, The 4th International Conference on Ambient Systems, Networks and Technologies (ANT 2013), The 3rd International Conference on Sustainable Energy Information Technology (SEIT-2013)
6. D.B. Kirk, W.M.W. Hwu, *Programming Massively Parallel Processors: A Hands-on Approach*, 2nd edn. (Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2013)
7. Q. Mu, L. Cui, Y. Song, The implementation and optimization of Bitonic sort algorithm based on CUDA, (2015), http://arxiv.org/abs/1506.01446
8. J. Nickolls, I. Buck, M. Garland, K. Skadron, Scalable parallel programming with cuda. Queue **6**(2), 40–53 (2008), http://doi.acm.org/10.1145/1365490.1365500
9. A. Pandey, P. Singh, N.H. Gebreegziabher, A. Kemal, Chronically evaluated highest instantaneous priority next: a novel algorithm for processor scheduling. J. Comput. Commun. **4**, 146–159 (2016), http://www.scirp.org/JOURNAL/PaperInformation.aspx?PaperID=65949
10. H.B. Parekh, S. Chaudhari, Improved round robin CPU scheduling algorithm: round robin, shortest job first and priority algorithm coupled to increase throughput and decrease waiting time and turnaround time, in *2016 International Conference on Global Trends in Signal Processing, Information Computing and Communication (ICGTSPICC)*, pp. 184–187, Dec 2016
11. N. Satish, M. Harris, M. Garland, Designing efficient sorting algorithms for manycore gpus, in *Proceedings of the 2009 IEEE International Symposium on Parallel & Distributed Processing* (*IPDPS'09*) (IEEE Computer Society, Washington, DC, USA 2009), pp. 1–10, https://doi.org/10.1109/IPDPS.2009.5161005
12. P. Zhang, E. Holk, J. Matty, S. Misurda, M. Zalewski, J. Chu, S. McMillan, A. Lumsdaine, Dynamic parallelism for simple and efficient GPU graph algorithms, in *Proceedings of the 5th Workshop on Irregular Applications: Architectures and Algorithms* (*IA3'15*) (ACM, New York, NY, USA, 2015), pp. 11:1–11:4, http://doi.acm.org/10.1145/2833179.2833189
13. Y. Zhang, J.D. Owens, A quantitative performance analysis model for GPU architectures, in *Proceedings of the 2011 IEEE 17th International Symposium on High Performance Computer Architecture* (HPCA'11) (IEEE Computer Society, Washington, DC, USA, 2011), p. 382, http://dl.acm.org/citation.cfm?id=2014698.2014875

Comparison of Machine Learning Models in Student Result Prediction

Vaibhav Kumar and M. L. Garg

Abstract Prediction of result of students in a particular subject based on their performance in continuous assessment during the semester can be accomplished by various available machine learning models. Every model has its own advantage and limitation due to the algorithm on which they work. Linear regression models have been used very popularly in the area of predictive analytics. Artificial neural networks have also proven their capabilities in prediction. Deep learning techniques are a trend nowadays in data analytics due to their accuracy and performance. This research paper will present a comparison of performance of five popular machine learning models used in predictive analytics—generalized linear model, multilayer perceptron, gradient boost model, Random Forest model, and deep neural network. We have used the result data of students at DIT University, Dehradun, which comprises of schooling marks, continuous assessment marks, and final marks in a subject. On the basis of schooling marks and continuous assessment marks, all these models will predict the final marks of a student in the subject. We will then compare and present the result and performance of these five models.

Keywords Machine learning · Predictive analytics · Linear regression
Neural networks

1 Introduction

It is a great matter of concern among the students of a course and instructors of that course about the result of students in that course. There may be many components of the course which will be divided into continuous assessment and final examination. If we see the curriculum of graduate students in almost all the universities, the continuous assessment of a course including class tests, assignments, and quizzes is conducted throughout the semester and final examination is conducted at the end

V. Kumar (✉) · M. L. Garg
Department of Computer Science & Engineering, DIT University, Dehradun, India
e-mail: vaibhav05cse@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_46

of the semester. On the basis of performance in continuous assessment, one can estimate the result of final examination for a student in a course. This estimation may or may not be accurate, but one can easily give a wide range of result on the basis of performance in continuous assessment.

There are many machine learning techniques which have been used in prediction of performance of students [1, 2]. Many researchers have tried to predict the result of students on the basis of some machine learning technique [3]. This prediction may help students to improve their study in the course so that they can achieve better in final examination. It may also help the instructors of the course in identifying the students who are going to achieve poor result in final examination.

In this research paper, we have used five machine learning models—generalized linear model, multilayer perceptron model of artificial neural network (ANN), random forest model, gradient boost model, and deep neural network—to predict the final result of students in a course. After prediction by each model, we will present a comparison of all these models in this work.

Outline: In this paper, we will present the related work in the field of student result prediction in Sect. 2. Methodologies used in the research paper will be discussed in Sect. 3. Each methodology will be discussed in subsections of Sect. 3. Experiments on the models and their performance will be discussed in Sect. 4. Section 5 will present the survey of results produced by each model. Conclusion of the research and future scope will be discussed in Sect. 6.

2 Related Work

There are many researchers who have worked on predicting the result or performance of students based on certain parameters using different techniques. Stamos T. Karamouzis et al. have used multilayer perceptron model in predicting the graduation outcome of students [4]. Shaobo Huang et al. have used multivariate linear regression model for predicting the academic performance of students in engineering dynamics course [5]. Pauziah Mohd Arsad et al. have used neural network model to predict final grades in graduation based on the grades in initial years [6]. Bogdan Oancea et al. have used multilayer perceptron model to predict the dropout of students based on their data at the time of enrollment in the course [2]. Dr. M. Shakil et al. have used multiple linear regression model to predict the students' grades in mathematics based on their performance in internal tests [7]. Hashima Hamsa et al. have used decision tree model and fuzzy genetic algorithm to predict the academic performance of students based on their marks in sessional examinations [8]. Sparse linear and low-rank matrix factorization models have been used by Agoritsa Polyzou et al. in grade prediction of students [9].

The above is not a limited description of related work in the area of result prediction of students. Many data mining techniques have been used in predicting the various types of performance of students [10].

3 Methodology

There are various machine learning methods which are used in predictive analytics. Each model works on an algorithm and trained on the previous datasets. A model after training will be able to predict the output value on the given set of input values. In this section, we will present a brief description of the three above discussed models.

3.1 Generalized Linear Model

Generalized linear model, introduced by John Nelder and Robert Wedderburn, is an extension of ordinary linear regression models which is useful in the cases when dependent variable does not have normal error distribution [11]. This model has a special characteristic that it unifies the statistical models like linear regression, logistic regression, and Poisson's regression models [12]. It is represented in Fig. 1.

In generalized linear model, it is assumed that the dependent variable Y follows a distribution in exponential family. In this model, the mean μ depends on the independent variables x_i. This mean is assumed to be a nonlinear function of $x_i\beta$ as:

$$E(Y) = \mu = g - 1(x_i\beta) \tag{1}$$

where E(Y) is the expected value of Y, β is unknown parameter, and *g* is called the link function. Link function in GLM specifies the linking relation between linear model and dependent variable.

GLM consists of three elements—Probability distribution, linear predictor, and link function. The probability distribution originates from the exponential family. The linear predictor, denoted by η, combines the information about the independent variables in the model. It is expressed as the linear combination of unknown parameters as $\eta = x_i\beta$. The link function mainly relates this linear predictor with the mean of distribution function.

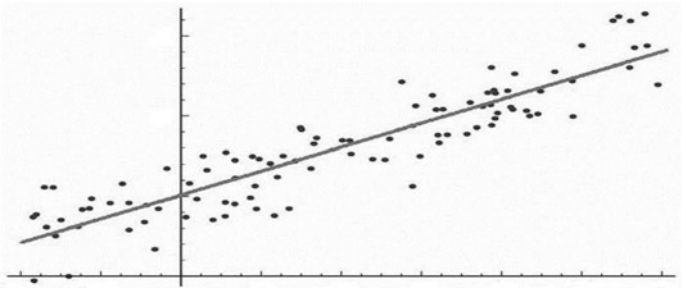


Fig. 1 Generalized linear model

There are mainly two fitting methods used in GLM—maximum likelihood and Bayesian. The estimation process in maximum likelihood method may be found using Newton–Raphson method as:

$$B(n+1) = \beta(n) + \tau\_1(\beta(n)u\beta(n)) \tag{2}$$

where τ (β(n)) is the observed information matrix and ($u\beta$(n)) is the score function. Bayesian method is mainly used for approximation of posterior distribution.

GLM models are used in such cases where linear regression models have limitation in work. Linear models which work on normal distribution may have limitation in modeling measured proportions. It has limitation to observe data where the variance in data increases with the mean.

3.2 Multilayer Perceptron

Multilayer perceptron is a feedforward model of artificial neural network. It maps a set of input data onto appropriate outputs. This model consists of an input layer, an output layer, and one or more than one hidden layers between input and output layers. These layers consist of a set of processing neurons. Every neuron of a layer is connected to each neuron of its neighboring layers [13]. The connecting paths are unidirectional in the forward direction—from input layer toward output layer. That is why this model is termed as feedforward model. For the training of multilayer perceptrons, backpropagation learning method is mainly used [14]. The architecture of this model is represented in Fig. 2.

The net input X_{in} and net output Y_{out} of this network can be obtained as:

$$X_{in} = X \times W \tag{3}$$

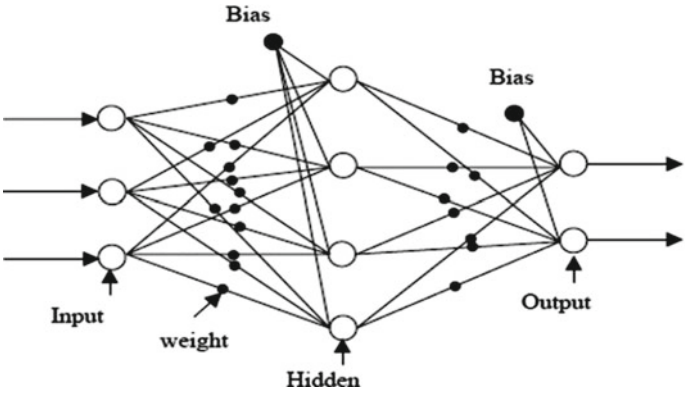


Fig. 2 Multilayer perceptron model

and

$$Y_{out} = H_{out} \times V \tag{4}$$

where X is the input vector, W is the weigh matrix of weights between input and hidden layer, H_{out} is the output from hidden layer, and V is weight matrix of weights between hidden layer and output layer. When an input vector is applied at the input layer of the network, it produces the corresponding output vector. The process of learning is followed to match the produced output with desired output. In the process of learning, weights associated with the interconnections between layers are changed at each iteration of training. This change in weight is based on the error calculated in the produced output. This process is followed in backpropagation learning, a supervised learning approach. This is an extension of least square method [15].

If the error in *n*th output node at *i*th iteration is denoted by $e_n(i)$, then it can be minimized as:

$$\varepsilon(i) = {}_{12}\Sigma e_{n2n}(i) \tag{5}$$

And the change in weight can be obtained as:

$$\Delta w_{nj}(i) = -\eta_{\partial\varepsilon(i)/\partial vn(i)} y_j(i) \tag{6}$$

where y_j is the output of previous *j*th neural node and η is the learning rate which used for convergence of weights.

The multilayer perceptrons with backpropagation learning have been used very widely in many classification and prediction applications. With the feature of machine learning, they have advantage over ordinary linear regression models [16].

3.3 Gradient Boost Model

Gradient boost is a machine learning method popularly used in regression and classification to develop predictive models [17, 18]. Gradient boosting consists of three elements—a loss function, a weak learner, and an additive model. The loss function is used for optimization, and it is used as per the requirement of problem. Weak learners are used for making predictions. Decision trees are mainly used as weak learners in this model. Additive models are used to add weak learners which can minimize the loss function.

Decision trees are generally used as base learners in gradient boosting. A gradient boosting method has been proposed by Friedman which is used to improve the learning with gradient boosting [19]. According to Friedman, a gradient boosting model using decision trees can be formulated as:

$$F_m(x) = F_{m-1}(x) + \gamma_m h_m(x) \tag{7}$$

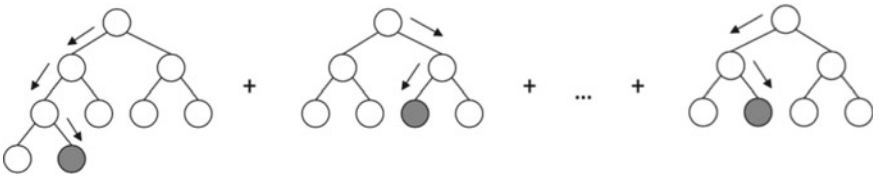


Fig. 3 Gradient boosting model with decision trees

and

$$\gamma_m = \arg\min_{\gamma} \sum_{i=1n} L(y_i, F_{m-1}(x_i) + \gamma\, h_m(x_i)) \tag{8}$$

where F_m is the model at the *m*th stage of gradient boosting, x is the input value, γ is the coefficient, and h_m is the decision tree at *m*th stage.

There is a parameter called number of nodes in the tree which is used in this model. This parameter allows the level of interaction between variables. It can be adjusted according to the dataset used for training. A gradient boosting model with decision trees is represented in Fig. 3.

3.4 Random Forest

Random Forest is a machine learning method belongs to the class of ensemble learning. It is widely used for classification and regression problems [20]. This model is formed with the decision trees, and it predicts the output as mean of the prediction of each individual decision trees in case of regression. In case of classification, it predicts output as the mode of the outputs of each individual decision tree [21]. A typical Random Forest can be represented as given in Fig. 4.

When this model is used in regression, leaf nodes of each individual tree predict the real valued numbers. In this model, the data is split at some split points for each independent variable based on the homogeneity of the data [22].

A technique known as bagging is used to train the decision trees. Let X is the set of independent variables, Y is the set of dependent variables, and $b = 1, \ldots, B$ is the steps of bagging in which in which random samples are selected at each step. Let X_b and Y_b are the training examples to train the decision tree f_b. After the training process, the model can predict for an unknown sample x' as:

$$f' = {}_{1B}\, \Sigma_{b=1B}\, f_b(x') \tag{9}$$

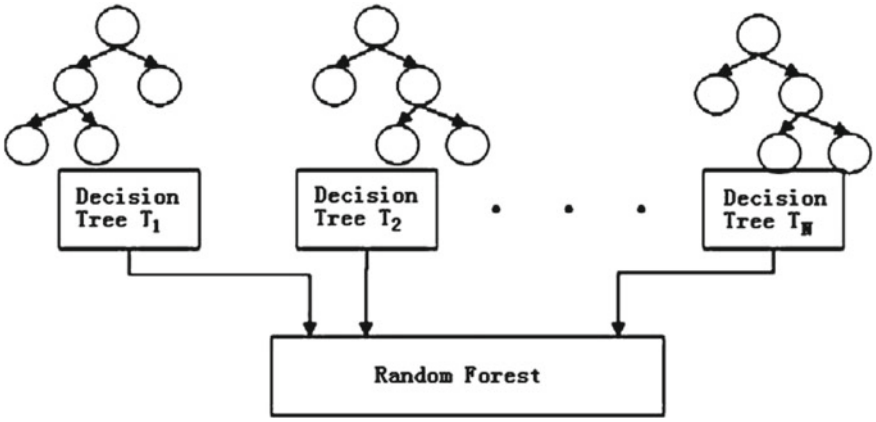


Fig. 4 Random forest with N no. of decision trees

3.5 Deep Neural Network

A deep neural network (DNN) is a variety of artificial neural network which has more than one hidden layers [23]. DNNs have special feature that they can model the complex relationships which are nonlinear. DNNs generally have the feedforward architecture. Some recurrent architecture has also been used by researchers in language modeling [24]. Convolutional neural networks have been used as deep neural networks very popularly in the area of image processing [18]. A typical architecture of deep neural network is represented in Fig. 5.

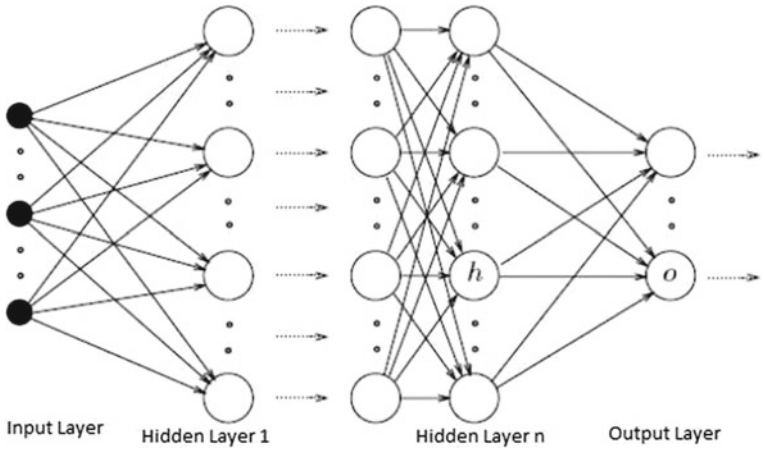


Fig. 5 Deep neural network with n-hidden layers

All the processing DNNs are very much similar to the feedforward model of artificial neural networks. Backpropagation learning can be performed to find the matching between produced outputs and desired output. The change in weight in this process can be obtained as:

$$w_{ij}(t+1) = w_{ij}(t) + \eta_{\partial C/\partial wij} + \xi(t) \tag{10}$$

where η is the learning rate, C is the cost function, ξ is the stochastic term, and w_{ij} is the weight associated with the interconnection between *i*th node of one layer and *j*th node of next layer.

3.6 Dataset

The dataset used in the experiment is the result of students in the subject artificial intelligence at DIT University, Dehradun, in 2015. This dataset consists of the records of 382 students registered in the course during the semester. This record consists of the marks of students in schooling marks, continuous assessment during the semester, and the final marks in the course. The description of attributes in the dataset is given in Table 1.

The idea behind taking the schooling marks is to consider the potential of students during their school times. The feature scaling method is used to normalize the dataset in preprocessing step to bring all the values in the range of [0, 1] as:

$$X' = \frac{X - min(x)}{[\max(x) - min(x)]} \tag{11}$$

Out of 382 datasets, 300 datasets have been used for training and 82 datasets are used as sample for testing and validation in all the learning methods.

Table 1 Description of attributes

S. no	Attribute	Description	Component
1	HSC	Marks in higher secondary	Schooling marks
2	SSC	Marks in senior secondary	
3	M1	Marks in midterm-1 examinations	Continuous assessment
4	M2	Marks in midterm-2 examinations	
5	CT	Marks in class tests	
6	ASS	Marks in assignments	
7	QZ	Marks in quizzes	
8	Total	Final marks	Final examination

4 Experiment and Results

4.1 Generalized Linear Model

In this model, a generalized linear model is used with Gaussian distribution. It has final marks of student as response variable and rest of the seven attributes are independent variables. It has given 95.52% accuracy in prediction. A total of 82 samples from dataset are used to test the performance. This performance of prediction is given in Fig. 6.

4.2 Multilayer Perceptron

A 7-input, 12-hidden, 1-output feedforward neural network architecture is used as multilayer perceptron model. Resilient backpropagation algorithm is used as learning algorithm. Learning rate is kept at 0.5, and threshold is set to 0.01. It has given 96.38% accuracy in prediction. Its performance of prediction in a sample of 82 is given in Fig. 7.

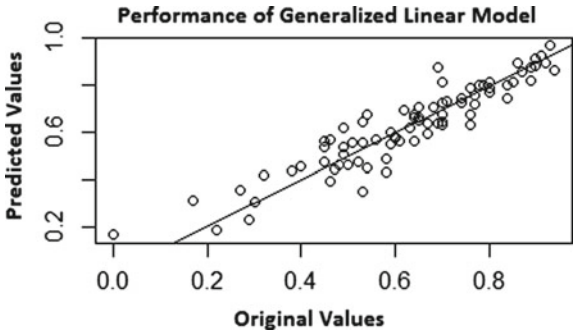


Fig. 6 Performance in prediction by generalized linear model

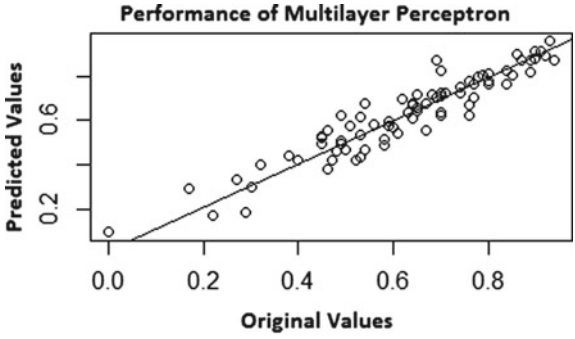


Fig. 7 Performance in prediction by multilayer perceptron model

4.3 Gradient Boost Model

A gradient boosting tree with number 1000 decision trees with a depth of 8 is used in this model. Learning rate is set to 0.01 and seed for random number is set to 1112. It has given 98.26% accuracy in prediction. Its performance in the sample is given in Fig. 8.

4.4 Random Forest Model

A Random Forest model with 1000 decision trees is used in this model. Three numbers of variables randomly sampled as candidate for each split during training. Seed is set to 1122. It has given 96.10% accuracy in prediction. Its performance of prediction in the sample is given in Fig. 9.

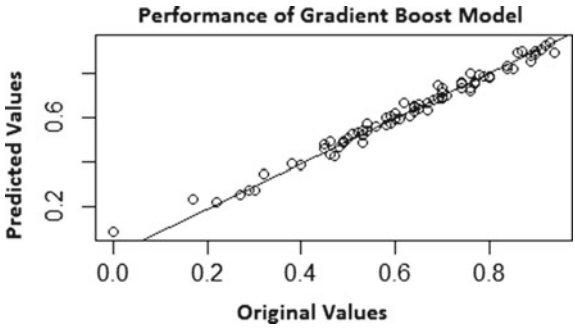


Fig. 8 Performance in prediction by gradient boost model

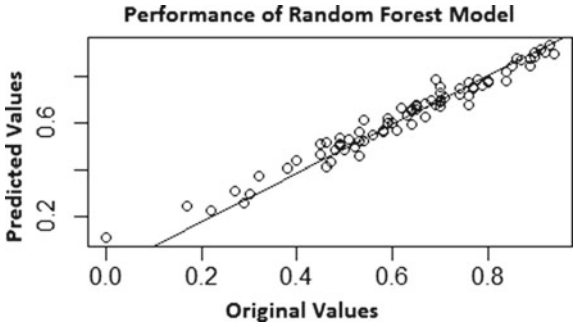


Fig. 9 Performance in prediction by Random Forest model

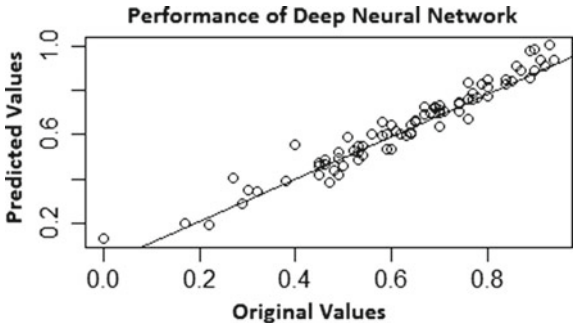


Fig. 10 Performance in prediction by deep neural network

4.5 Deep Neural Network

A deep neural network with three hidden layers each consisting 12 neurons is used in this model. Rectifier function is used as activation function of neuron in this model. After 1000 epochs of training, it has given 97% of accuracy in prediction. Its performance of prediction in the sample is given in Fig. 10.

5 Survey of Results

In this section, we will present and discuss the survey of performance of all the five models. In the next sections, we will use the notations as GLM: generalized linear model; MLP: multilayer perceptron; GBM: gradient boost model; RFM: random forest model; and DNN: deep neural network.

5.1 Training Performance

Performance in training of each model is given in Table 2.

Table 2 Training performance

	GLM	MLP	GBM	RFM	DNN
Mean squared error	0.0046	0.0034	**0.0006**	0.0057	0.0025
Root mean squared error	0.0675	0.0468	**0.0254**	0.0757	0.0505
Mean absolute error	0.0507	0.0321	**0.0186**	0.0561	0.0385

Table 3 Accuracy in prediction

	Run 1	Run 2	Run 3	Run 4	Run 5	Average accuracy
	Correct (%)	Correct (%)	Correct (%)	Correct (%)	Correct (%)	
GLM	94.37	98.90	93.51	98.81	92.00	95.52
MLP	91.78	98.90	94.81	96.43	100.00	96.38
GBM	**95.71**	**100.00**	**96.10**	**96.43**	**98.04**	**98.26**
RFM	94.37	98.90	94.81	96.43	96.00	96.10
DNN	97.01	100.00	98.70	96.55	98.00	97.05

In the above table, we can see that the gradient boost model has best performance in training.

5.2 Accuracy in Prediction

The accuracy in prediction by each model in five runs and average accuracy is given in Table 3.

It can be seen in the last table that gradient boost model has highest accuracy in prediction.

5.3 Correlation of Prediction with Actual Values

A sample of size 82 is taken to check the correlation and covariance of predicted values and original values. It is presented in Table 4.

The gradient boost model has best correlation covariance among all the models.

Table 4 Correlation and covariance

	Correlation	Covariance
GLM	0.931	0.032
MLP	0.949	0.033
GBM	**0.992**	**0.031**
RFM	0.983	0.034
DNN	0.974	0.036

Table 5 Percent relative efficiency (in %)

	GLM	MLP	GBM	RFM	DNN
GLM	100.00	75.22	125.77	14.19	55.85
MLP	132.94	100.00	167.19	18.86	74.25
RFM	79.51	59.81	100.00	11.28	44.41
GBM	**704.73**	**530.12**	**886.31**	**100.00**	**393.60**
DNN	179.05	134.69	225.18	25.41	100.00

5.4 Percent Relative Efficiency

To find out the performance of a model in comparison of other models can be obtained by calculating the percent relative efficiency (PRE). It will show how efficient a model is in from of the rest of the similar models. To calculate the PRE, mean squared error of the new model and the compared old model is used. It can be calculated as:

$$\text{PRE(new)} = MSE(old) * 100/MSE(new)$$

The PRE of each model as per the above formula in comparison with other models is given in Table 5.

In the above table, it can be seen that the gradient boost model is the strongest model in comparison with rest other models in student result prediction task.

6 Conclusion and Future Scope

This research paper has presented an experimental survey of popular machine leaning models used in predictive analytics. There are five models used to predict the final result of students in a course. Every model has predicted the result with an accuracy of more than 95%. In all the five models, gradient boost model has proven itself as the strongest predictive model in task. It has shown the accuracy of more than 98% and is the strongest model in front of rest of the four models as shown in percent relative efficiency. These models can be used by the students as well as the instructors of the course to predict the result in a course. It can be helpful in identifying weaker student, and timely corrective steps can be taken to improve their performance. Further, there is a scope of research in this domain. All the models discussed in the paper can give better performance in some other generalized datasets. Parameter tuning can also help to improve the performance of these models. There is a scope to create new features in these models so that they can be applied in many domains with better performance.

References

1. O.C. Asogwa, A.V. Oladugba, Prediction of students academic performance rates using artificial neural networks. Am. J. Appl. Math. Stat. **3**(4), 151–155 (2015)
2. B. Oancea, R. Dragoescu, S. Ciucu, Predicting students' results in higher education using neural networks, in *International Conference on Applied Information and Communication Technologies*, Jelgava, Latvia (2013)
3. H. Agrawal, H. Mavani, Student performance prediction using machine learning. Int. J. Eng. Res. Technol. **4**(03) (2015)
4. S.T. Karamouzis, A. Vrettos, An artificial neural network for predicting student graduation outcomes, in *World Congress on Engineering and Computer Science*, San Francisco, USA (2008)
5. S. Huang, N. Fang, Regression models of predicting student academic performance in an engineering dynamics course. Comput. Educ. 133–145 (2013)
6. P.M. Arsad, N. Buniyamin, J.-L. Ab Manan, Neural network model to predict electrical students' academic performance, in *4th International Congress on Engineering Education (ICEED)*, Georgetown, Malaysia (2012)
7. Dr. M. Shakil, A multiple linear regression model to predict the student's final grade in a mathematics class. Sam Houston State University
8. H. Hamsa, S. Indiradevi, J.J. Kizhakkethottam, Student academic performance prediction model using decision tree and fuzzy genetic algorithm, in *Global Collegium in Recent Advancements and Effectual Researches in Engineering, Science and Technology* (2016)
9. A. Polyzou, G. Karypis, Grade prediction with models specific to students and courses. Int. J. Data Sci. Anal. **2**, 159–171 (2016)
10. A.M. Shahiria, W. Husaina, N.A. Rashida, A review on predicting student's performance using data mining techniques. Proc. Comput. Sci. **72**, 414–422 (2015)
11. S. Eguchi, Model Comparison for generalized linear models with dependent observations. Econom. Stat. **59** (2017)
12. J. Nelder, R. Wedderburn, Generalized linear models. J. R. Stat. Soc. (1972)
13. F. Rosenblatt, Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms. Spartan Books (1961)
14. D. Rumelhart, G. Hinton, R.J. Williams, Learning internal representations by error propagation, in *Parallel Distribute Processing: Explorations in the Microstructure of Cognition*, vol. 1 (1986)
15. S. Haykin, Neural Networks: A Comprehensive Foundation, 2nd edn. (Prentice Hall, 2012)
16. J. Kan, Evaluation of mining engineering technology innovation ability and application based on BP neural network, in *International Conference on Industrial Technology and Management (ICITM)* (2017)
17. Y. Zhang, A. Haghani, A gradient boosting method to improve travel time prediction. Trans. Res. Part C (2015)
18. Y. LeCun et al., Gradient based learning applied in document recognition, in *Proceedings of the IEEE*, vol. 86 (1998)
19. J.H. Friedman, Greedy Function Approximation: A Gradient Boosting Machine (1999)
20. T.K. Ho, Random decision forests, in *3rd International Conference on Document Analysis and Recognition*, Montreal (1995)
21. T.K. Ho, The random subspace method for constructing decision forests. IEEE Trans. Pattern Anal. Mach. Intell. **20** (1998)
22. T.K. Ho, A data complexity analysis of comparative advantages of decision forest constructions. Pattern Anal. Appl. (2002)
23. J. Schmidhuber, Deep Learning in Neural Networks. Technical Report IDSIA-03-14 arXiv: 1404.7828
24. J. Schmidhuber, LSTM recurrent networks learn simple context free and context sensitive languages. IEEE Trans. Neural Netw. **12** (2001)
25. V.K. Ojha, A. Abraham, V. Snasel, Metaheuristic design of feedforward neural networks: a review of two decades of research. Eng. Appl. Artif. Intell. **60** (2017)

Image Steganography Based on Random Pixel Addition and Discrete Cosine Transform (DCT)

Rohit Patidar, Balwant Prajapat and Anwar Sakreja

Abstract As the boom in digital technology has gained momentum, there has been a great increase in the importance of digital images. Though encryption mechanisms are able to protect the data from the unauthorized intruders, still they remain vulnerable to threats and attacks on the encryption method being breached. Therefore, steganography plays an effective role as being a strong technique that evades the intruder from identifying or recognizing the confidential digital image under a normal cover data. This paper presents a steganographic approach based on Discrete Cosine Transform and subsequent pixel injection for achieving image steganography. It has been shown that the proposed technique achieves high values of PSNR and throughput while yielding low values of MSE.

Keywords Image steganography · Pixel injection
Discrete cosine transform (DCT) · Peak signal-to-noise ratio (PSNR)
Mean square error (MSE)

1 Introduction

An image can be represented as a two-dimensional function I, where $I = f(x, y)$ and x and y are spatial coordinates. Steganography is a process of conveying confidential messages and secretive images through cover images in a confidential manner that only the receiver can recognize and find out the existence of a message. The purpose of steganography is to hide the image in such a manner that the real existence of the hidden message is incomprehensible and concealed [1]. Steganography thus gives another strong layer of security and protection over the secret message that shall be embedded in another form of medium so that the conveyed information is understandable to the intended receiver but concealed from everyone else. The fundamental attributes of image steganography are [2]:

R. Patidar (✉) · B. Prajapat · A. Sakreja
Department of CSE, VITM, Indore, India
e-mail: rohitpatidar180@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_47

- **Imperceptibility of Hidden Image**: This requirement infers that the embedded messages cannot be discerned or recognized by the human eye.
- **Image Embedding Capacity**: This refers to the capacity of embedding the concealed image.
- **Image Security**: This necessitates the secret stegano image's robustness and also that it is unerring.

2 Image Steganography in Transform Domain

Inserting a secret image in the veil of a cover image and making it completely imperceptible for adversaries is a challenging task. Inserting the pixels directly does not render imperceptibility [2]. Hence, steganography in transform domain yields better results as the secret image occupies a place which is imperceptible to the human eye if the amount of resulting changes is not substantial. Such a transformation can be defined as:

$$I(f)F \leftrightarrow F^{-1}\, I(x, y, t) \tag{1}$$

Here I(f) represents the image in transform domain and I(x, y, t) represents the image in the spatial-time domain. The transform domain can be the complex frequency domain. The choice of the transform domain depends upon the type of signal under consideration. The following provides the nuances of selecting an appropriate domain.

3 Steganography in the Discrete Cosine Transform (DCT)

One of the key features of image steganography is the abruptly changing nature of image pixel values not complying with Dirichlet's conditions for the existence of the Fourier Transform given by [3]:

$$X(f) = \int_{-\infty}^{+\infty} x(t) e^{j2\pi f t} dt \tag{2}$$

Hence, the kernel of the transform should be abruptly changing in nature which could separate the image into different frequency components. As it occurs, the Discrete Cosine Transform (DCT) fits the bill. Mathematically, the DCT is defined as:

$$y(k) = w(k) \sum_{i=1}^{N} x(n) \cos\left(\frac{\pi(2n-1)(k-1)}{2N}\right) \tag{3}$$

Here,

$$w(k) = 1/\sqrt{N};\ \mathrm{k} = 1 \tag{4}$$

$$w(k) = \sqrt{2/N};\ 2 < \mathrm{k} < \mathrm{N} \tag{5}$$

The DCT has the property of separating the image signal into its DC and different AC components. Another property of the DCT is the concentration of the DCT coefficients in the inner elements of the DCT matrix. This can be expressed as:

$$\begin{matrix} 1 & 5.1 & 0 \\ 3.5 & 1.9 & 0 \\ 0 & 0 & 0 \end{matrix}$$

Let the above matrix denote the DCT coefficient matrix with the LSBs in the outer elements of the matrix while the MSBs in the inner elements of the matrix. If the stegano image inserted in the region of empty LSBs, then the image would visually render imperceptibility.

Figure 1 illustrates the discussed concept of LSB pixel injection mechanism. It is worth mentioning here that the injection is done in the transform domain due to the frequency separation property of the DCT. Moreover, the convention Fourier methods fail in case of image steganography due to the non-compliance of image pixel values with Dirichlet's conditions.

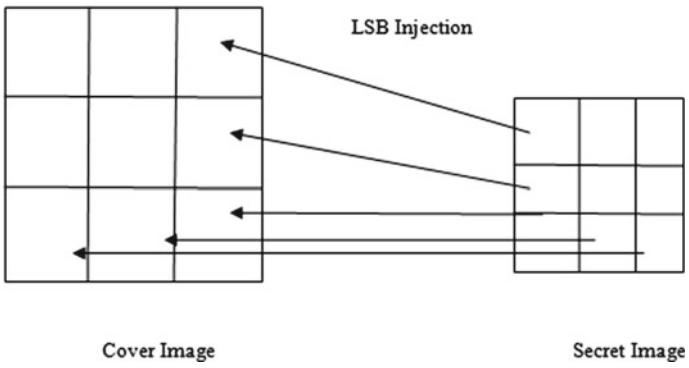


Fig. 1 Illustration of DCT-based image steganography

4 Proposed Methodology

The proposed methodology is explained under the following heads.

4.1 Image Embedding Mechanism

a. The first step is to select the original image and cover image.
b. The DCT is applied to both images which yields images broken and separated into different and varying frequency sub-bands of 8×8 blocks.
c. In the next step, DCT coefficients are rounded off to the nearest value and approximated as it is time-consuming and also requires more space complexity to store and execute upon fractional values.
d. Then the DCT coefficient values of original image are embedded into the cover image's LSB positions.

4.2 Random Pixel Generation in LSB Positions

a. Random pixel values are generated for the 8×8 block using the following mathematical relation:

$$P_{LSB}^{rand} \leftarrow rand(\mathrm{R_{LSB}},\ \mathrm{C_{LSB}}) \tag{6}$$

Here, $\mathrm{R_{LSB}}$ and $\mathrm{C_{LSB}}$ denote the row and columns of the LSB bits, respectively.

b. Finally generate the stegano image using Inverse Discrete Cosine Transform (IDCT) with the LSB pixel injection technique. The random pixel values can be generated using a pseudo-random sequence generator which generates seemingly or apparently random values from the perspective of adversaries.

4.3 Image Extraction Mechanism

a. In this extraction process, the random pixels injected into stegano image are removed after the computation of DCT of the stegano image, from the LSB positions where the random pixels were originally inserted.
b. The Inverse Discrete Cosine Transform (IDCT) of the above image is computed and the original image is obtained.
c. The Mean Square Error (MSE), throughput, and peak signal-to-noise ratio (PSNR) are calculated to evaluate the performance of the proposed system.

5 Results and Discussions

The parameters used for the evaluation of the proposed algorithm are:

$$\text{RMSE} = \sqrt{\sum\sum[I(x, y) - K(x, y)]^2}/\text{MN} \tag{7}$$

Here, M and N represent the number of pixel along x- and y-axes, respectively.

$$PSNR = 10\log_{10}(MN)^2/MSE \tag{8}$$

$$Throughput = \frac{D}{t} \tag{9}$$

Here,
D represents data size.
t represents time of execution.

Root Mean Square Error (RMSE) indicates the amount of difference between the original secret image and the recovered image from the stegano image. It is desirable to attain a low value of RMSE.

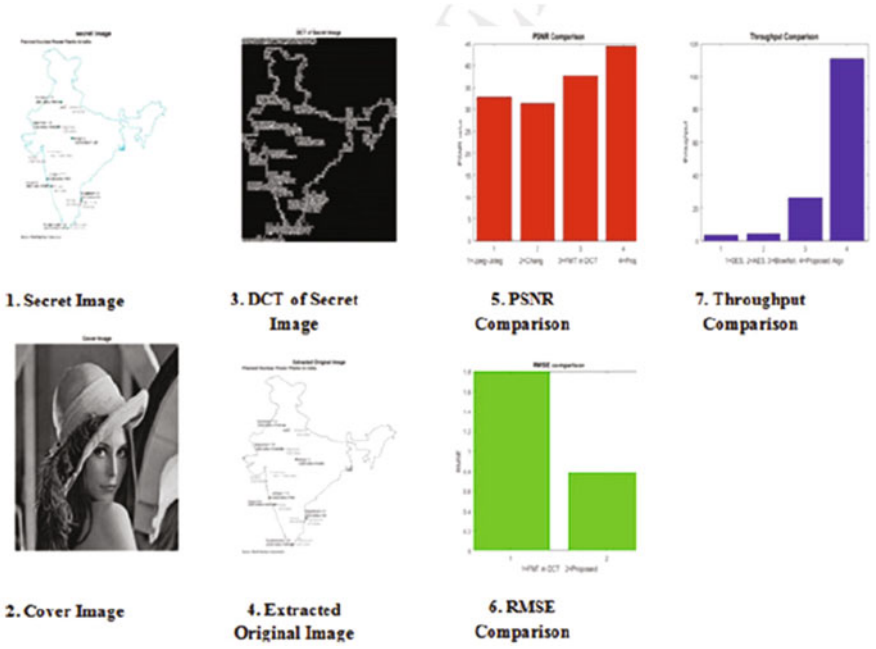


Fig. 2 Results obtained

Peak signal-to-noise ratio is the ratio of signal power to noise power and is a measure of the quality of the recovered image under the effects of residual errors and noise. A high value of PSNR is desirable.

Throughput is an indicator of the processing ability of the system. High values of throughput indicate that the algorithm can process large amounts of data in lesser amounts of time. Figure 2 depicts the various stages of the steganographic process.

The results obtained shown in the form of figures depict the secret image, cover image, DCT of secret image, extracted original image, PSNR comparison with previous work, viz. Jpeg–Jsteg, FMT in DCT, Chang et al. [4]. The MSE comparison is done with FMT in DCT. The throughput is compared to standard encryption algorithms such as DES, AES, and Blowfish. The obtained values of RMSE are found to be 0.7841, PSNR is found to be 44.2224, and the throughput is found to be 109.5263.

6 Conclusion and Future Scope

This paper presents an image steganographic mechanism in the transform domain to minimize visual perceptibility by adversaries. The DCT is used to embed the secret image in the cover image's LSB locations. Moreover, random pixels are added to the cover images. It can be seen that the proposed mechanism attains better MSE and PSNR values compared to standard previous techniques [4] and throughput compared to standard encryption algorithms.

Future researchers can use emerging transform domain tools such as the Contourlet Transform or the Maximum Overlap Discrete Wavelet Transform (MODWT) with an aim of better spectral resolution which may yield better values of PSNR. The applications of the technique can be used for securing highly classified images used in military and defense applications.

References

1. R. Ridzon, D. Levisky, T. Kanocz, Information hiding within still images based on the DCT coefficients flipping and encryption, in *52nd International Symposium*, September 2010, pp. 147–150
2. N. Provos, P. Honeyman, Hide and seek: an introduction to steganography. IEEE Secur. Priv. **1**(3), 32–43 (2003)
3. Gonzalez, Woods, Digital Image Processing, 4th edn. (Pearson Publications)
4. I. Banerjee, Dr. S. Bahttacharya, Dr. G. Sanyal, Robust Image Steganography with Pixel Factor Mapping (IEEE, 2014)

A Comparative Analysis of Medical Image Segmentation

Neeraj Shrivastava and Jyoti Bharti

Abstract Image segmentation is the technique of dividing an image into one of kind regions (segments) following some homogeneous criteria. It is an important technique in any image analysis process. Segmentation of medical images like magnetic resonance images, mammogram, cardiac Magnetic resonance (MRIs) images helps in detection and diagnosis of breast tumor, brain tumor, etc. We need a strong and efficient image segmentation method, as most segmentation methods are computationally high priced, and the amount of medical imaging information is growing and very sensitive. In this paper, we delve into different methods available for medical image segmentation with their standpoints. We also compare the two authors' results based on the parameters True Positive Factor (TPF), True Negative Factor (TNF), and Sum of True Volume Factor (SVTF).

Keywords Magnetic resonance images · Mammograms · Micro-calcification Cardiac · True positive fraction · True negative fraction · Sum of volume fraction

1 Introduction

Segmentation is an important step in any image analysis process. Image segmentation is the splitting of an image into one of kind regions according to few homogeneous criteria. It is the first step in any image processing technique. The regions with somewhat similar intensity pixels can provide important cues specifying a region and extracting semantic objects as masses [1].

N. Shrivastava (✉) · J. Bharti
Department of Computer Science & Engineering, MANIT, Bhopal, Madhya Pradesh, India
e-mail: neeraj0209@gmail.com

J. Bharti
e-mail: jyoti2202@gmail.com

N. Shrivastava
Department of Computer Science & Engineering, IES IPS Academy, Indore, Madhya Pradesh, India

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_48

The segmentation of medical images is the most critical image-related application and visualization process. It provides help to medical doctors to identify the disease in the body of any patient without doing any surgery process. It reduces the image analysis time, to find the vicinity of a lesion and to determine the chance of disease. A lesion is any abnormal infection or harm inside the body part or organ [2].

Medical image has been presented in various forms like breast MRI, mammograms, micro-calcifications, cardiac images. MRIs has been mostly used for medical tumor detection like brain tumor and breast tumor. The detection and prognosis of the brain tumor from the MRI is continuously reducing the rate of casualties. It is very difficult to treat brain tumor due to the fact brain has a very complex structure and structure incorporate tissue that is connected with every different in complicated ways. MRI gives good contrast for the diagnosis of soft breast tissues. Mammograms are an image that can be found by mammography. Mammography is the method of using low power x-rays to have a look at the human breast and is used as a diagnostic and screening tool. Micro-calcification is small granular deposits of calcium that can be seen on the mammogram as tiny bright spots [1–12].

There are numerous image segmentation techniques available like seed region growing, k-means, edge-based image segmentation, watershed segmentation, fuzzy logic-based segmentation, genetic algorithm, neural network-based segmentation, normalized cut, split and merge. But medical image segmentation requires fast and accurate result because diagnosis end result is noticeably relying upon on the segmentation end result. If segmentation method fails to segment image properly, then it may cause harm dangerously [11, 13].

Many segmentation algorithms are expensive, particularly when dealing with large medical datasets. Segmentation of image is completed, simply before the operation in addition during the operation. Furthermore, the amount of data available for any given affected person is increasing; making fast segmentation algorithm is very crucial [8].

The rest of the paper is prepared as follows: Sect. 2 describes the review of segmentation of medical images provided by different authors. In Sect. 3, we explain different performance measures that are available for image segmentation methods like True Positive Factor (TPF), True Negative Factor (TNF), and Sum of True Volume Factor (SVTF). In Sect. 4, we analyze result of Al Faris et al. [3] and Usman and Rajpoot [10]. In Sect. 5, we conclude the review process of the paper, and finally in Sect. 5, references are presented.

2 Literature Review

Afifi et al. [2] proposed region growing segmentation method for MRIs. It combines the local search process with the traditional seed region growing to gain higher performance. It automatically finds the seed point for region growing and finds threshold using an average of the maximum and the minimal gray value inside the image. The algorithm is tested on images taken from the simulated brain database of McGill University. Melouah and Layachi [1] proposed an algorithm for automatic selection

of seed point for seed region growing in mammograms. They calculate threshold using mean maximum raw thresholding algorithm (MMRT). Mean maximum raw (MMR) is calculated using the sum of the row intensity of the row divided by number of rows. Apply threshold of the image for black and white intensity, and divide the image into black and white region. Black region will be ignored, and white is declared as suspected region. Revert the original image, and then they apply k-nearest neighbors algorithm (KNN). That determines all the statistical features of base entries, and finds the seed point. Then KNN applies to the growing seed region in image. They tested their result on MiniMIAS database furnished by way of Mammographic Image Analysis Society (MIAS).

Al Faris et al. [3] presented computer-aided segmentation for breast MRI tumor. Magnetic Resonance Imaging (MRI) has most useful for breast tumor detection. They also used mean maximum raw thresholding (MMRT) for thresholding of an image. The usage of MMRT divides the image into two intensity regions, i.e., white region and black region. White region is considered as suspected region. Appling morphological operation could eliminate the undesirable small white speckles inside the image. The speckles do not now belong to tumor regions, thus beautify the boundary of suspected regions. Then they calculated region's density by dividing the area of each region with a mean intensity of that region. Sort the density value of suspected regions and declare the highest density value as seed point. After calculating seed, they apply seed region growing segmentation on the RIDER image dataset.

Malek et al. [4, 5], proposed an algorithm for segmentation of micro-calcifications. In this, they first identify the initial seed using the region maxima. For these, they calculate regional maxima of the image and identify the location of regional maxima, and they set regional maxima as "1" and all other settings as "0". Then they create matrix of n * 1 size in which n is the number of regional maxima available. Apply dilation operation and calculate local maxima and consider an initial seed point to average of local maxima. Now using these initial seed points, they segment the mammogram (obtained from the National Cancer Society, Malaysia) using seed region growing segmentation.

Eklund et al. [6] reviewed medical image processing on the GPU. In medical imaging, GPUs are in few cases critical for practical use of computational stressful algorithms. In this paper, they cover assessment on GPU acceleration of simple image processing operations like filtering, interpolation, histogram estimation, and distance transforms. Figure 1 shows the result of segmentation method proposed by Malek et al. [5]. Smistad et al. [8] focused on performance of numerous image segmentation techniques for graphical processing units like thresholding, region growing, morphology, watershed, active contour.

Petitjean and Dacher [7] reviewed segmentation methods in cardiac MR images. Cardiovascular illnesses are the principal cause of death in any country. Treatment of the patient is highly depending on these pathologies cardiac imaging modalities that are echography, CT (computerized tomography), cardiac MRI. In this paper, author reviewed different segmentation methods for segmentation of cardiac image with their result. Figure 2 shows the segmentation result by 3D AAM [14].

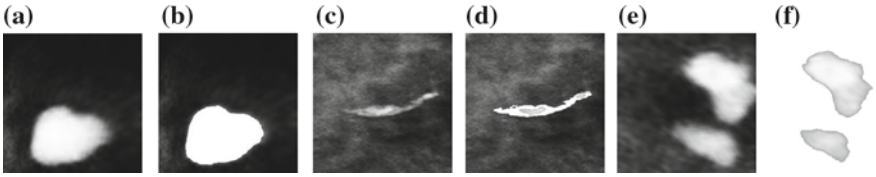


Fig. 1 **a**, **c**, **d** Shows the original image, **b**, **d**, **f** shows the segmentation outcome obtained from Malek et al. [4]

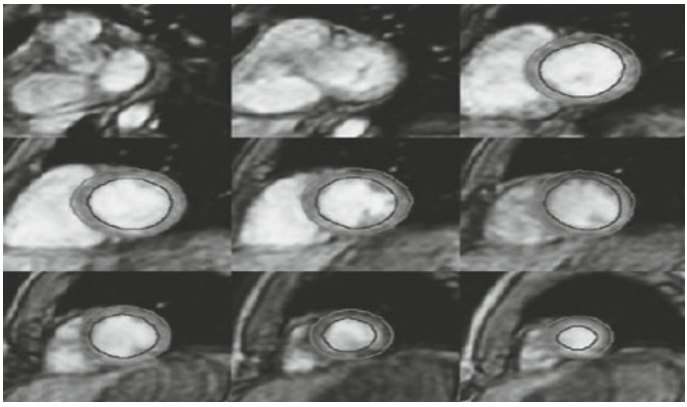


Fig. 2 Segmentation results by 3D AAM [7]

Usman and Rajpoot [10] proposed brain tumor type of multi-modality MRI. In these, the algorithm works in three steps. In the first step, preprocessing of images is taking place, in which, first they remove complete blank slices, create a mask and use it to discover bounding field after ground fact and use the box to crop multi-modality images. In the second step, feature extraction takes place, where feature extraction consists of four varieties of features: intensity, intensity difference, neighborhood information, and wavelet-based texture feature. Third step is supervised classification, a machine learning approach. Image classifies using the training data, which creates a model. Later test data evaluate the model on unseen data and measures the performance of the algorithm. Figure 3 shows the segmentation result of Usman and Rajpoot [10]. for T1, T2, T1C, FLAIR, ground truth, and the final result.

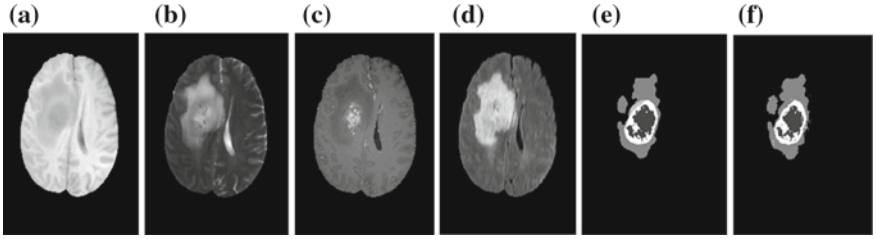


Fig. 3 Segmentation result using Usman and Rajpoot [10]. Each row represents a distinct subject. **a** T1, **b** T2, **c** T1C, **d** FLAIR, **e** ground truth, and **f** final segmented result

3 Performance Metric

3.1 True Positive Factor (TPF)

True Positive Factor (TPF) is also called as sensitivity. It can be measured as in equation with the help of Fig. 4 (1) [10, 15]:

$$True\ Positive\ Factore\,(P, T) = \frac{P_1 \cap T_1}{P_1 \cup T_1} \tag{1}$$

3.2 True Negative Factor (TNF)

True Negative Factor (TNF) is also called as specificity. It may be measured as in equation with the help of Fig. 4 (2) [10, 15]:

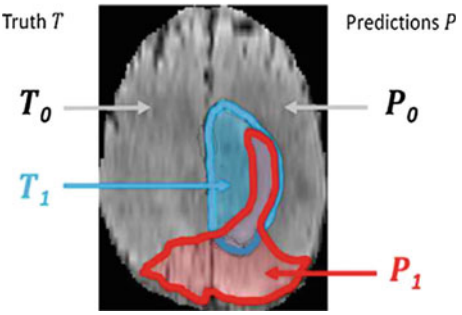


Fig. 4 T_1 is the ground truth, T_0 is the area outside T_1 within the brain. P_0 is the algorithm's predicated area outside P_1 within the brain, and P_1 is the algorithm's predicted lesion. The overlapped area between P_1 and T_1 gives us the true positive [10]

$$True\ Negative\ Factor\ (P,T) = \frac{P_1 \cap T_1}{T_1} \tag{2}$$

3.3 Sum of True Volume Factor (STVF)

Sum of True Volume Factor (STVF) is the sum of True Positive Factor and True Negative Factor and can be calculated as Eq. (3) [3]:

$$STVF = TPF + TNF \tag{3}$$

4 Comparative Analysis

In order to review different methods for image segmentation, we suggested a table, which summarized the work carried out by different researchers. Table 1 shows the review of the different methods.

Now we compare the result of Al Faris et al. [3] and Usman and Rajpoot [10] based on the parameters True Positive Faction (TPF), True Negative Faction (TNF), and Sum of True Volume Fraction (STVF), and it is shown in Table 2.

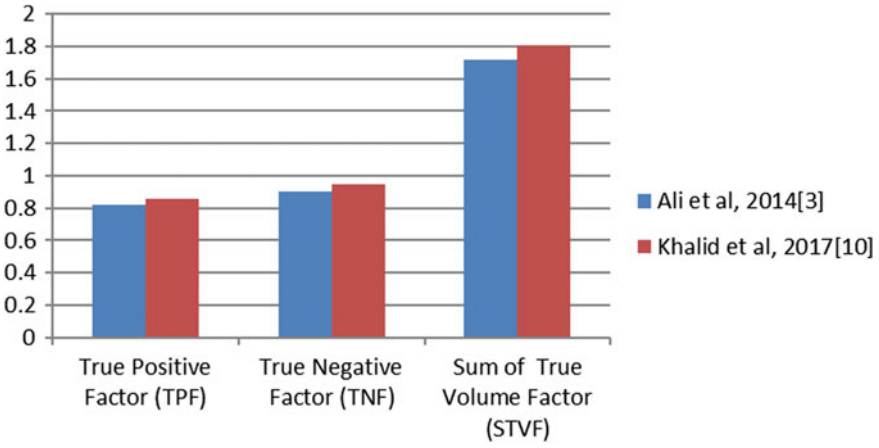


Table 1 Parametric review of different methods

Paper citation	Method	Dataset	Performance parameters	Research gap
Afifi et al. [2]	The seed point is selected by the position of highest amplitude in the histogram of the image and then applies SRG using calculated seed point	A brain dataset of McGill University (brain web) slice #72	Accuracy = 94%	Causes chaining effect especially for images with low contrast, shape boundaries or images with algorithm shift
Melouah and Layachi [1]	Computed the statistical features (mean, standard deviation, contrast, entropy, regularity, and uniformity) and declare it as seed point	MiniMIAS database of mammograms	Efficiency = 75%	The single seed point is selected in any case
Al Faris et al. [3]	The calculated density of the pixel intensity and highest density pixel declared as seed point	MRI breast dataset of RIDER	TPF = 0.82, TNF = 0.9, STVF = 1.73, RO = 0.75, MCR = 0.18	The algorithm is limited to the segmentation of the breast MRI tumor
Malek et al. [4]	They compute the average of local maxima and set the average as the initial seed point	Mammogram available on National Cancer Society of Malaysia	Accuracy = 0.94	Only one seed point is given in every case
Malek et al. [5]	Mathematical morphology is applied for evaluating seed point	Mammogram available on National Cancer Society of Malaysia	Accuracy = 0.98	Only one seed point is given in every case
Usman and Rajpoot [10]	There are three steps: preprocessing, feature extraction, classification	Brain images of multimodel brain tumor segmentation challenge (MICCAI BraTs 2013)	For complete (HG): Dice = 0.88, Jaccard = 0.79, Specificity = 0.86, Accuracy = 0.95	These are not a generalized algorithm

Table 2 Comparison between [3] and [10]

Author	TPF	TNF	STVF
Al Faris et al. [3]	0.82	0.90	1.72
Usman and Rajpoot [10]	0.86	0.95	1.81

5 Conclusion

In this review, the maximum common medical image segmentation algorithms have been discussed, and evaluation of different medical image segmentation was supplied. There are many image segmentation algorithms available which may be applied to medical images. Segmentation methods are necessary to be quick, efficient because medical image segmentation result is very sensitive, if the result is not accurate or process is lengthy, it will affect the diagnosis process that may be very harmful for patients. From Table 1, we can see Aminah Abdul et al. 2014 gives better accuracy than others, which is 98%. From Table 2, we clearly say that result of Usman and Rajpoot [10] gives better results in comparing to Al Faris et al. [3] on medical images which is shown in the graph. Many research gives an algorithm for automatic seed point selection, but their work is limited to find either fixed number of seed points. So in the future, focus can be applied in automatic seed point selection which gives better seed point.

References

1. A. Melouah, S. Layachi, A novel automatic seed placement approach for region growing segmentation in mammograms, in *IPAC'15,* pp. 23–25 Nov 2015 (ACM, Batna Algeria © 2015). ISBN 978-1-4503-3458-7/15/11. https://doi.org/10.1145/2816839.2816892
2. A. Afifi, S. Ghoniemy, E.A. Zanaty, S.F. EI-Zoghdy, New region growing based on thresholding technique applied to MRI data, I. J. Comput. Netw. Inf. Secur. 61–67 (2015), Published Online June 2015 in MECS (www.mecs-press.org/). https://doi.org/10.5815/ijcnis.2015.07.08
3. A.Q. Al Faris, U.K. Ngah, N.A.M. Isa, I.L. Shuaib, Computer-aided segmentation system for breast MRI tumour using modified automatic seeded region growing (BMRI) (MASRG). J. Digit Imaging **27**, 133–144 (2014). https://doi.org/10.1007/s10278-013-9640-5. Springer
4. A.A. Malek, W.E.Z.W.A. Rahman, A. Ibrahim, R. Mahmud, S.S. Yasiran, A.K. Jumaat, Region and boundary segmentation of microclassifications using seed-based region growing and mathematical morphology, in *International Conference on Mathematics Education Research 2010 (ICMER 2010),* pp. 634–639, www.sciencedirect.com. Elsevier
5. A.A. Malek, W.E.Z.W.A. Rahman, S.S. Yasiran, A.K. Jumaat, U.M.A. Jalil, Seed point selection for seed-based region growing in segmenting microclassifications, in *International Conference on Statistics in Science, Business and Engineering (ICSSBE) IEEE Conference* 2012
6. A. Eklund, P. Dufort, D. Forsberg, S.M. LaConte, Medical image processing on GPU—past, present and future. Med. Image Anal. **17**, 1073–1094 (2013)
7. C. Petitjean, J.-N. Dacher, A review of segmentation methods in short axis cardiac MR image. Med. Image Anal. **15**, 169–184 (2011)

8. E. Smistad, T.L. Falch, M. Bozorgi, A.C. Elster, F. Lindseth, Medical image segmentation on GPUs—a comprehensive review. Med. Image Anal. **20**, 1–18 (2015)
9. J. Liu, M. Li, J. Wang, F. Wu, T. Liu, Y. Pan, A survey of MRI-based brain tumor segmentation methods. Tsinghua Sci. Technol. **19**(6), 578–595, ISSN: 1007-0214 04/10, Dec 2014
10. K. Usman, K. Rajpoot, Brain tumor classification from multi-modality MRIs using wavelets and machine learning. Pattern Anal. and Appl. **20**, 871–881 (2017)
11. N. Shrivastava, J. Bharti, Empirical analysis of image segmentation techniques, in *SmartCom 2016* © Springer Nature Singapore Pte Ltd, CCIS 628 (2016), pp. 143–150. https://doi.org/10.1007/978-981-10-3433-6_18
12. R. Karim, P. Bhagirath, P. Claus, R.J. Housden, Z. Chen, Z. Karimaghaloo, H.M. Sohn, L.L. Rodriguez, S. Vera, X. Alba, A. Hennemuth, H.O. Peittgen, T. Arbel, M.A. Gonzalez Ballester, A.F. FRangi, M. Gotte, R. Razavi, T. Schaeffeter, K. Rhode, Evaluation of state-of-the-art segmentation algorithms for left ventricle infract from late Gadolinium enhancement MR images. Med. Image Anal. **30**, 95–107 (2016)
13. N.M. Zaitoun, M.J. Aqel, Survey on image segmentation techiniques. Proc. Comput. Sci. **65**, 797–806 (2015). Elsevier
14. V. Tavakoli, A.A. Amini, A survey of shaped-based registration and segmentation techniques for cardiac images. Comput. Vis. Image Underst. **117**, 966–989 (2013)
15. M. Polak, H. Zhang, M. Pi, An evaluation metric for image segmentation of multiple objects. Image Vis. Comput. **27**, 1223–1227 (2009). Elsevier

Stream and Online Clustering for Text Documents

Iti Sharma, Aaditya Jain and Harish Sharma

Abstract Spherical k-means is a popular version of k-means to cluster directional data, especially text documents. Each iteration of spherical k-means can be viewed as batch process because the centroids are updated only once per iteration. Performing updations in online manner gives better performance and a technique to deal with streaming text data too. Research works in this direction are reviewed through experiments in this paper. Further solutions to the discovered issues are suggested.

Keywords Spherical k-means · Text mining · Clustering · Text stream Dirichlet priors

1 Introduction

Exponential growth in people using Internet as a result of ease of access and economical computers and smartphones has led to generation of huge amounts of data. This data is mostly in form of text and is great source of information valuable for researchers and analysts. Knowledge discovery from text data requires variety of machine learning processes; clustering is a major part of these processes. Clustering of text data aims at labeling the documents with topics or categories. A cluster is a group of documents that belong to a similar concept. Clustering is useful for topic detection, categorization, and organization of documents.

I. Sharma
Career Point University, Kota, India
e-mail: itisharma.uce@gmail.com

A. Jain (✉)
R.N. Modi Engineering College, Kota, India
e-mail: jain.aaditya58@gmail.com

H. Sharma
Rajasthan Technical University, Kota, India
e-mail: harish.sharma0107@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_49

Clustering methods popularly used can be broadly seen as partitional or hierarchical. Generally, hierarchical clustering techniques do not scale well and are not recommended for huge data like text. Partitional methods like k-means have been used for clustering text data successfully. Characteristics of text documents like huge vocabulary giving high dimensional representations, high sparsity, and non-Gaussian distribution of values are unique enough to treat such data separately and devise clustering techniques specific to it.

Dhillon and Modha [1] proposed that k-means when applied to normalized text data can produce concept vectors that summarize the text data very close to best. The text data is normalized such that each document is a unit length vector, making the data space hyperspherical. This variant of k-means is called spherical k-means.

Clustering text data streams is used in number of applications such as newsgroup filtering, text crawling, document organization, and TDT (topic detection and tracing). In such applications, text data comes as a continuous stream and this presents many challenges to traditional static text clustering [2]. Hence, steam clustering techniques based on spherical k-means and others are an interesting field to review.

This paper presents discussions on methods proposed in literature to cluster streams of text data. Paper is organized as follows: Sect. 2 is about why spherical k-means may fail for streaming text or huge-sized corpus and why online variants are required; Sect. 3 discusses methods based on spherical k-means and their experimental evaluation; and Sect. 4 briefly discusses other techniques for clustering text streams.

2 Motivation

Though scalable and having a linear runtime, spherical k-means may not well handle corpus with large number of text documents as its memory requirement is high. It requires all data to be in memory for proper initialization. Besides this, the batch processing style implies centroids to be updated only once per iteration. This gives a slow convergence to the algorithm when initialized poorly. In case of corpus where all data cannot be stored in memory and has to be loaded from external storage, spherical k-means will not work. A variant that works for streaming data is required in these situations. Surprisingly, for both the situations, a single solution works, that is, update centroids accordingly. A poor initialization can be overcome if the centroids are updated more frequently. When processing data in chunks, updating centroids and generating more centroids or merging works better.

Dealing with streaming text as in newswire feeds, the number of clusters existing is not known a priori, while it is mandatory for spherical k-means. Hence, spherical k-means is not suitable for topic discovery in streaming text.

3 Partition-Based Methods

Spherical k-means (SKM) [1] is a gradient approach that partitions data into disjoint groups based on cosine similarity. The heuristic tries to maximize following objective function:

$$\mathcal{Q} = \sum_{j=1}^{k} \sum_{\mathbf{x}_i \in \pi_j} \mathbf{x}_i \mathbf{c}_j^T$$

where $\mathbf{x}_i$ are all document vectors and $\mathbf{c}_j$ are all concept vectors that are centroids of the clusters. The algorithm proceeds in iterative manner, and each iteration assigning cluster label to document vectors based on maximum similarity as

$$\pi_j = \left\{ \mathbf{x}_i \,|\, j = \arg\max_l \; \mathbf{x}_i \mathbf{c}_l^T \right\}, 1 \leq j \leq k$$

and then updating concept vectors as

$$\mathbf{c}_j = \frac{\sum_{\mathbf{x}_i \in \pi_j} \mathbf{x}_i}{\left\| \sum_{\mathbf{x}_i \in \pi_j} \mathbf{x}_i \right\|}$$

Zhong [3] suggested a Winner-Take-All approach to obtain an online variant instead of batch processing as in SKM. After each assignment of a document vector to a cluster, the concept is updated as

$$\mathbf{c}_j = \frac{\mathbf{c}_j + \eta \, \mathbf{x}}{\left\| \mathbf{c}_j + \eta \, \mathbf{x} \right\|}$$

where η is learning rate. The updates are never done at the end of assignment step. The algorithm suggested in [3] is called Online Spherical K-means (OSKM). The authors suggest that instead of using a flat constant learning rate, an exponentially decreasing rate should be used. The iteration variable is *i,* and learning rate at every iteration is decided as:

$$\eta_i = \eta_0 \left(\frac{\eta_f}{\eta_0} \right)^{\frac{i}{NM}}$$

where N is total number of document vectors and M is number of iterations. Such gradual decrease produces an annealing effect.

A straightforward adoption of OSKM may not give desired results when practically applied. The reason is that OSKM can be greatly affected by the order of inputs. There are two options to mitigate the bad effects: Use a learning rate schedule that guarantees equal contribution of each input; or randomize the order of inputs at every iteration.

Practical considerations for implementation of OSKM are as follows: (i) To avoid computation bottleneck of normalizing the centroids after every update, the step is deferred until the magnitude of the vector becomes too large; (ii) Empty clusters are handled by assigning the points that are farthest from their centroids to new (empty) clusters; (iii) Speed-up can be achieved by running algorithm for few iterations with samples instead of entire data, and later run it over whole data.

Scalable and stream clustering based on SKM is proposed in [4]. Instead of processing documents in the stream continuously, segments are created. Size of segment is decided based on available memory buffer and application. At any instance, the centroids of all past documents are retained as K history vectors. The S vectors of current segment and K history vectors are clustered through OSKM. The vectors are weighted according to the rule: Each new data vector has weight 1 and k-th history vector has weight $\gamma(N_k + W_k)$, N_k is number of data vectors assigned to k-th cluster and W_k is sum of weights of history vectors assigned to k-th cluster. The decay factor γ has a value between 0 and 1.

Aggarwal and Yu [5] have proposed a condensation-based stream clustering method and named it ConStream. It presents condensed summary of a data stream and is specifically for streaming text like newsfeed where clusters are formed of vectors arriving in same time interval. The idea is to maintain a statistical summary of the data stream seen so far in the form of cluster droplets, instead of centroids. A time-sensitive weightage is given to data points so that inactive clusters can be removed and new clusters can be formed. The weight is through a fading function $f(t) = 2^{-\lambda t}$. A new data point that is used to start a new cluster is not rejected that waits for other points to join its cluster until its half-life. Half-life is the time instant t' when $f(t') = \left(\frac{1}{2}\right) f(0)$. Similarly, any cluster dies when weighted number of points in the cluster is 0.5. A cluster droplet $\mathcal{D}(t, \mathcal{C})$ is a tuple $\left(\overline{DF2}, \overline{DF1}, n, w(t), l\right)$, where vector $\overline{DF2}$ contains those pairs of words that co-occur in at least one document of the cluster. Vector $\overline{DF1}$ the sum of the weighted counts for each word occurring in the cluster. The n, $w(t)$, l are number of points in the cluster, sum of weights of points at time t and time stamp when last point was added to the cluster, respectively. These droplets have additive and decay properties such that the updates to the droplets can be computed very efficiently. Moreover, if two clusters are merged, the droplets can be merged. The algorithm begins with first k points as k different clusters. For each arriving new point $\bar{X}$ its cosine similarity with $\overline{DF1}$ of every droplet, if the maximum similarity with any cluster is larger than a set threshold, the point is assigned to this cluster. Else, a new cluster is created. The least recently updated cluster is removed. Though this process is very effective in producing summaries of the text stream, it cannot be adapted for massive data. There is no guarantee that a cluster removed at an instant of time may never be active in future. The algorithm will produce too many clusters for huge corpus when operating on it in chunks. A merging process is the required for the summaries.

Khalilan et al. [6] have suggested clustering using average silhouette analysis of clusters being produced. Documents arriving within a time window are analyzed. Based on silhouette values, they are either treated as one concept, or divided into

two subsets, or are partitioned into few micro-clusters using k-means or the stream may be ignored and each data point is clustered independently. At any instant, if any micro-cluster grows very large, it is split into more. If two concepts are too close, the micro-clusters are merged. Only issue here is to decide the number of subsets into which a window is partitioned. It will highly affect the performance.

4 Evaluation and Comparison of SKM Based Methods

In order to compare performances of the methods based on partitional clustering, we performed experiments over conference data of Neural Information Processing Systems (NIPS) available at UCI machine repository in processed document-term matrix form and Topic detection and tracking (TDT2) corpus of 20 categories available preprocessed at http://cad.zju.edu.cn. The NIPS dataset is a part of bag-of-words dataset without any predefined number or labeling of categories. The documents are input as arriving in random order. While no ground truth is known in NIPS, we compare performance through the value of objective function. For TDT2, ground truth is known; hence, adjusted random index is used for comparison. Table 1 lists the issues discovered during experiments with the algorithms and the suggested changes. Table 2 lists the average values in results. The offline SKM is implemented for a baseline comparison with random initialization among the data, SKMO with random initialization and $\eta_0 = 0.1$ and $\eta_f = 1$. We have initialized OSKM-s with first k arriving documents and set $\gamma = 0.1$. Different sizes of segments produce different results. If size of segment is too large, it is more close to offline clustering. For ConStream, we have not removed any micro-cluster at any instant, rather saved it in memory and

Table 1 Issues in partition-based clustering methods for text streams and suggested solutions

Method	Issues	Suggested solutions
SKMOs [4]	Initial centroids are random values, totally filled vectors	First k vectors as seeds so that sparsity is maintained for longer
	Size of segment affects performance	Open problem
ConStream [5]	Removal of micro-clusters as outliers before complete stream is processed	Add an offline phase that tests all removed micro-clusters to be merged with others or treated as outlier
	Value of k—the number of current clusters—decides memory complexity and trend detection	Open problem
	Threshold value may be very less when a new cluster begins and the initial points are very close. If the cluster is inherently larger, this will fragment the cluster	Add an offline phase that merges two clusters after testing with the current value of threshold

Table 2 Results of experimental evaluation for comparison

Method		ARI for TDT2	Objective function value for NIPS
SKM offline [1]		0.59 ± 0.032	761 ± 3.7
SKMO [3]		0.52 ± 0.081	768 ± 7.4
SKMOs [4]	S = N/5	0.44 ± 0.091	517 ± 4.8
	S = N/10	0.41 ± 0.035	445 ± 3.4
	S = N/20	0.37 ± 0.014	438 ± 2.9
ConStream [5]	Outliers removed	0.56 ± 0.067	419 ± 4.6
	No outliers	0.48 ± 0.025	521 ± 3.8

then an offline phase after the stream is over is introduced that merges the clusters or marks them as outliers according to their current values of threshold.

It can be observed that ConStream produces ARI value most close to that of offline SKM. Besides ConStream, SKMO also achieves ARI higher than 0.5 that can be considered acceptable. SKMO produces objective function value higher than that of offline SKM, as expected. But none of the stream clustering methods achieve such high values. In stream clustering, we do not expect very high objective function value. The ConStream method is winner here too having highest objective value among the stream clustering algorithms.

5 Other Methods for Clustering

Zhang et al. [7] proposed an online clustering of text documents using Dirichlet process mixture model. Every cluster is modeled according to a multinomial distribution whose parameter follows a Dirichlet prior. For every arriving point, the cluster to join or to open a new cluster is decided through probabilities computed using Dirichlet process. Whenever a point joins an existing cluster, the model is updated using Bayes rule.

Liu et al. [8] have proposed an improved smoothing semantic model and used it to cluster text streams. The cluster structure of the stream is captured as cluster profile which is much similar to cluster droplets of [5] but the contents of the cluster profile include both independent word counts and phrase counts.

Other topic models, viz., Latent Dirichlet Allocation, Dirichlet Compound Multinomial mixture, and von Mises–Fisher mixture model, are implemented in an online variant for text stream clustering by Banerjee and Basu [9]. They concluded that vMF is better than other two for cluster discovery. Further, a hybrid topic model is proposed in [9] that uses both online and offline phases for efficient clustering.

A dynamic clustering topic (DCT) model based on Dirichlet Multinomial Mixture with collapsed Gibbs sampling is suggested by Liang et al. [10] to handle temporal characteristics of text streams. Short text is handled by assigning single topic to it.

Already seen data (history) is used to derive priors, and Bayesian rules are applied to include changes in the learned distribution as new documents arrive.

6 Conclusion

Text steam clustering has several important applications in information retrieval. Though topic-based semantic models are more effective in capturing the structure of documents, the bag-of-words model still maintains attraction. Spherical k-means-based online clustering for streams should therefore be thoroughly investigated. We have presented a brief overview and empirical evaluation of such methods. Method of [4] is fastest and easiest, but value of k is to be set a priori. Both ConStream [5] and DCStream [6] techniques are effective but require rigorous experiments to set appropriate values of parameters. The proposed solutions to the issues of these algorithms are good for practical usage.

Challenges of k-means-based techniques are deciding the number of clusters in streams. Maintenance of history, decay factors, and fading functions are aspects of research. Finding effective metrics that can be used to decide the hierarchy of discovered clusters in the stream can be a good research direction.

References

1. I.S. Dhillon, D.S. Modha, Concept decompositions for large sparse text data using clustering. Mach. Learn. **42**, 143–175 (2001)
2. C.C. Aggarwal, A framework for diagnosing changes in evolving data streams, in *Proceedings ACM SIGMOD* (2003), pp. 575–586
3. S. Zhong, Efficient online spherical k-means clustering, in *IEEE International Joint Conference of Neural Networks* (2005)
4. S. Zhong, Efficient Streaming text clustering. Neural Netw. **18**(5–6), 790–798 (2005)
5. C.C. Aggarwal, P.S. Yu, A framework for clustering massive text and categorical data streams. in *Proceedings of the SIAM Conference on Data Mining* (2006), pp. 477–481
6. M. Khalilian, N. Mustapha, N. Sulaiman, Data stream clustering by divide-and-conquer approach based on vector model. J. Big Data **3**, 1 (2016)
7. J. Zhang, Z. Ghahramani, Y. Yang, A probabilistic model for online document clustering with application to novelty detection, in *Neural Information Processing Systems: TDT2 Datasets* (2004)
8. Y. Liu, J. Cai, J. Yin, A.W. Fu, Clustering masisve text data streams by semantic smoothing model, in *ADMA 2007, LNAI 4632R*, ed. by R. Alhajj et al. (2007), pp. 389–400
9. A. Banerjee, S. Basu, Topic models over text streams: a study of batch and online unsupervised learning, in *ICML Conference* (2007)
10. S. Liang, E. Yilmaz, E. Kanoulas, *Dynamic Clustering of Streaming Short Documents*. KDD'16, San Francisco, CA, USA (2016). https://doi.org/10.1145/2939672.2939748

Design and Analysis of Low-Power PLL for Digital Applications

Ashish Tiwari and Renu Prabha Sahu

Abstract This paper discusses the challenges and outcomes in designing the low-power PLL for digital applications. PLL being an important block for providing clocking scheme in many electronic circuits raises the requirement of decreasing the power with the growing CMOS technology. The state-of-the-art designs adopted are critically reviewed, and improvements are suggested.

Keywords PLL · CMOS · Clocking · Power

1 Introduction

The PLL is the predominant and constructing a part of digital electronics, communication (wireless and wire-line) and excessive-speed (low propagation delay) digital methods. A PLL designed by means of built-in CMOS has executed the fine importance within the last few many years considering the fact that of the high-performance system design within the digital and communication areas. It is in actual fact utilized in many methods for frequency synthesis, clock/data restoration, clock de-skewing, and many others. For an IC, the important element to be regarded for its designing is its low power consumption (because of smaller chip dimension) and increased operating speed. Right here the enhancement of speed is not the major obstacle; here, the focus is more commonly on reduction in power consumption via PLL as a lot as viable. There are basic components present inside PLL whose circuitry might be modified to obtain the desired outcomes, without effecting different fundamental phenomena.

A. Tiwari · R. P. Sahu (✉)
Department of Electronics and Telecommunication, FET-SSGI, Bhilai, India
e-mail: rpsahu0503@gmail.com

A. Tiwari
e-mail: tiwari.ashish99@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_50

The rest of the paper is organized as follows: A brief description of the PLL is described in Sect. 2. Section 3 contains the architecture adopted for low-power PLL design, while conclusions are drawn in Sect. 4.

2 General Description of PLL

A PLL can be seen as a circuit which synchronizes an output signal (generated by an oscillator) with a reference or provided input signal in phase as well as frequency [1]. As described in Fig. 1, PLL constitutes phase frequency detector (PFD), charge pump, loop filter and voltage-controlled oscillator (VCO); these are the basic building blocks of the PLL.

Phase frequency detector (PFD). It is basically used to compare the phase of feedback signal from VCO with the phase of input or reference signal and generate outputs (UP or DOWN) as per phase difference also known as phase errors.

Charge pumps (CPs). It is mainly used for conversion of the digital output of PFD into the current signal, such that a stable controllable signal has been generated for oscillator to control the oscillation frequency.

Loop filter. The loop filter is the heart of PLL. It is used to stabilize the system and achieve the desired response.

3 Previous Work on Low-Power PLL

Gursoy et al. [1] presented a work in which they had designed, verified, system integrated and physically realized a high-speed monolithic PLL-based high-performance clock and data recovery (CDR) circuit as described in Fig. 2. Overall power consumed is 18.6 mW, and area of CDR is approximately 0.3 mm^2 [1].

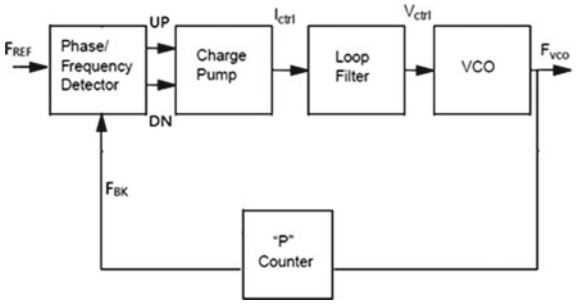


Fig. 1 Basic building block of PLL

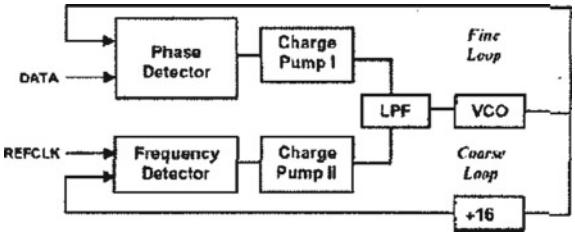


Fig. 2 Architecture proposed by Z. O. Gursoy et al. of two loop CDR blocks

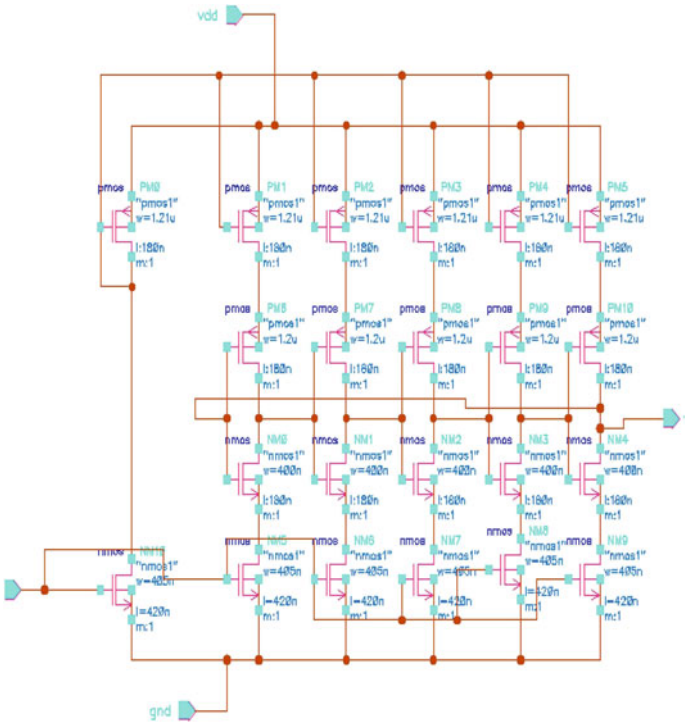


Fig. 3 Architecture proposed by Ashish Mishra et al. of 5-stage CS-VCO

Mishra et al. [2] analyzed by using 5-stage current-starved VCO (CS-VCO) as shown in Fig. 3 with large VCO gain and less lock time with oscillation frequency range of 431.683 MHz–1.7966 GHz. PLL power consumed is 7.08 mW with improved phase noise performance for the 5-stage VCO [2].

Moorthi and Aditya [3] focused on designing a 1 GHz range PLL with low power consumption of 0.34 mW. A telescopic (as shown in Fig. 4) OTA-based ICO (current-controlled oscillator) is designed. The system is simulated in CADENCE UMC180 nm technology [3].

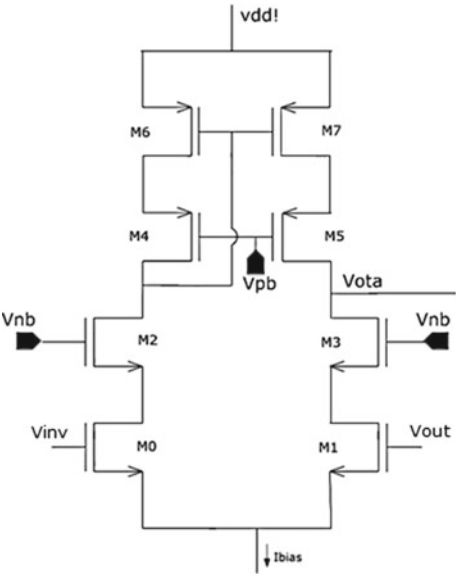


Fig. 4 Architecture proposed by S. Moorthy et al. of telescopic OTA

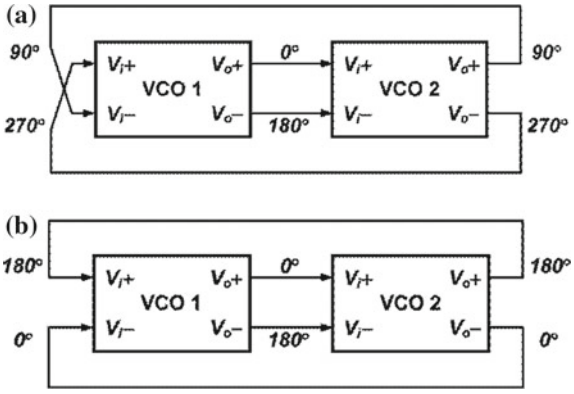


Fig. 5 Architecture proposed by Chung-Ting Lu et al. **a** Antiphase **b** inphase coupling structures

Lu [4] demonstrated a circuit topology based on the 0.018-pm CMOS of the quadrature voltage-controlled oscillator (QVCO). Quadrature output phases could be generated at minimum power consumption, with the antiphase coupling (as shown in Fig. 5). PLL has frequency range of 2.4 GHz consumed a DC power of 14.4 mW from a 0.6 V supply [4].

Agrawal and Khatri [5] performed and designed the PLL in which PFD has designed to make PLL became free in dead zone; VCO (as described in Fig. 6) is giving larger tuning range with consumed power of 277.2 μW and supply of 1.8 V.

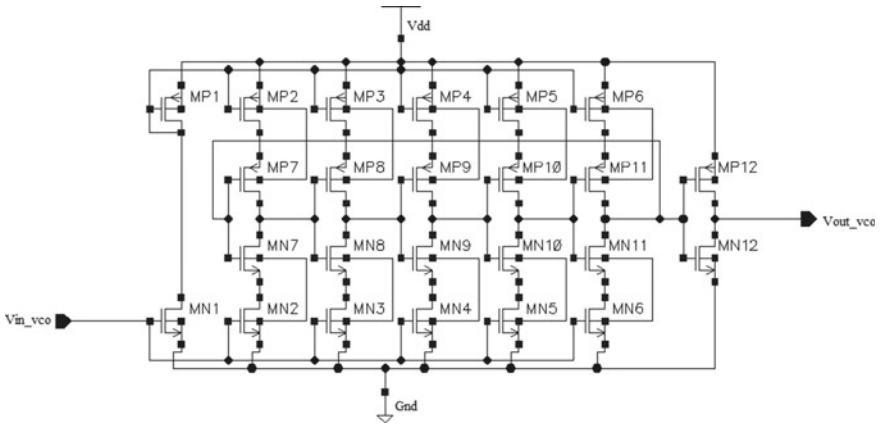


Fig. 6 Architecture proposed by Anshul Agrawal et al. of CS-VCO

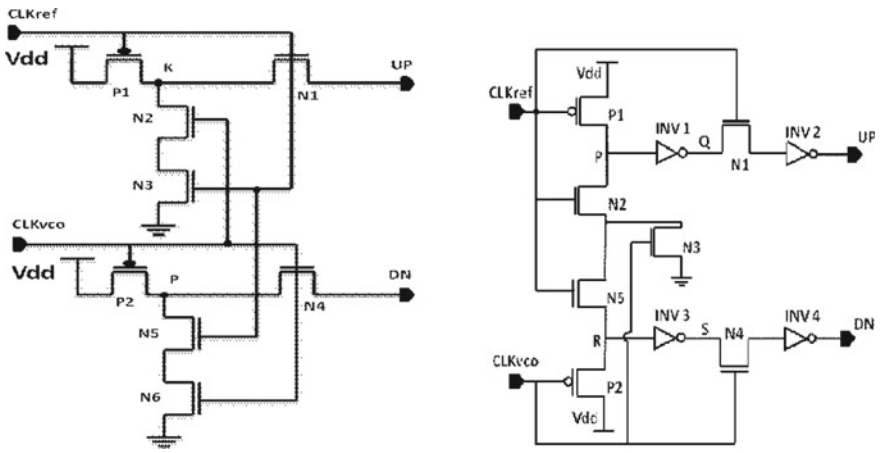


Fig. 7 Architecture proposed by Abdul Majeed K. K et al. of PFD1 and PFD2

Majeed and Kailath [6] presented two phase frequency detectors PFD1 and PFD2 by using 15 and 8 transistors, respectively, as described in Fig. 7. With a supply voltage of 1.8 V, power consumption has reduced by 80.3 and 99.2%. A 180-nm CMOS technology in CADENCE virtuoso environment had used [6].

Kaipu [7] designed third-order, fully integrated low-power PLL with some modifications in VCO as described in Fig. 8. The overall power consumed here is 274.34 μW at 1.4/1.8 V [7].

Garg and Sulochana Verma [8] presented third-order PLL; they have tried to reduce power consumption to 37% with minimum 12 mW at 350-nm technology node by using SPICE model.

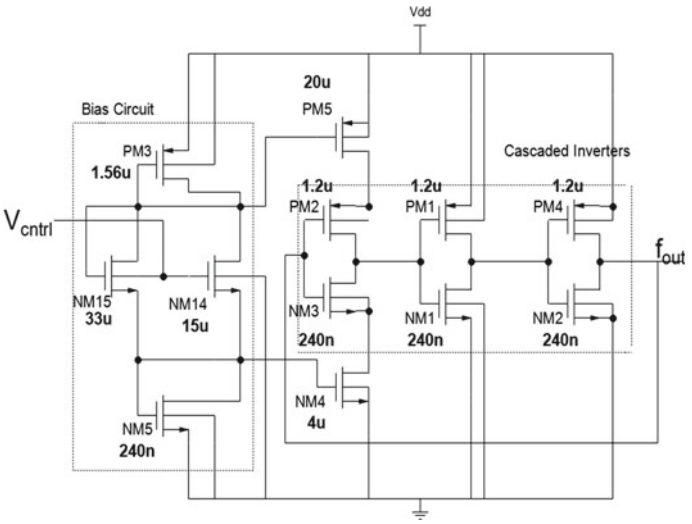


Fig. 8 Architecture proposed by S. V. R Kaipu et al. of VCO along with transistor widths

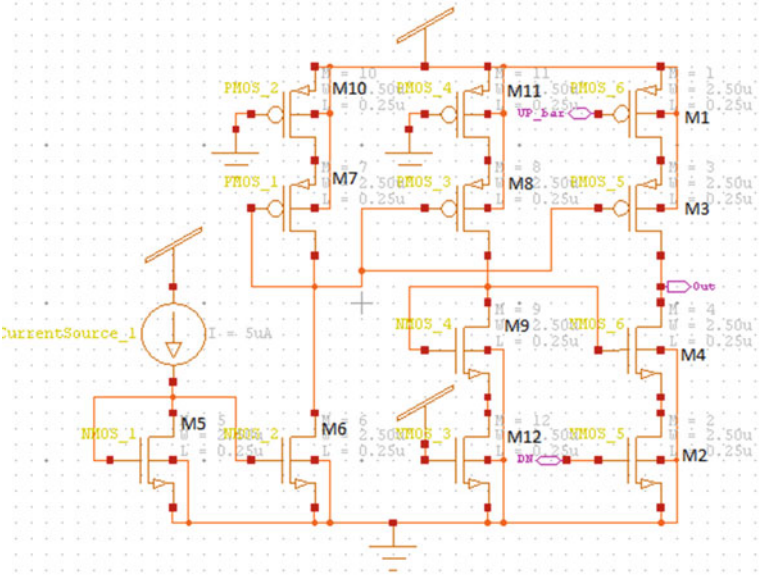


Fig. 9 Architecture proposed by Swati Kasht et al. for charge pump

Kasht et al. [9] designed low-power PLL and fast locking using 180-nm CMOS technology. To achieve this, charge pump has used current mirrored structure to decrease the current mismatch with enhanced output voltage as described in Fig. 9. And it results with consumed power as 0.38 mW [9].

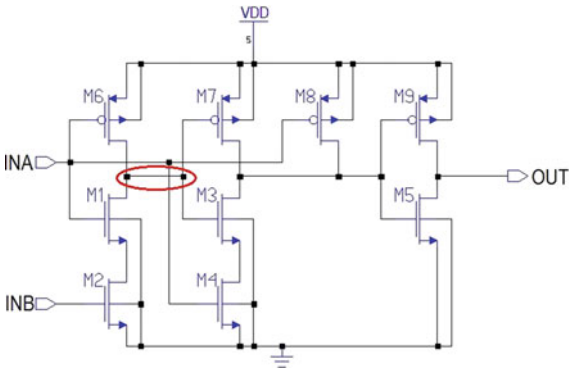


Fig. 10 Architecture proposed by K. N. Minhad et al.

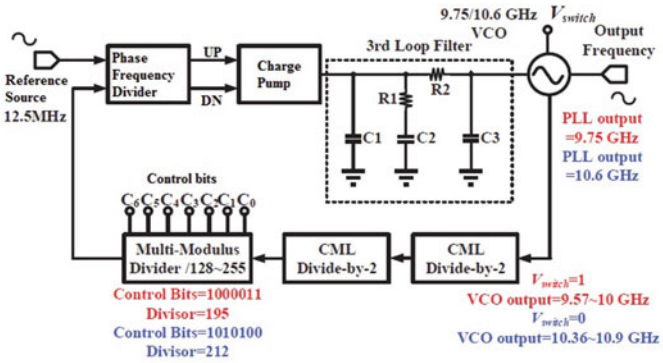


Fig. 11 Architecture proposed by Jeng-Han Tsai et al.

Minhad et al. [10] focused on power supply, power consumption, layout area, dead zone size and wide input frequency range. The total power consumption is 59 pW as described in Fig. 10 [10].

Tsai et al. [11] designed and fabricated an X-band 9.75/10.6-GHz fully integrated low-power PLL on 180-nm CMOS process. Nearly 24 mW power has consumed, with output frequency of 9.75 GHz as described in Fig. 11 [11].

Huang et al. [12] designed a 5.5-GHz fully integrated low-power PLL on standard 180-nm CMOS process. Here 9.23 mW power consumption has achieved by the utilization of transformer feedback VCO [12].

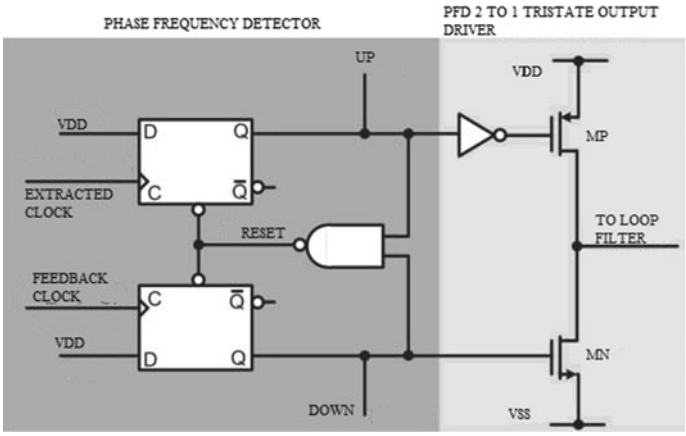


Fig. 12 Proposed PFD and tristate output driver

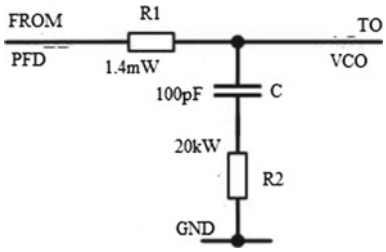


Fig. 13 Loop filter configuration

4 Proposed Method and Simulation Result

4.1 Phase Frequency Detector and Loop Filter

The proposed type of phase frequency detector (PFD) [13] is also known as "sequential phase detector" (as shown in Fig. 12). PFD is divided into two parts. In the first part, the positive edges of the input signals that extracted from clock and feedback clock, respectively, has been detected by two registers; the output of the registers goes high if an edge has been detected. On the other hand, if feedback clock leads, then the DOWN signal will become high to decrease the input voltage of loop filters and so on the output frequency.

Figure 13 shows loop filter principle. For slow variations, loop filter acts like integrator averaging the outputs of the PFD.

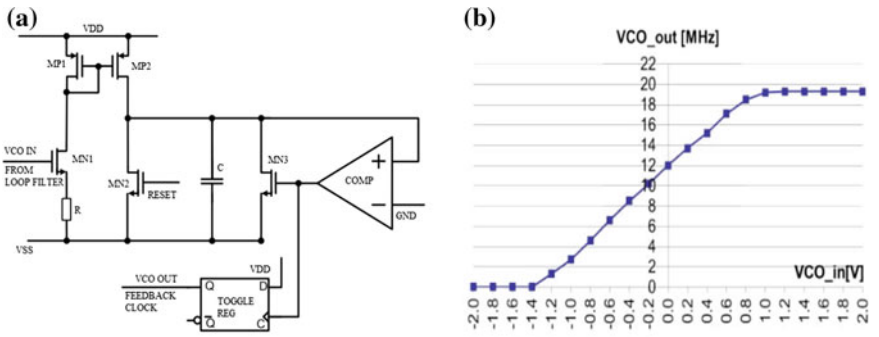


Fig. 14 **a** Proposed VCO configuration. **b** VCO output frequency w.r.t input control voltage

4.2 Voltage-Controlled Oscillator

The output and feedback clock of PLL are generated by VCO as shown in Fig. 14a. The loop filter output is connected with NMOS MN1 transistor, and a resistor R with high ohmic value is connected at the source to linearize the current. This current is mirrored with MP1 and MP2 PMOS transistors which are used to charge the capacitor C from VSS to GND. It has been monitored with comparator. The comparator output goes high and opens the switch MN3 to quickly discharge C, when the voltage reaches GND.

The VCO characteristic over the operating range is shown in Fig. 14b. The output frequency can vary linearly between 0 Hz at −1.4 V and around 19 MHz at +1 V. The proposed PLL works on two modes. The first mode acts as basic PLL in which the output follows the input data. In the second mode, the PLL is disabled by pulling the UP and DOWN output signals of the PFD to VSS. The proposed system has been verified by simulations and fabricated in a 90-nm high-voltage CMOS process technology.

The total power consumption in this approach is 800.64 μW.

5 Conclusion

The low-power PLL design circuits discussed in this paper mainly describe the modification in the design of VCO and PFD. We have additionally included the design of new loop filter along with modified design of VCO and PFD. The proposed design is more efficient in extracting the high frequencies with lower power consumption. The total power consumption obtained using the 90-nm CMOS technology is 800.64 μW.

References

1. Z.O. Gursoy, Design and realization of a 2.4 Gbps–3.2 Gbps clock and data recovery circuit using deep-submicron digital CMOS technology, in *2003 Proceedings IEEE International SOC Conference [Systems-on-Chip]*
2. A. Mishra, G.K. Sharma, D. Boolchandani, Performance analysis of power optimal PLL design using five-stage CS-VCO in 180 nm, in *International Conference on Signal Propagation and Computer Technology (ICSPCT)* (IEEE, 2014)
3. S. Moorthi, S. Aditya, A low jitter wide tuning range phase locked loop with low power consumption in 180 nm CMOS technology, in *2013 IEEE Asia Pacific Conference on Postgraduate Research in Microelectronics and Electronics (PrimeAsia)* (2014)
4. C.-T. Lu, A low-power quadrature VCO and its application to a 0.6-V 2.4-GHz PLL. IEEE Trans. Circuits Syst. I: Regul. Pap. **57**(4) (2010)
5. A. Agrawal, R. Khatri, Design of low power, high gain PLL using CS-VCO on 180 nm technology. Int. J. Comput. Appl. (0975—8887) **122**(18) (2015)
6. K.K.A. Majeed, B.J. Kailath, Low power, high frequency, free dead zone PFD for a PLL design (IEEE, 2013)
7. S.V.R. Kaipu, K. Vaish, S. Komatireddy, A. Sood, M. Goswami, Design of a low power wide range phase locked loop using 180 nm CMOS technology 978-1-5090-2684-5/16/$31.00 ©2016 IEEE
8. K. Garg, V. Sulochana Verma, Low power design analysis of PLL components in submicron technology, in *Advances in Computing & Information Technology, AISC*, vol. 178 (Springer, Berlin, Heidelberg, 2013), pp. 687–696
9. S. Kasht, S. Jaiswal, D. Jain, K. Verma, A. Somani, Designing of charge pump for fast-locking and low-power PLL Int. J. Comput. Technol. Electr. Eng. (IJCTEE) **2**(6) (2012)
10. K.N. Minhad, M.B.I. Reaz, J. Jalil, A low power 0.18-μm CMOS phase frequency detector for high speed PLL. ELEKTRONIKA IR ELEKTROTECHNIKA, vol. 20, no. 9, ISSN 1392-1215 (2014)
11. J.-H. Tsai, C.-Y. Hsu, C.-H. Chao, An x-band 9.75/10.6 GHz low-power phase-locked loop using 0.18-μm CMOS technology, in *Proceedings of the 10th European Microwave Integrated Circuits Conference*, Paris, France, 7–8 Sept 2015
12. S.-W. Huang, J.-H. Tsai, J.-P. Chou, A 5.5 GHz low-power PLL using 0.18-μm CMOS technology, 978-1-4799-2181-2/14/$31.00 © 2014 IEEE
13. R. Baker, CMOS: Circuit Design, Layout, and Simulation, vol. 18 (Wiley-IEEE Press, 2011)

Sentiment Analysis of Social Media Data Using Bayesian Regularization ANN (BRANN) Architecture

Neha Sahu and Anwar Sakreja

Abstract Of late, big data and big data analytics have fund applications in diverse fields. Social media and allied applications are one such domain for research, where artificial intelligence has shown unprecedented impact. In this paper, a mechanism has been proposed which can classify text data into classes of different sentiments. Data in the form of tweets have been used in this case. Pre-processing of raw data has been done prior to using it to train a neural network. A neural network is then trained using the categories of the data which are tweets that correspond to happy, neutral and sad moods of the Twitter users. The Bayesian regularization (BR) algorithm has been used for training the artificial neural network. It has been observed that this proposed technique achieves an accuracy of 98%. The mean square error is a mere 2% (approx).

Keywords Artificial neural network (ANN) · Text mining
Bayesian regularization · Mean square error (MSE)

1 Introduction

The advent of data analytics has been enormous, and text mining and opinion mining has garnered huge importance because of its broad range of applications in a variety of domains like the social media, analytics of data, business applications, etc. Sentiment analysis can be defined as a study that is based on a computational analysis and determination of textual opinions, emotions, behaviour and the attitude exhibited towards any entity [1, 2]. Sentiment analysis tries to find out the attitude or an opinion

N. Sahu (✉) · A. Sakreja
Department of CSE, VITM, Indore, India
e-mail: nehasahu.developer005@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_51

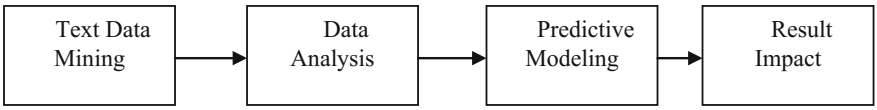


Fig. 1 Basic text mining and predictive modelling approach

of the user-based user's textual data. It also aids in making decisions. Sentiment analysis helps in determining whether the piece of tweet or any piece of writing is positive, negative or neutral [3]. It analyses the sentiment behind the text of any user; hence, it helps companies for product reviews and enhances business prospects. It has got a broad range of applications today, especially in the areas where outcomes are dependent on human sentiments and opinions. It can also be considered as opinion mining. To be able to analyse and implement such tasks, artificial intelligence is used. In this context, the concept of data mining is utilized which a knowledge-based procedure is based on extraction of skilled patterns and information. The extracted data are then used in visualization of applications and creation of real-time programs for the process of decision-making [4]. The applications can be diverse such as marketing and finance, advertising, opinion polls, social media, product reviews just to name a few. The following diagram illustrates the mechanism (Fig. 1).

2 Mathematical Modelling of Deigned ANN Structure

Out of the different approaches of [5]:

$$W_{i+1} = Wi - \left(J_K J_K^T - \mu I\right) e_{lj} J_k^T. \tag{1}$$

Here,

J_K represents the Jacobian matrix.
J_K^T stands for the transpose of the Jacobian matrix.
μ stands for step size of input data vector.
I represents an identity matrix.
J_K mathematically represents the second-order derivative of error (e) with respect to weight (w).

$$\partial^2 e / \partial w^2 = J_K \tag{2}$$

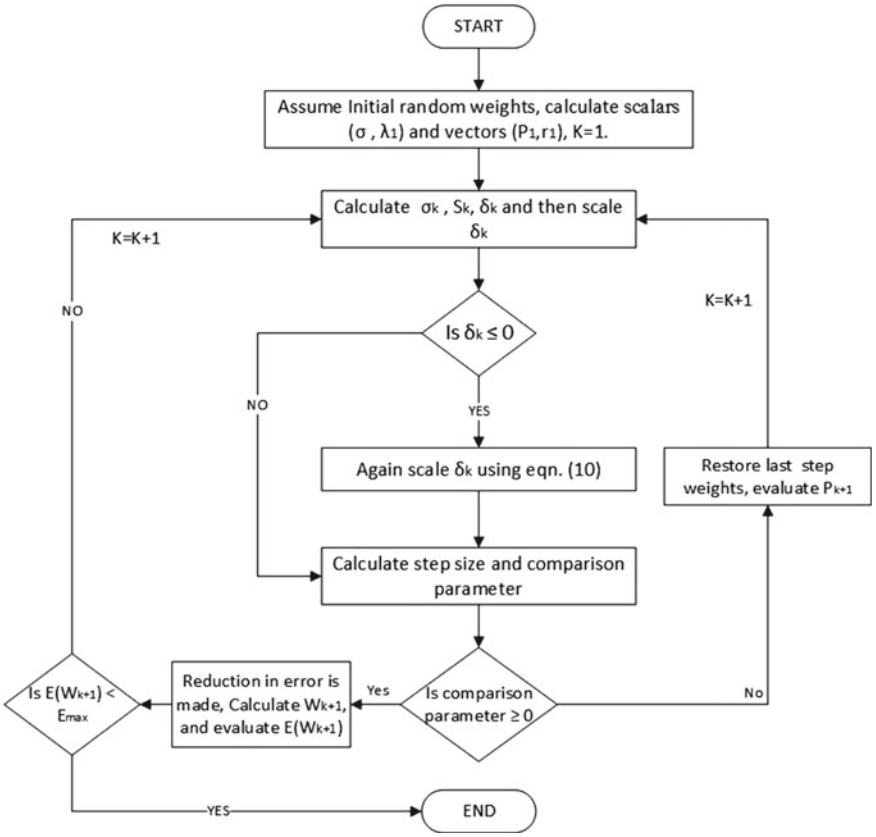


Fig. 2 Flowchart of BR algorithm

The classification mechanism though resembles the Bayes' theorem of conditional probability given by (Fig. 2):

$$P(A/B) = [P(B/A) \cdot P(A)]/P(B) \tag{3}$$

Here,

P(A/B) stands for the probability A given event B is true.
P(B/A) stands for the probability of B given event A is true.
P(A) represents the individual probability A.
P(B) represents the individual probability B.

3 Proposed System

The algorithm for the proposed system is given below:

(1) Text Mining: In this step, a total of 998 tweets from Twitter across different ethnicities have been taken.
(2) Pre-processing of raw data: In this step, repeated words, special characters and emoticons have been removed since such categories of data in unintelligible for an ANN [6].
(3) Characterization of data into positive, negative and neutral tokens.
(4) Division of data into training and texting samples. Following a general convention, 70% of the data have been used for training, whereas 30% of the data have been used for testing.
(5) Compute cumulative token sum of each tweet prior to training the designed ANN. The cumulative sum is given by [7]:

$$\sum t_p + t_n + t_{neu} = C_S \tag{4}$$

Here,

t_p, t_n and t_{neu} represent positive, negative and neutral tokens, respectively, and C_S represents the cumulative sum.

(6) Design an ANN and train it using the BR algorithm's yielding the performance function given by:

$$F(\omega) = A \cdot e_W + B \cdot E_D \tag{5}$$

where

$$E_D = \sum e_k^2 \tag{6}$$

Here, α represent objective functional parameters.

The final step is the evaluation of performance metrics given by the mean square error expressed as under:

$$\frac{\sum_{i=1}^{i=N} e_i^2}{N} = M \tag{7}$$

And

$$e = y_p - y_a \tag{8}$$

Here,

y_p indicates the predicted output and e denotes the error in prediction.

And

y_a denote actual output.

4 Results

The results contain the performance metrics of the designed BRANN architecture. The number of neurons in the hidden layer has been taken as 10 (Fig. 3).

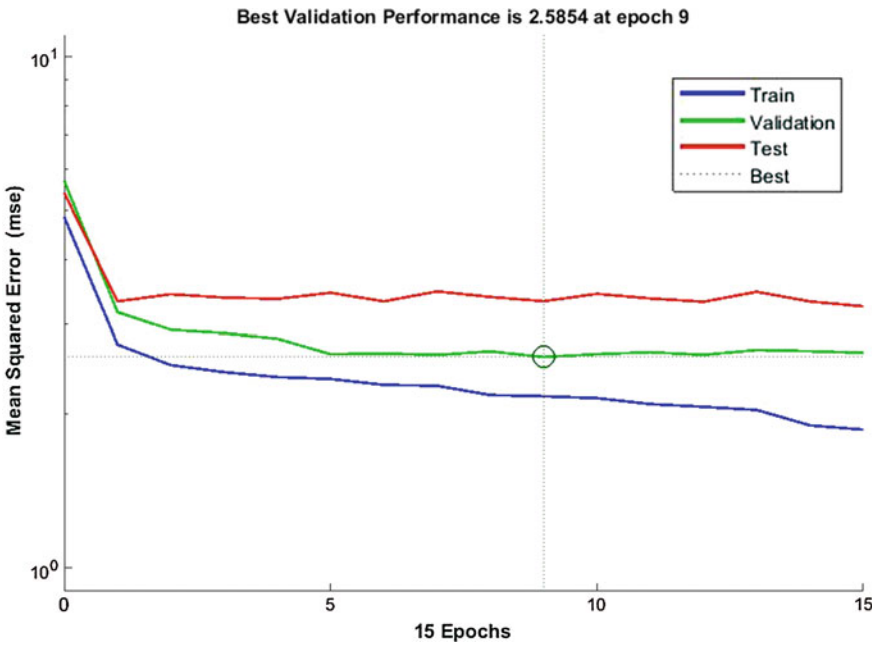


Fig. 3 Training states of the proposed algorithm

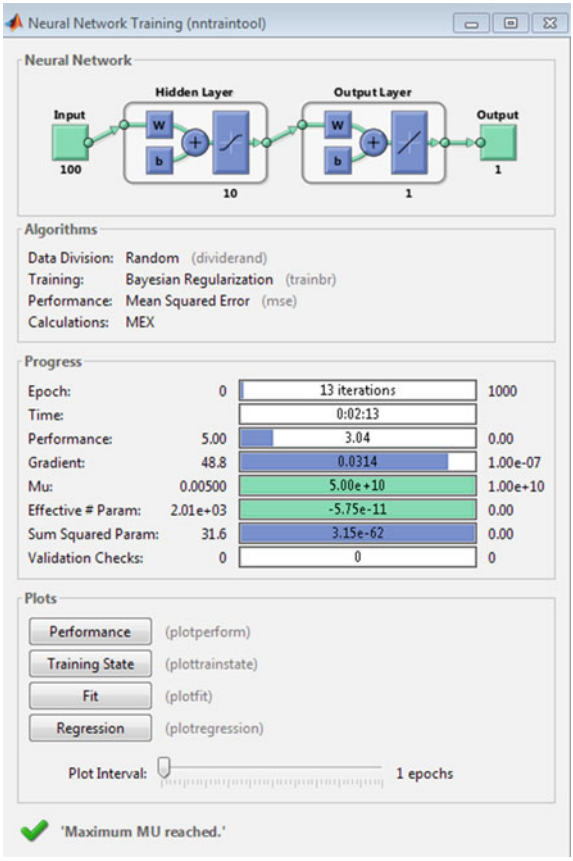


Fig. 4 Performance metrics of designed ANN algorithm

The training states illustrate the variation of step size (μ) and gradient (g) with the number of epochs. It is clearly observed that the error falls by a maximum step size (μ), the gradient exhibits a reduced rate of change that justifies the theoretical context of the back propagation methodology as discussed earlier.

From Figs. 4 and 5, it can be observed that the mean square error comes down to 3.04 after 13 iterations. After 8 epochs, the mean square error in the validation becomes nearly and almost constant. Moreover the training stops are 6 validations fails to find any fluctuation of the error in prediction [8]. This signifies that the proposed algorithm shows a fast convergence.

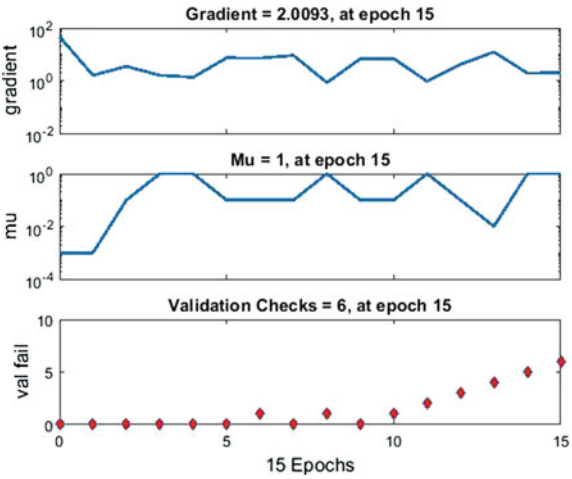


Fig. 5 Variation of training parameters with respect to training epochs

5 Conclusion

It can be concluded based on the mathematical background and the obtained results that the proposed system proves to a highly accurate model high for text mining of data. Here, the tweets are used in the form of data. The proposed model makes use of 997 tweets out of which 70% have been used for training method and the remaining 30% have been used for testing. The divisions of data for training and testing methods have been done randomly. In the hidden layer, the number of neurons has been kept as 10. The accuracy in the form of mean square error (MSE) of 97% (approx) has been obtained by the proposed system. The high accuracy can be attributed to the efficacy of the BRANN architecture and the pre-processing of raw data.

References

1. E. Baucom, A. Sanjari, X. Liu, M. Chen, Mirroring the real world in social media: twitter, geolocation, and sentiment analysis, Copyright 2013 ACM 978-1-4503-2415-1/13/10
2. Hanjun Lee Business School, Korea, The influence of negative emotions in an online brand community on customer innovation activities, in *2014 47th Hawaii International Conference on System Science*—978-1-4799-2504-9/14 $31.00 © 2014 IEEE
3. V. Sahayak, et al., Sentiment analysis on twitter data. Int. J. Innov. Res. Adv. Eng. (IJIRAE), ISSN: 2349-2163, vol. 2, Jan 2015
4. X. Hu, L. Tang, J. Tang, H. Liu, Exploiting social relations for sentiment analysis in microblogging, permission and/or a fee. WSDM'13, 4–8 Feb 2013 (Rome, Italy). Copyright 2013 ACM 978-1-4503-1869-3/13/02

5. M. Wang, D. Cao, L. Li, S. Li, R. Ji, Microblog sentiment analysis based on cross-mediabag-of-words model, ICIMCS'14, 10–12 July 2014 (Xiamen, Fujian, China). Copyright 2014 ACM 978-1-4503-2810-4/14/07
6. F. Jiang, A. Cui, Y. Liu, M. Zhang, S. Ma, Every term has sentiment: learning from emoticon evidences for Chinese microblog sentiment analysis (Springer, Berlin, Heidelberg, 2013)
7. F. Bravo-Marquez, M. Mendoza, B. Poblete, Combining strengths, emotions and polarities for boosting twitter sentiment analysis, WISDOM '13, 11 Aug 2013 (Chicago, IL, USA). Copyright 2013 ACM 978-1-4503-2332-1/13/08
8. P. Calais Guerra, W. Meira Jr., C. Cardie, Sentiment analysis on evolving social streams: how self-report imbalances can help, WSDM'14, 24–28 Feb 2014 (New York, New York, USA). Copyright 2014 ACM 978-1-4503-2351-2/14/02

Invertibility in Well-Covered and Perfect Graphs

D. Angel

Abstract Vertex cover is important to solve problems in developing and organizing intercommunication framework. A set of vertices of a graph is called a vertex cover, if this set covers all the edges in a graph. The minimum cardinality of such a set is called the covering number. In this paper, characterizations for a graph to possess inverse vertex covers are presented. Furthermore, the connection between well-covered graphs and different kinds of perfect graphs with invertibility is given.

Keywords Inverse vertex cover · Invertible graphs · Strong perfect Very strong perfect · Super strong perfect graphs

1 Introduction

The vertex cover problem is interesting to computer scientists for designing applications like wireless sensor placements and algorithms, particularly following soft computing techniques. In the construction of interconnection network defense, the fundamental target is to keep the edges of the network secure by placing the minimum number of defenders (devices). One the other hand, apart from safeguarding the network, one has to compensate for the devices if any of the devices have faults. This issue can be rectified with the help of invertible graphs. Invertible graphs are a subclass of bipartite graphs. Even though these are quite a small group of graphs, they find wide applications where the independence and covering numbers play an important role, since these graphs are special in the sense that a maximum independent set is also a covering set.

Characterizing invertible graphs is a quite interesting task as inverse cover does not exist for all the graphs. In this paper, the class of invertible graphs that contains independent covering sets is studied. The standard graph theory terminology is used

D. Angel (✉)
Department of Mathematics, Sathyabama Institute of Science and Technology, Chennai 600119, India
e-mail: angel.skye11@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_52

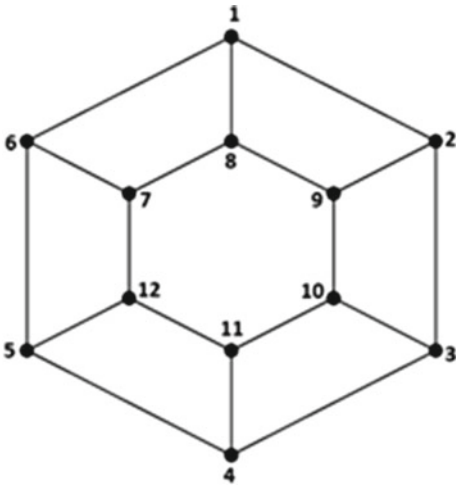


Fig. 1 An invertible graph

[1, 2]. For a given graph G, the set of vertices of a graph is called a vertex cover, if this set covers all the edges in a graph. The minimum cardinality of such a set is called the covering number $\beta(G)$. This set is also termed as β-set.

Impelled by the idea of inverse domination number [3], the concept of inverse vertex covering was considered by Kulli et al. in [4]. If there is a minimum vertex cover set K such that $V - K$ is also a vertex cover, then $V - K$ is called an inverse vertex covering of G with respect to K. The cardinality of the minimum inverse vertex covering set is called the inverse covering number $\beta^{-1}(G)$. Figure 1 shows an example of an invertible graph. Inverse covering number does not exist for every graph. A graph G for which the set $V - K$ is a covering is called an invertible graph [4]. An edge cover of a graph is the edge set which can cover all its vertices. The cardinality of the minimum edge covering is the edge covering number $\beta'(G)$. The size of the largest vertex independent set is called the independence number of G and is denoted by $\alpha(G)$, and the maximum independent set is called an α-set. All invertible graphs are bipartite graphs, whereas all bipartite need not be invertible (for example, see Fig. 2). This was the motivation to characterize bipartite graphs for invertibility.

2 Invertibility in Well-Covered Graphs

Graph characterizations are essential in evaluating the performance of the network. Construction and analysis of covering sets and invertible graphs helps us in the security of the networks as these sets are helpful in supervising the links in the network. In this section, several characterizations for invertibility are presented.

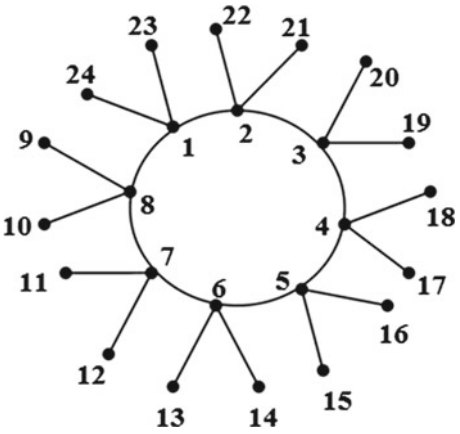


Fig. 2 A bipartite graph which is not invertible

A graph G is called a well-covered graph if each maximal independent set of G is an α-set. The following theorem provides a characterization of an invertible graph to be well covered.

Theorem 2.1 *If G is an invertible graph, then G is well covered if and only if G is balanced.*

Proof Given that G is an invertible graph. First it is necessary to prove that G is balanced assuming the fact that G is well covered. By the definition of a well-covered graph, an invertible graph is well covered if every α-set is also a β-set.

$\Rightarrow \alpha(G) = \beta^{-1}(G) = \frac{n}{2}$.

$\Rightarrow$ G is balanced [5].

Conversely, assume that G is balanced.

Then, $\beta'(G) = \beta(G) = \frac{n}{2}$.

To prove that G is well covered, take an α-set D of G.

Since G is invertible, D is also a vertex cover.

Since $\beta(G) = \frac{n}{2}$, $\alpha(G) = n - \beta(G)$.

$\Rightarrow \alpha(G) = n - \frac{n}{2} = \frac{n}{2}$.

$\Rightarrow$ D is also a β-set.

$\Rightarrow$ Every α-set is also a β-set.

$\Rightarrow$ G is well covered. □

3 Invertibility in Different Kinds of Perfect Graphs

A Berge graph is one which cannot have a cycle of odd vertices with length at least five or its complement as an induced subgraph. A graph is perfect if for each of its induced subgraphs H, $\chi(H)$ is equal to the maximum number of mutually adjacent vertices in it. A graph G is very strong perfect if for each induced subgraph H of G, every vertex of H is in an independent set of H which meets all the maximal complete subgraphs of H. A strong perfect graph is beneficial in modeling a position in which we can select an ideal set of officers from a given set of people [6]. This concept when the independent sets are replaced by dominating sets gives rise to super strong perfect notion. The idea of super strong perfect graphs was introduced by Murty in [7]. The following theorem characterizes different kinds of perfect graphs with respect to its invertibility.

Theorem 3.1 *For an invertible graph G, the following properties are equivalent.*

(i) *G is perfect.*
(ii) *G is very strong perfect.*
(iii) *G is super strong perfect.*
(iv) *G is Berge.*

Proof Given that G is an invertible graph.

(i) Since all invertible graphs are bipartite and all bipartite graphs are perfect, all invertible graphs are perfect.
(ii) Suppose if, G is not very strong perfect.

$\Rightarrow$ There exists a vertex of an induced subgraph H, which is not in an independent set of H and do not have all K_2 in H.
$\Rightarrow$ G does not possess any inverse cover.
Therefore, we get a contradiction to our assumption that G is an invertible graph.
$\Rightarrow$ G should be very strong perfect.

(iii) Since G is bipartite, G should be super strong perfect.
(iv) Since G is perfect, G is also a Berge graph. □

4 Conclusion

Invertible graphs are a class of graphs which find wide applications in areas where the independence and covering numbers play a significant role, as these graphs are special in the sense that a maximum independent set is also a covering set. Suppose in an invertible network, S is a vertex covering set and if an intruder attacks some nodes in S then the inverse covering set $V - S$ will take care of the role of S. This is possible only if the network is an invertible network. In this aspect, it is worthwhile to concentrate on covering and inverse covering sets in invertible graphs. It would

be of great interest to provide further results toward characterization of invertible graphs. Computing β-sets and inverse covers for graphs helps us in the defense of the networks as these sets are beneficial in supervising the links in the network.

References

1. F. Harary, Graph Theory (Narosa Publishers, 1993), pp. 21–23
2. M.C. Golumbic, Algorithmic graph theory and perfect graphs, 2nd edn. (Academic Press Inc., 2004)
3. E. Sampathkumar, L. Pushpalatha, Strong weak domination and domination balance in a graph. Discrete Math. **161**, 235–242 (1996)
4. V.R. Kulli, R.R. Iyer, Inverse vertex covering number of a graph. J. Discrete Math. Sci. Cryptogr. **15**, 389–393 (2012)
5. D. Angel, A. Amutha, On invertible graphs and its perfectness. J. Global Res. Math. Arch. **5**(1), 1–7 (2018)
6. G. Ravindra, Some classes of strongly perfect graphs. Discrete Math. **206**, 197–203 (1999)
7. U.S.R. Murty, Open problems. Trends Math. **25**, 381–389 (2006)

An Edutainment Approach to Enhance Teaching–Learning Process

Prashant Panse, Trishna Panse, Rakesh Verma, Dinesh K. Bhayal and Amit Agrawal

Abstract Edutainment is education through entertainment which is one of the innovative teaching–learning methodologies. In this paper, we are discussing various methods of teaching–learning process which can be benefited to learners (students). Edutainment also increases the interest of the learner (student) in classroom, and it provides good environment to understand the depth of the subject knowledge with real-life applications. It also involves learner into the discussion rather than traditional teaching where learner has to study from chalkboard, PPT and OHP. In this paper, few of the techniques like snake and ladders and crosswords are presented which can help learners (students) to improve their knowledge. The methodology presented in the paper is adopted in classroom which increases the interest of learners.

Keywords Snake and ladder · Education with entertainment · Crosswords · Game

1 Introduction

Edutainment is a vast area which includes PowerPoint presentations, spreadsheet games, video games, movies, stories, TV programs, puzzles, crosswords. These are found to catch learner interest, engage and motivate the learner and help them to retain knowledge for the longer period of time. "The idea of embedding academic

P. Panse (✉) · D. K. Bhayal
Department of IT, Medi-Caps University, Indore, India
e-mail: prashantpanse@gmail.com

T. Panse
Department of IT, Sushila Devi Bansal College of Technology, Indore, India

R. Verma
Department of CS/IT, Government Engineering College, Ajmer, India

A. Agrawal
Department of CSE, Medi-Caps University, Indore, India

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_53

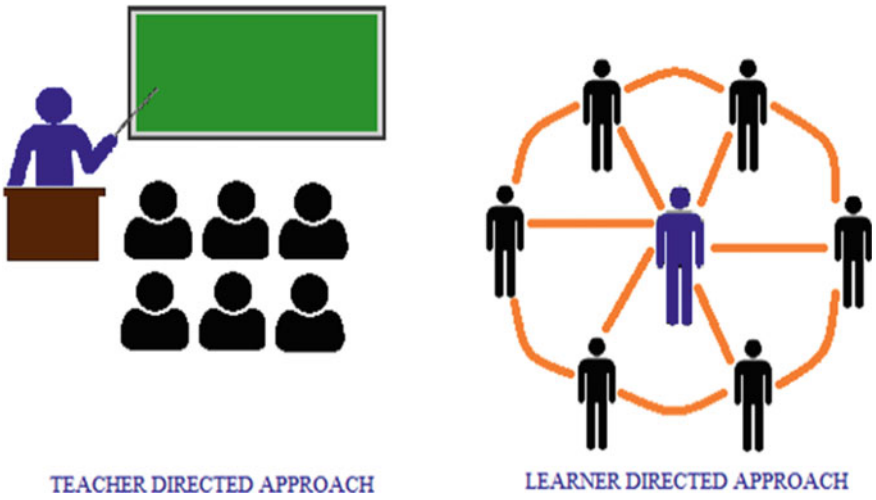


Fig. 1 Teacher-directed versus learner-directed approach

learning in an entertaining format has been used for centuries, because it works" [1]. One of the problems faced by many teachers is that "how to create interest for students in classroom?" Modern techniques available with us are helpful to answer this question. The above aids are useful for teachers to make lectures more effective than traditional (chalkboard) teaching method.

Learner-centric environment is needed to improve teaching–learning process. Learner-centric approach involves more participation of students, improves retention of knowledge, makes learning more fun, increases student performance and provides an opportunity for collaborative learning. Figure 1 shows a teacher-directed versus learner-directed approach. When we discuss new trends in teaching–learning process, the question that comes is what is the minimum requirement to opt for these technologies in teaching? Here the answer is that these technologies require a planning and execution of the plan as per requirement. If proper planning has been done on any course, it makes it easier to execute it in a proper manner which leads to positive output. Instructors are increasingly made aware of techniques that can be of benefit to their student's learning [2, 3].

2 Approaches for Effective Teaching

There are different approaches available for teaching. Many approaches are running from ancient time, e.g., chalkboard and explaining the topic in classroom using verbal communication. These techniques will lead to memorization of the topic for learner. These approaches have their own significance in teach-

ing process. With the change of scenario in level of students (learner), new technologies are coming which emphasizes a teacher to change from traditional to new approaches which include use of overhead projectors (where transparencies can be used), LCD projectors (where PowerPoint presentations can be used), puzzles, crossword, flipped classrooms, questioning, demonstrating, collaborating, quizzes, use of multimedia in classrooms, audio/video lectures.

2.1 Traditional Teaching Methods

Teaching gives knowledge or skills to students. It is also an act of imparting knowledge to the students in classroom session. Discussions are monopolized by teacher. This method emphasizes "what." It is passive in nature. It uses fixed methods of teaching and bounded under a classroom. Students learn topic before understanding how it might be useful. This method is also guided step by step, and learning is structured.

Traditional teaching generally is "one-way flow" of information. In this method teacher continuously delivers the information without knowing response from students. There is a lack of interaction with students, with more emphasis on theory rather than practical. It is learning from memorization and less on understanding.

2.2 Modern Teaching Methods

In modern method, it is said that "teaching is the process of providing guided opportunities for learner". In this, teacher becomes the facilitator. This method is active in nature. It is a student-centric approach where student takes participation in discussions. It emphasizes the "why" and "how" of learning, which motivates learners to explore the topic in depth.

Modern teaching methods involve problem solving, fieldtrip, demonstration, brainstorming, etc. There is no structure for learning. Flow of information is multidimensional, i.e., from facilitator to learner, learner to facilitator and from learner to learner. It is a kind of multimedia-driven learning which may include audio, video, presentations, puzzles, crosswords, game shows, role plays to describe a topic to the learners. National Training Laboratories in Betel, Maine, illustrates the percentage of learner recall that is associated with various approaches. Figure 2 shows the learning

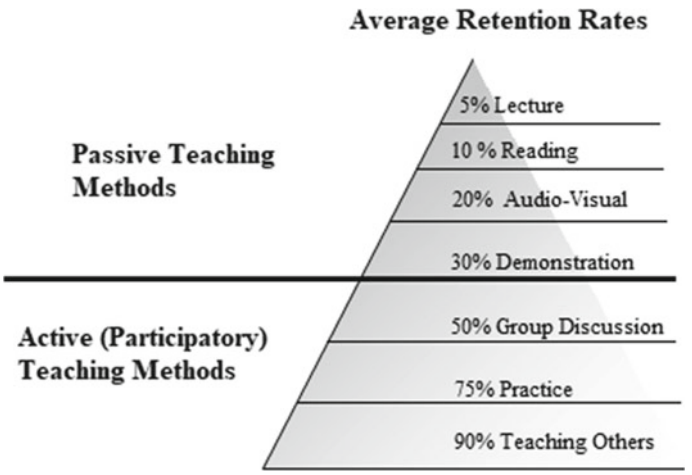


Fig. 2 Learning pyramid

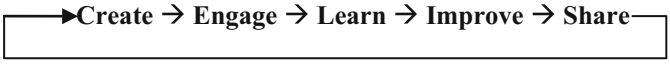
pyramid where the first four levels (lecture, reading, audiovisual and demonstration) are passive learning methods. In contrast, the bottom three levels (discussion group, practice by doing and teaching others) are participatory (active) learning methods. The learning pyramid clearly illustrates that active participation in the learning process results in a higher retention of learning [4].

Effective teaching process involves efforts of the teacher to create and maintain the interest of students. It requires planning of topic to be taught with respect to available time. The fundamental process involved in teaching is to prepare a content of topic. When the topic is prepared from different sources, it generally creates interest among students. Sources must be shared among students (learners) so that they can refer later on. Effective teaching strategies say that teacher must be able to make classroom as interactive as possible because this will help student to learn more. Teacher must respect students and student learning; it will motivate students. Teacher creates learning tasks rather than traditional assignment which enforce students to engage themselves in learning process. Teacher can use flipped classroom concept where video of the topic which is to be taught is already shared with students before lecture then in classroom; it is expected from all students that they should discuss with teacher and among themselves regarding topic which they have gone through a video. Teacher can motivate students of higher education to write an article/research paper toward their interest.

3 Approaches for Effective Learning

Effective learning is used to ignite the minds of students. Effective learning can be implemented only when the learning environment is created. It is sometimes difficult in classroom to create learning environment. It has been observed that students are more interested in getting degree or job rather than knowledge. The students must be motivated for learning. LBF (Learn By Fun) common technique is available in schools which activate the minds of students toward learning. Change in systems is based upon a comprehension of components which influence student learning [5].

Effective learning is very useful for both types of learners (slow and fast). It allows them to learn in their pace. It gives alternatives for students to learn from different resources. It can save teacher's time and effort to reproduce material (avoid reinventing the wheel). Effective learning has different phases. It includes activation, application, demonstration and integration. Effective learning also includes the process as shown below.



3.1 Learning Techniques

There are different learning techniques available for a learner. Some of them are underlining, highlighting, rereading, summarizing, mental imagery, mnemonics, self-explanatory, practice test, distributed practice, flashcards. These techniques have their own boundaries so learner can get benefited from these. Table 1 describes mapping of these techniques with efficiency of different methods with spent time for doing these. It is clear from the table that learning can be efficient only when learner is doing by own. Although the time requires for this is little more, it has been found that undergraduate learners are more interested in techniques which require less time.

4 Methodology Adopted

In this section we will describe the methods which have been adopted in teaching –learning. It includes flipped classroom, learning by doing, flash cards, summarizing the topic, lectures; we have also given assignments to our students which are different

Table 1 Mapping of efficiency and time for learning methods

Learning methods	Efficiency of methods	Time spent for doing this
Highlighting	Low	Very less
Underlining	Low	Very less
Rereading	Low	Less
Summarizing	Low	Less
Mental imagery	Intermediate	Medium
Mnemonics	Intermediate	Medium
Self-explanation	Intermediate	Medium
Flash cards	High	More
Practice testing	High	More
Distributed practice	High	More

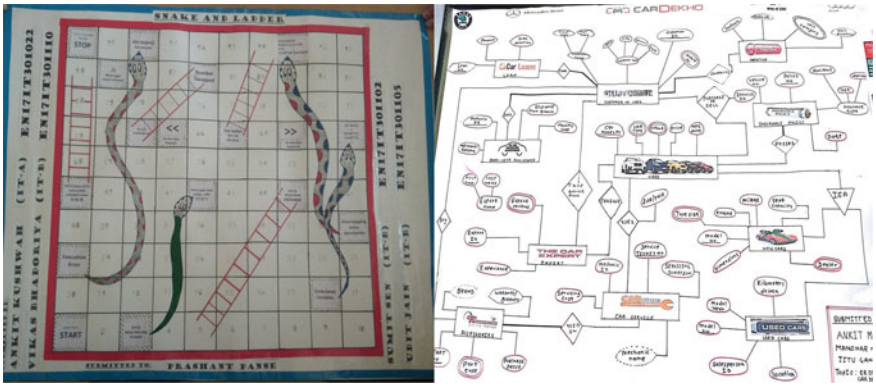


Fig. 3 Snake and ladder and ER diagram designed by learner

from traditional writing work. Assignments include games like prepare a snake and ladder game on topic. Students are evaluated on the basis of crosswords which are a kind of brainstorming; it leads to give idea to them for deep understanding of topics. Students are also encouraged for developing an ER diagram (used in DBMS subject) for real-life problems like www.cardekho.com. Students have also created their own crosswords for a subject like C programming, which helps them to remember the terminology used in the subject. Figure 3 shows the pictures of the work done by learner.

The assignments are given to student after teaching. Figure 4 shows the crosswords created by students, which help them to enhance their knowledge in the subject.

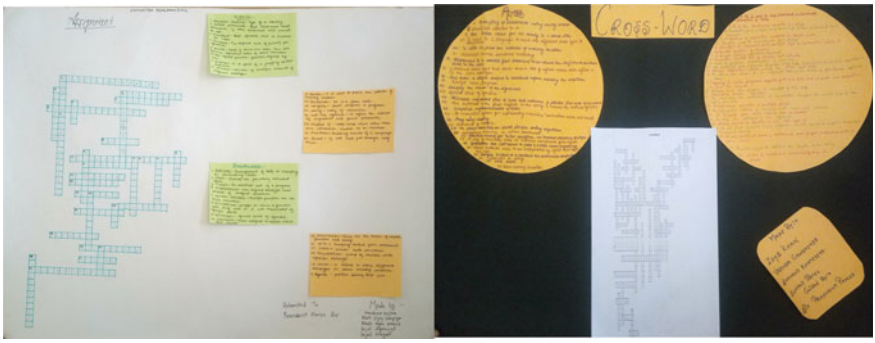


Fig. 4 Crosswords made by students for C programming subject

5 Experimental Design and Analysis

To analyze and evaluate the impact of modern techniques in learning, three experiments were designed and performed. Engineering undergraduates are considered as subject population to accomplish designed experiments. Experiments were designed in the form of "**snake and ladder**" game for computer network course that given as assignments. The second one was "**crossword**" for C programming, and the third one was ER diagram for real-world database application, i.e., "www.cardekho.com." The undergraduates of subject population belong to diverse years such as first, second and third years. The experiments were performed in three phases.

Phase 1—Induction phase: In this phase, we had intended to aware and tell about numerous modern methods of learning and also aware of how to make interest in assignment completion.

Phase 2—Execution Phase: After the induction phase, we were given assignment task to undergraduate of different years. As an assignment, design snake and ladder game for computer network assigned to third-year undergraduate, design ER diagram for www.cardekho.com assigned to second-year undergraduate and crossword for the first-year undergraduate.

Phase 3—Response Phase: Here, response and views had taken on assignments from the subject population. Response was collected by asking of few designed questions such as how much you learn, how it was experienced, is it create interest or not.

The designed experiment is analyzed through storing taken views or feedback from undergraduate in the form of response that was provided during the response phase. The views or feedback taken through the designed questionnaire consists of

Learner Views on Pedagogy Tools and Techniques in Learning

Please tick (√) mark on appropriate option.

Sr.	Details	Yes	No	Can't Say
1	Is modern learning including charts, games, animations, videos, imaging etc. more impactful than traditional learning?			
2	Does it make interest and attention in class?			
3	Does it create zeal and enthusiasm in learners?			
4	Is it improves assignment nature as well as interest in doing assignment?			
5	Does it make clear proper meaning and relation between concepts to the learner?			
6	Does it results non-volatile things in minds of learner about any concept?			
7	Is it enhancing learning process?			
8	Does it results less time to learn?			
9	Is it more effective and efficient compare to traditional learning?			
10	May it continue for further?			
11	May it apply in all area of learning?			
12	Is it time consuming than others?			
13	Is it bore to the learners?			
14	Are you agreed to learn with this technique?			
15	Can you suggest to uses this for all areas?			

Suggestions:

Please suggest any changes or improvements which are required in learning technique enhancement. All of your suggestions are welcome.

Fig. 5 Designed questionnaire form

several aspects. The designed questionnaire was shared to subject population via digital form, i.e., designed Google form. Figure 5 shows the designed questionnaire form for taking view or response.

The data facts collected from the response were stored and processed to analyze the impacts of learning and assignment methods. Views of subject population were taken in one among three choices: YES, NO and CAN'T SAY on different aspects. They had individual view on all aspects that was given in the form of YES or NO or CAN'T SAY. We analyzed how much of population agreed, disagreed or confused on various aspects. To represent all the views accordingly, desired aspects and data facts were processed and depicted in the following figures.

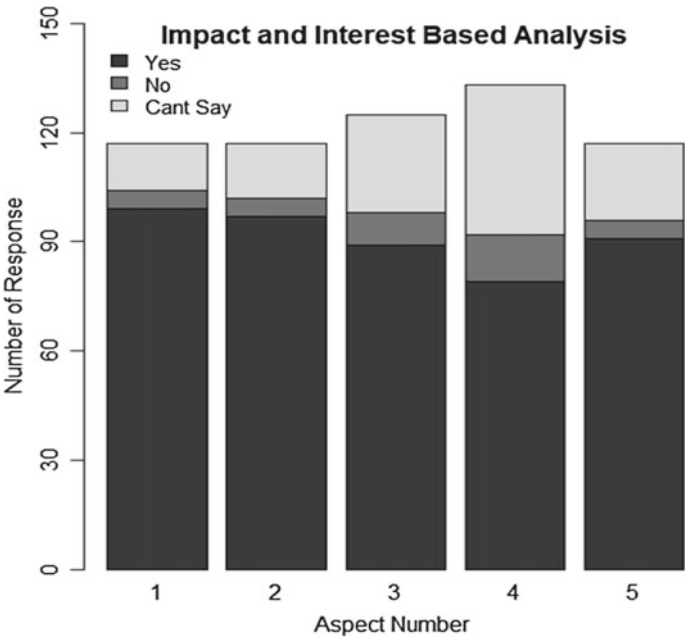


Fig. 6 Learner response based on impact and interest

The response of the subject population is represented in the figure as the form of column chart that was categorized as impact and interest based, and efficiency and time computation based. Figure 6 shows the response of learners with respect to impact and interest.

Further, Fig. 7 represents response accordingly to efficiency and time computation aspect based.

At last, but not least we concluded the entire analysis of modern learning techniques in all desired aspects. The entire analysis is represented in Fig. 8.

As the discussion and analysis on modern learning methods, it resulted as fruitfulness in learning and teaching as well as assignments in nature. Crux of the whole discussion was that these kinds of methods may be used in learning of any field.

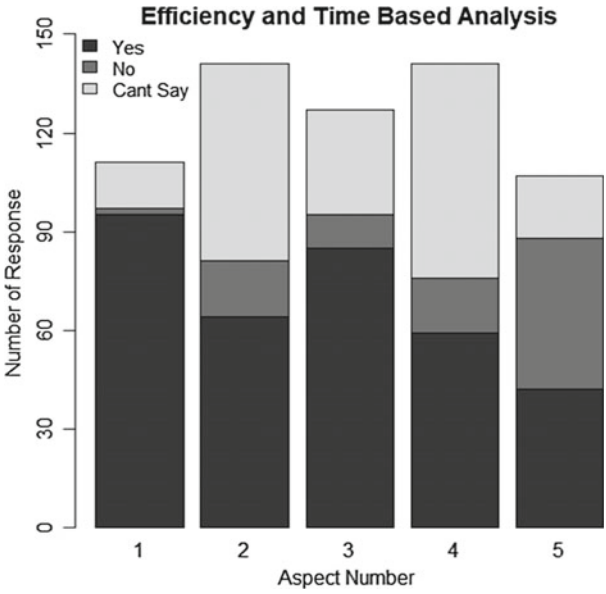


Fig. 7 Learner response based on efficiency and time computation

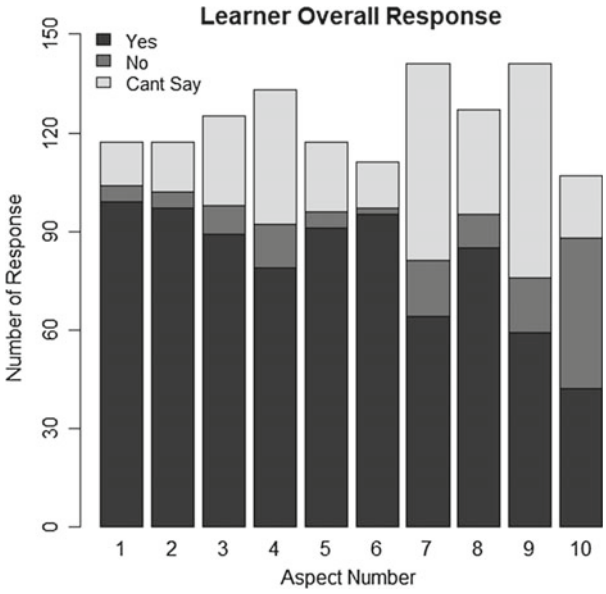


Fig. 8 Overall learner response

6 Conclusion

It has been observed that after integration of different methods it would be possible to make class interesting. Learners are enjoying in classroom after adopting effective teaching and learning methods discussed in the paper.

References

1. E. Jensen, *The Learning Brain*, 1st edn. (Turning Point Publishing. San Diego, CA, 1995)
2. R.J. Dufresne, W.J. Gerace, W.J. Leonard, J.P. Mestre, L. Wenk, Classtalk: a classroom communication system for active learning. J. Comput. High. Educ. **7**, 3–47 (1996)
3. L. Wenk, R. Dufresne, W. Gerace, W. Leonard, J. Mestre, Technology-assisted active learning in large lectures, in *Student-Active Science: Models of Innovation in College Science Teaching*, ed. by C. D'Avanzo, A. McNichols (Saunders College Publishing, Philadelphia, PA, 1997), pp. 431–452
4. http://thepeakperformancecenter.com/educational-learning/learning/principles-of-learning/learning-pyramid/
5. A.K. Ditcher, Effective teaching and learning in higher education, with particular references to the undergraduate education of professional engineers. Int. J. Eng. Educ. **17**(1), 24–29 (2001)

Two-Step Anomaly Detection Approach Using Clustering Algorithm

Praphula Kumar Jain and Rajendra Pamula

Abstract The goal of anomaly detection is to find out the new feature that is different to others. In this paper, we are finding anomalies using a two-step approach. In the first step, the modified k-mean algorithm is used for dividing the data patterns/objects or points into clusters. Data points in the same cluster will be mostly anomalies or all non-anomalies. In second step, constructing a max heap is based on the number of points in the clusters. Points are present in the cluster at the leaf level; they all are distant from the other clusters regarded as anomalies.

Keywords Anomaly detection · K-mean · Clustering · Max heap

1 Introduction

Patterns or points in the data that do not follow the normal behavior refer to anomalies or outliers [1]. The process of finding anomalies is called anomaly detection. There are different anomaly detection techniques are available in the literature that include statistical anomaly detection, data mining-based methods, and machine learning-based techniques [2]. Clustering technique is the example of the data mining-based method. Clustering is the task of grouping a set of objects/patterns in such a way that object in the same group is similar to each other and dissimilar to the objects in the different groups. The goal of such clustering technique is to identify the distinct groups in the data set.

In this paper, we are presenting the anomaly finding method based on clustering approach. There are various application of anomaly detections, such as credit card fraud detection, insurance or health care, fault detection in safety-critical systems and military surveillance for enemy activities. The importance of anomalies detection is due to the fact that anomalies in the data require serious attention in a wide variety of applications. For example, in a computer network hacked computer system sending

P. K. Jain (✉) · R. Pamula
Indian Institute of Technology (Indian School of Mines), Dhanbad, Jharkhand, India
e-mail: praphulajn1@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_54

out sensitive data to the unauthorized user, this is due to the presence of an anomalous traffic pattern [3]. An anomalous MRI image may indicate the presence of malignant tumors [4]. Anomalies in credit card transaction data could indicate credit card or identity theft [5].

2 Related Work

Anomaly detection is the topic for many research paper like [6, 7], and from the many review papers one paper is [8]. Anomaly detection is implemented using different techniques, from them statistical approaches were presented by [9, 10]. The main goal of the clustering algorithms is to find out the data points that are closest to each other [6]. There is no exist of the best clustering algorithm because it depends on the observation and type of the data also. Now a days, Clustering algorithms are very useful in the fields such as computer science, medicine, biology, engineering, and future prediction.

There are several of clustering algorithms which are available from the previous literature work. It is very difficult to classify the available clustering algorithm. Among the many of clustering algorithms, some are k-means and modified k-means are

2.1 K-mean Clustering Algorithm

The K-mean clustering algorithm [7] uses for optimal clustering with a fixed number of cluster. Firstly start partitioning with chosen number of clusters next improve the partitions iteratively. Let $X = \{X_1, X_2, \ldots, X_n\}$ be the set of data points and $V = \{V_1, V_2, \ldots, V_t\}$ be the set of centers.

K-mean Algorithm

1. At random select t number of cluster centre.
2. Find out the distance form each data point to each cluster centres.
3. Allocate the data point to the nearest cluster centre in comparing with other Clusters centres.
4. Again find out the new cluster centre by $v_i = \sum_{j=1}^{ti} X_i$ where t_i represents number of data points in i^{th} cluster.
5. Again find out the distance form each data point to new cluster centre.
6. If no data points was reassigned then stop, otherwise go to step 3.

2.2 Modified K-mean Algorithm

For the different data and with different initial cluster, k-mean cannot generate the optimal results. In modified k-means clustering approach [8] if any new point is far away from all cluster centers, then create a new cluster and assign that point to created cluster.

Consider t′ is the number of clusters that can be flexible in number. Let $X = \{X_1, X_2, \ldots, X_n\}$ be the set of data points. In the beginning, select $t' = t$, and initially choose random t′ points as the cluster centers, $V = \{V_1, V_2, \ldots, V_{t'}\}$. In the time of iteration, when assigning one pattern to its nearest cluster, we compute two things here,

(i) Minimum distance between any two cluster centers,

$$\min(m) = \min \left\| V_j - V_h \right\|^2 \text{for } j, h = 1, 2, \ldots, t'.$$

(ii) For any pattern X_i the minimum distance of its nearest cluster center

$$\min(X_i, d) = \min \left\| X_i - V_j \right\|^2 \text{ for } j = 1, 2, \ldots, t'$$

Modified K-mean Algorithm

1. Randomly select t' cluster centre.
2. For i=1 to n compute min (m) and min (Xi, d).
3. If min (Xi, d) ⩽ min (m) then go to Step 6.
4. Splitte Process: Assign *Xi* to be the center of a new cluster Vt' and set $t' \leftarrow t'+1$.
5. Merge Process: if $t' > t_{max}$, combines the two nearest clusters into single cluster and set $t' = t_{max}$.
6. Allocate PROCESS: Assign *Xi* to its closest cluster.
7. Stop if cluster membership stabilized otherwise go to step 2.

2.2.1 Merging Process

In this process merge the two existing clusters. Set Vk1 and Vk2 are the centers of the two nearest clusters that are going to merge. Thereafter, both clusters combined by $Vk^* = 1/(Nk1 + Nk2) * (Nk_1 * X_1 + Nk_2 * X_2)$, where Vk* is the center of the new cluster, and Nk1, Nk2 is the number of points in the two clusters. Clearly, Vk1 and Vk2 are replaced by Zk* and the number of clusters $t' = t' - 1$.

2.2.2 Allocate Process

For each clustering cycle, we dispense all pattern to the nearest group, with the end goal that the total of squared Euclidean separations between each example and the center of the relating cluster are limited.

2.3 Binary Heap

A Complete binary tree that should follow the concept of heap ordering property referred as a binary heap. The heap ordering property can be min or max heap.

(1) The min-heap property: parent node having equal or less value than its child node. (2) The max-heap property: parent node having equal or greater value than its child node.

In the whole paper, the word "heap" will always refer to a max heap. In a heap the highest/lowest priority element is always stored at the root, hence the name "heap".

3 Proposed Approach

In the first step, we are using the updated k-means algorithm in place of the k-mean algorithm. In a second step, we are constructing the max heap and declaring the points in leaf node as an anomaly.

4 Result Analysis

4.1 Iris Dataset

This publically available dataset (https://archive.ics.uci.edu/ml/datasets/iris) contains 3 classes (Iris Setosa, Iris Versicolor, Iris Virginica) of 50 instances each, where each class refers to a type of iris plant. The dataset has 4 attributes (Sepal length in cm, Sepal width in cm, Petal length in cm, Petal width in cm) (Tables 1, 2, 3 and 4).

From the above result analysis, we can say that in the case of k-means algorithm, Iris dataset having fewer number points in the cluster 3 when K = 3 and from cluster 4 when K = 4. In case of modified k-means algorithm, cluster 3 and cluster 4 have less number of points, respectively, for the k values 3 and 4.

Figure 1 represents the max heap for the modified K-mean clustering algorithm. In this figure when K = 3 cluster 3 is at the leaf node and having only 11 data points so these points are anomalies. When K = 4 cluster 3 is at the leaf node and having only 6 data points that represents the anomalies.

Table 1 K-means clustering with K = 3

Number of clusters = 3	Mean		Type 1	Type 2	Type 3	Total points
	Sepal length	Sepal width				
Cluster 1	6.28488	3.20698	22	21	43	86
Cluster 2	5.42353	2.77255	15	29	7	51
Cluster 3	4.56923	3.14615	13	0	0	13

Table 2 K-means clustering with K = 4

Number of cluster = 4	Mean		Type 1	Type 2	Type 3	Total points
	Sepal length	Sepal width				
Cluster 1	6.28488	3.20698	22	21	43	86
Cluster 2	5.45	2.72917	12	29	7	48
Cluster 3	4.78	3.38	10	0	0	10
Cluster 4	4.4333	2.91	6	0	0	6

Table 3 Modified k-means clustering with K = 3

Number of cluster = 3	Mean		Type 1	Type 2	Type 3	Total points
	Sepal length	Sepal width				
Cluster 1	5.20625	3.11375	50	26	4	80
Cluster 2	6.39661	2.95763	0	24	35	59
Cluster 3	7.50909	3.13636	0	0	0	11

Table 4 Modified k-means clustering with K = 4

Number of cluster = 4	Mean		Type 1	Type 2	Type 3	Total points
	Sepal length	Sepal width				
Cluster 1	5.01633	3.44082	49	0	0	49
Cluster 2	6.10133	2.863	0	46	29	29
Cluster 3	4.9000	2.3333	1	4	1	6
Cluster 4	7.185	3.14	0	0	0	20

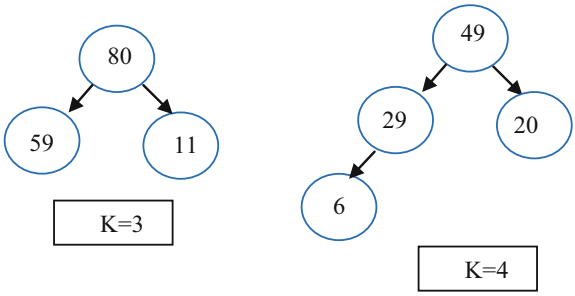


Fig. 1 Max heap for iris dataset

4.2 Wine Dataset

The publically available dataset (https://archive.ics.uci.edu/ml/datasets/wine) contains total 178 number of instances; it contains three types of wines grown in the same region in Italy but derived from three different cultivars. The dataset has 13 attributes (Tables 5, 6, 7 and 8).

From the above result analysis, we can say that in the case of k-means algorithm, WINE dataset has fewer number points in the cluster 3 when K = 3 and from cluster 4 when K = 4. In case of modified k-means algorithm, cluster 1 and cluster 3 have less number of points, respectively, for the k values 3 and 4.

Figure 2 represents the max heap for the modified K-mean clustering algorithm. In this figure when K = 3, cluster 1 is at the leaf node and having only 15 data points so these points are anomalies. When K = 4, cluster 3 is at the leaf node and having only 9 data points that represents the anomalies.

Table 5 K-means clustering with K = 3

Number of cluster = 3	Mean		Type 1	Type 2	Type 3	Total points
	Alcohol	Malic acid				
Cluster 1	12.6103	1.65204	25	59	9	93
Cluster 2	13.0538	3.82213	9	11	33	53
Cluster 3	14.0469	1.86594	25	1	6	32

Table 6 K-means clustering with K = 4

Number of cluster = 3	Mean		Type 1	Type 2	Type 3	Total points
	Alcohol	Malic acid				
Cluster 1	12.5222	1.66586	20	59	8	87
Cluster 2	13.0685	3.84019	9	10	33	52
Cluster 3	13.9848	1.6208	21	2	2	25
Cluster 4	13.9643	2.195	9	0	5	14

Table 7 Modified k-means clustering with K = 3

Number of cluster = 3	Mean		Type 1	Type 2	Type 3	Total points
	Alcohol	Malic acid				
Cluster 1	12.7673	4.64067	0	4	11	15
Cluster 2	12.9347	1.89291	56	67	18	141
Cluster 3	13.5818	3.60727	3	0	19	22

Table 8 Modified k-means clustering with K = 4

Number of cluster = 3	Mean		Type 1	Type 2	Type 3	Total points
	Alcohol	Malic acid				
Cluster 1	12.5809	3.44455	4	14	15	33
Cluster 2	12.9900	1.6574	52	56	11	119
Cluster 3	13.2778	5.070	0	1	8	9
Cluster 4	13.7429	3.49059	3	0	14	17

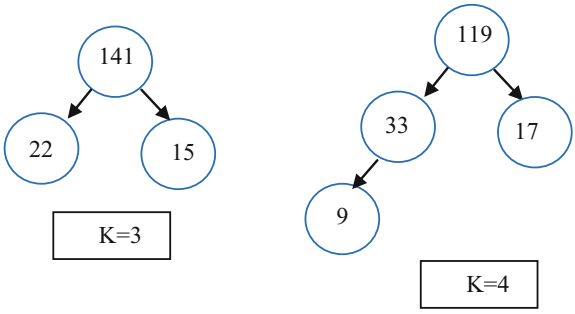


Fig. 2 Max heap for wine dataset

4.3 Max Heap Construction

From the result obtained in Sects. 4.1 and 4.2, we can construct a max heap on the basis of a number of points in the cluster. The cluster has a minimum number of points which will appear on the leaf node label. Leaf node of max heap declared as anomalies because of the points in that node having.

5 Conclusion and Future Work

Mostly in the case of the clustering algorithm, the anomalous points are avoided or associated with the other patterns. However, for a particular application, we have to search out that anomalous patterns from a huge dataset. In this work, we presented anomaly detection based on a two-step process to identify anomalies. In the begin-

ning, the k-means and the modified k-means algorithm are used to cluster the dataset. In the next step, we find out the anomalies based on the points in the clusters. In this step, a maximum heap is constructed based of the number of points in the cluster. The small clusters, the tree with less number of nodes, come at the leaf level which are selected and regarded as anomalies.

According to the iris dataset, the results represent the implementation finds out the minor and anomaly patterns. In wine dataset mostly anomalies belong to the clusters 3 and 4. We can also further implement this work for the social media anomaly detection based on the high and value of comments on the message box.

References

1. V. Chandola, A. Banerjee, V. Kumar, Anomaly detection: a survey. ACM Comput. Surv. (CSUR) **41**(3), 15 (2009)
2. I. Foster, C. Kesselman, J. Nick, S. Tuecke, The physiology of the grid: an open grid services architecture for distributed systems integration. Technical report, Global Grid Forum (2002)
3. P. May, H.C. Ehrlich, T. Steinke, ZIB structure prediction pipeline: composing a complex biological workflow through web services, in *Euro-Par 2006*, vol. 4128, LNCS, ed. by W.E. Nagel, W.V. Walter, W. Lehner (Springer, Heidelberg, 2006), pp. 1148–1158
4. I. Foster, C. Kesselman, *The Grid: Blueprint for a New Computing Infrastructure* (Morgan Kaufmann, San Francisco, 1999)
5. K. Czajkowski, S. Fitzgerald, I. Foster, C. Kesselman, Grid information services for distributed resource sharing, in *10th IEEE International Symposium on High-Performance Distributed Computing* (IEEE Press, New York, 2001), pp. 181–184
6. C.C. Aggarwal, *Outlier Analysis* (Springer Science and Business Media, 2013)
7. C.C. Aggarwal, P.S. Yu, Outlier detection for high dimensional data. ACM Sigmoid Record **30**(2)
8. M.-F. Jiang, S.-S. Tseng, C.-M. Su, Two-phase clustering process for outliers detection. Pattern Recogn. Lett. **22**(6), 691–700 (2001)
9. J. Han, J. Pei, M. Kamber, *Data Mining Concepts, and Techniques* (Elsevier, 2011)
10. V. Barnett, T. Lewis, Outliers in statistical data (Wiley, Chichester, 1995, 1964), 584 p.

Outlier Detection Using Subset Formation of Clustering Based Method

Gaurav Mishra, Shubham Agarwal, Praphula Kumar Jain and Rajendra Pamula

Abstract Data points which do not show the common characteristics of the data are called outliers. Outlier Detection can be used for various applications such as detecting frauds, customized marketing, and the terrorism search. Outliers are, in general, rare and hence only form a very small fraction of the data. Using Outlier Detection for various purposes is a difficult task. This paper proposes clustering based K-means algorithm with three closest cluster count K, K+1, K − 1 and computing radius of each cluster. Data point which is greater than radius of each cluster treated as outliers and distance based function helps us to detect top n outliers. Distance based values of outlier group point are considered as score. Higher the distance function value of a point, farther is the point from centroid and more is its probability of being an outlier. The results produced by the technique we propose here are found better in terms of outliers detected and time complexity, than techniques existing.

Keywords Data mining · Outlier · Cluster based · Distance based

1 Introduction

Outlier is a data point which deviates extremely from the remaining set of data. To find the deviant point among the data point is one of the major problems in data mining

Gaurav Mishra and Shubham Agarwal have contributed equally to the work.

G. Mishra (✉) · S. Agarwal · P. K. Jain · R. Pamula
Indian Institute of Technology (Indian School of Mines) Dhanbad, Dhanbad, Jharkhand, India
e-mail: grvmishra788@gmail.com

S. Agarwal
e-mail: agarwal.shubham7599@gmail.com

P. K. Jain
e-mail: praphulajn1@gmail.com

R. Pamula
e-mail: rajendrapamula@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_55

application that necessitates for the introduction of concept of outlier detection [1]. Detection of outliers reduces the possibility of delivering futile results based on erroneous data and also aids in preventing faulty behaviour. Detection of outlier eliminates the objects mostly deviating from the given dataset [2, 3]. Outliers may be considered as erroneous or real observation, that are incorrect due to recording error in the data collect process. The important methods of outlier detection can be classified into one of these—distribution based, depth based, clustering, distance based or density based.

The proposed system is a hybrid of two techniques—clustering based and distance based. We use K-means algorithm with three different closet cluster count to cluster data and to detect outliers from clustering of each cluster count, we use distance based approach.

In this proposed work, we trace the data point which belongs to outlier group based on clustering algorithm and distance based function. Data points which is in outlier group must belongs to any one of the cluster. Then, we calculate measure of distance of all points and classify all those points whose distance based measure is more than average radius of the cluster in the outlier group. Then we take the subset of outliers from three different clustering to form our required set of outliers.

2 Related Work

Outlier detection has been proposed by many authors with different methods. Author [4] first suggested distance based method for outlier detection. An object o in the dataset DS is a DBM(p, dist)-outlier if more than or equal to a fraction p of the object lie at a distance greater than eqdist from o. This definition is generalized and used in several statistical outlier test. Author [2] recommend the extension of above definition using outlier score, ranked were assigned to all points.

Author [5] recommended a clustering based algorithm for outlier detection aimed towards efficacious data mining which uses augmented K-means algorithm to cluster the data sets and weight based centre approach. In this method a threshold value is used to detect outliers. This threshold value can be calculated as by taking minimum and maximum value of a particular cluster. Author [6] suggest method for outlier detection which is a local outlier mining algorithm based on depth based and distribution based technique and is also micro clustering based.

Clustering techniques are basically used to find clusters. These techniques are mainly emphasized to obtain optimize clustering not to optimize outlier detection. There are clustering algorithms such as CLARANS, DBSCAN, BIRCH and CURE which automatically consider anomalies, however only to the point so as to ensure that they don't meddle with the clustering procedure. Assist, the meaning of anomalies utilized is as it were subjective and identified with the clusters that are distinguished by these algorithms. This is as opposed to our meaning of separation based exceptions which is more goal and free of how cluster in the input dataset are distinguished. While previous work on anomalies concentrates just on the identification viewpoint,

the work in [7] likewise endeavours to give deliberate learning, which is essentially a clarification of why an identified anomaly is different from the rest.

3 Proposed Work

The key idea of our proposal is to reduce the computational cost of running k-means clustering three times separately for detection of outliers by simultaneously running k-means three times.

We use the following method to detect outliers. First, cluster the dataset. For clustering, we use K-means algorithm. We take three different cluster count of K i.e. K, K + 1 and K − 1 simultaneously. For each value of K, separate outlier will be detected using radius of each cluster. The data points which have greater distance from centre than radius of cluster are considered as exception points. In the wake of identifying anomaly independently, we take crossing point of these individual exception set and after this the points which we will get, are considered as real outliers.

3.1 Outline

Here are the steps which need to be performed by our algorithm.

(1) Initially, we divide the entire dataset into K clusters utilizing K-means algorithm and compute the radius of each cluster.
(2) Compute the distance of each point of a cluster from the centroid of the corresponding cluster. If this distance is greater than the span of a cluster, then the point is considered as an outlier point.
(3) We repeat steps 1 and 2 with two more cluster count of K i.e. K + 1 and K − 1. For initial cluster centres of K + 1-means, we add one more cluster centre seed point, which is the mean of all other existing k-cluster centres to the existing k-centres (step no. 12 in Algorithm 1). For initial cluster centres of K – 1 means, we remove one cluster centre seed point (centre of cluster having minimum elements) from the existing k-centres (step no. 16–22 in Algorithm 1). This helps in early convergence of K + 1 and K – 1 means respectively.
(4) Computing outlier points: Take subset intersection of outliers found in the three clustering.

3.2 System Architecture

Algorithm 1 Outlier Detection using K means with three different closest cluster count

```
1: Set Yk <- K_Means (k, it, Ds) with random Ds
2: for every cluster Cj ∈ Yk do
3:       Radius <- radius(Cj)
4:     for every point pi do
5:       if dist(pi ,Cj ) < Radius then
6:          Prune pi
7:       else
8:          Add pi to Og1
9:       end if
10:    end for
11: end for
12: Calculate the mean V0 of all existing centroids.
13: Ds1 = Ds ∨ V0
14: Set Yk+1 <- K_Means (k+1, it, Ds1)
15: Repeat steps 2 to 11 to get Og2
16: Min <- count(C1)
17: for each cluster C2 to Cj do
18:       if count (Cj) < min then
19:         Update min
20:       end if
21: end for
22: Delete Cj with minimum count whose centroid is Vj.
23: Ds2 = Ds - Vj
24: Set Yk-1 <- K_Means (k-1, it, Ds2)
25: Repeat steps 2 to 11 to get Og3
26: Oset <- Og1 /\ Og2 /\ Og3
```

In above algorithm, **Ds** is the set of centroids, **it** is the total number of iterations required for convergence of the algorithm, **k** is the total number of clusters, and n is the total number of data points.

The time complexity of K-means algorithm is (it * k * n). The proposed method takes centroid seed points from first K-means convergence and utilizes it in the (K+ 1)- and (K − 1)-means thus reducing the number of iterations in the latter parts. In case, we had applied K-means three times separately then the number of iterations for convergence would be sufficiently more each time than the proposed method. The total computation of proposed method is less than 3 times (it * k * n). Overall, since k and it are small, so the total time complexity of the proposed method is O(n).

4 Result Analysis

Here, we have applied our entire algorithm on multiple datasets for different K-values. For example, in case of IRIS dataset, we have applied our K, K + 1, K − 1 algorithm for three K-values—(3, 5, 20) and then pointed out the K-value which gives the best results. For a particular K-value, example for K = 3, the algorithm generated outliers by k-clustering dataset 3 times i.e. for k = 2, 3 and 4 and then took a subset of outlier detected in each case. Similarly, for K = 20, the algorithm generated outliers by k-clustering dataset for k = 19, 20 and 21 and then took a subset of outlier detected in each case. The results are calculated as per following formula

$$\begin{aligned} &\textit{Outlier percentage of a class} \\ &= \frac{\textit{Number of elements in the class deemed as outlier.}}{\textit{Total number of elements in the class}} \end{aligned}$$

4.1 IRIS Dataset

This data set (https://archive.ics.uci.edu/ml/datasets/iris) contains of 3 classes (Iris Setosa, Iris Versicolor, Iris Virginica). These classes have 50 instances each. Each instance has 4 attributes (Sepal length in cm, Sepal width in cm, Petal length in cm, Petal width in cm).

In IRIS dataset, Iris-Setosa class is linearly separable from rest 2. Hence, most of the outliers should belong to Iris-Setosa class. From the Table 1, it is clear that for a good k-value (here, for k = 3), we are getting most of the outliers from Iris-Setosa class (i.e. 20% of total flowers in Iris-Setosa class).

4.2 WINE Dataset

The data set (https://archive.ics.uci.edu/ml/datasets/wine) contains total 178 instances of three types of wines grown in the same region in Italy. The dataset has 13 attributes.

Table 1 Results for IRIS dataset

Value of K	% Outliers from IrisSetosa	% Outliers from IrisVersicolor	% Outliers from IrisVirginica
K = 3	20.00	10.00	14.00
K = 5	18.00	18.00	30.00
K = 20	36.00	32.00	42.00

Table 2 Results for WINE dataset

Value of K	% Outliers from class 1	% Outliers from class 2	% Outliers from class 3
K = 3	22.00	18.00	4.00
K = 10	38.00	48.00	36.00
K = 20	34.00	44.00	14.00

Table 3 Results for ZOO dataset

Value of K	% Outliers from different classes						
	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7
K = 3	12.00	0.00	20.00	7.69	25.00	100.00	100.00
K = 7	21.95	40.00	60.00	7.69	0.00	12.50	80.00
K = 20	39.02	25.00	60.00	7.69	0.00	21.50	80.00

In WINE dataset, most of the outliers belong to Class 1. From the Table 2, it is clear that for a good k-value (here, for k = 3), we are getting most of the outliers from Class 1 class (i.e. 22% of total elements in this class).

4.3 ZOO Dataset

It is a data set (https://archive.ics.uci.edu/ml/datasets/zoo) containing 17 Boolean-valued attributes of total 101 instances of animals in a zoo. The dataset belongs to 7 classes of animals.

In ZOO dataset, most of the outliers belong to Class 2, 3 and 7. From the Table 3, it is clear that for a good k-value (here, for k = 7), we are getting most of the outliers from these classes (i.e. 40% of total elements in this class 2, 60% of total elements in this class 3 and 80% of total elements in this class 7).

4.4 YEAST Dataset

This dataset (https://archive.ics.uci.edu/ml/datasets/yeast) contains information about localization sites of protein in yeast bacteria, based on several bio-statistical tests. It contains 1484 such instances, with each instance in dataset having 6 attributes each.

In YEAST dataset, Class ERL serves as collective outlier and apart from this all classes have a few outliers. From the Table 4, it is clear that for a good k-value (here, for k = 10), we are getting outliers from different classes and all points of class ERL are classified as outliers.

Table 4 Results for YEAST dataset

Value of K	% Age outliers from different classes									
	CYT	NUC	MIT	ME3	ME2	ME1	EXC	VAC	POX	ERL
K = 2	11.88	17.02	16.39	19.63	33.33	56.82	34.29	16.67	55.00	100.0
K = 4	19.44	16.08	30.74	14.72	13.73	11.36	22.86	13.33	20.00	20.00
K = 10	25.7	27.74	19.26	34.97	50.98	29.55	5.71	26.67	30.00	100.0
K = 20	27.21	31.93	20.08	33.74	43.14	34.09	22.86	36.67	30.00	60.00

5 Conclusion

In this paper, we discussed a unique strategy to identify outliers. We have utilized the clustering K-means algorithm with three closest cluster count and it is then compared with the traditional K-means clustering technique for discovering anomaly utilizing distance based approach. Utilizing and joining these two techniques we propose that the system performance is better than existing frameworks. K-means with three different nearest cluster count detect ideal anomaly contrast with conventional k-means calculation with one cluster count. The exactness of identifying anomalies of our strategy is at par or more than the strategies already existing however we ruled out a portion of the focuses.

References

1. C.C. Aggarwal et al., Fast algorithms for projected clustering, in *ACM SIGMOD Conference Proceedings* (1999)
2. S. Ramaswamy, R. Rastogi, K. Shim, Efficient algorithms for mining outliers from large data sets (2000), pp. 427–438
3. E.M. Knorr, R.T. Ng, Finding intensional knowledge of distance based outliers, in *VLDB 99: Proceedings of the 25th International Conference on Very Large Data Bases* (1999), pp. 211–222
4. E.M. Knorr, R.T. Ng, Algorithms for mining distance based outliers in large datasets, in *Proceedings 24th International Conference Very Large Data Bases* (1998)
5. A. Loureiro, L. Torgo, Outlier detection using clustering methods: a data cleaning application, ed. by C. Soares (2004)
6. R. Pamula, J.K. Deka, S. Nandi, An outlier detection method based on clustering, in *Second International Conference on Emerging Application of Information Technology* (2011)
7. J. James Manoharan, S. Hari Ganesh, J.G.R. Sathiaseelan, Outlier detection using enhanced k-means clustering algorithm and weight based center. App. Int. J. Comput. Sci. Mobile Co, putting **5**(4), 453–464 (2016)
8. K. Zhang, M. Hutter, H. Jin, A new local distance based outlier detection approach for scattered real-world data, in *PAKDD 09: Proceedings of the 13th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining* (2009)
9. M.M. Breunig, H.-P. Kriegel, R.T. Ng, J. Sander, Lof: identifying density-based local outliers. SIGMOD Rec. **29**(2), 93–104 (2000)
10. S. Aggrwal, P. Kaur, Survey of partition based clustering algorithm used for outlier detection. Int. J. Adv. Res. Eng. Technol. **1**(5), 57–62 (2013)
11. T. Zhang, R. Ramakrishnan, M. Livny, Birch: an efficient data clustering method for very large databases. SIGMOD Rec. **25**(2), 103–114 (1996)

FONI by Using Survivability Approach: An Overview

K. V. S. S. S. S. Sairam and Chandra Singh

Abstract Survivability depicts the network protection and restoration with respect to failures (Skoog and von Lehmen in IEEE J Light Wave Technol 22(11):2680–2692, 2015, [1]). The routing of the information in network assortment process determines the digital signal level connectivity by direct and indirect approach (Kavian et al in J Telecommun Inf Technol 68–71, 2016, [2]). This type of architecture is very essential for survivability of digital signal level formations. An integration of digital signal connectivity, protection technique, network assortment in order to represent the systems in queue in order to survive from the failures (Zhou and Subramaniam in IEEE Netw 14:16–23, 2015, [3]).

Keywords RPS · NSA · ONADIM · FUSR · ONRAM · PRS

1 Introduction

The rapid changes in taking place in today communication network which provide high quality and fast services to the users [4]. Optical cross connectivity is measured through digital cross connectivity through digital signal levels via optical carriers. Further the optical cross connectivity is used to measure the bandwidth in fiber optic networks can also evaluate the N × N cross connectivity for different networks. Then whenever the link fails in the network the protection technique is applied and protection ratio is calculated [5]. The protection ratio is the ratio of restoration link

K. V. S. S. S. S. Sairam (✉) · C. Singh (✉)
Department of E&CE, NMAM Institute of Technology, Nitte, Karkala, Karnataka, India
e-mail: drsairam@nitte.edu.in

C. Singh
e-mail: chandrasingh146@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_56

demand is the sum of all individual demands in an given network. The optical network assortment approach in which the hierarchical network which assigns the traffic to circuits in terms of demand signal levels. It relaxes the demands approach with most effective routes. In this network digital cross connectivity, protection technique is applied and protection ratio is calculated followed by network assortment, which gives information about routing information about the routing information from source to destination which finally leads to calculate capacity [6].

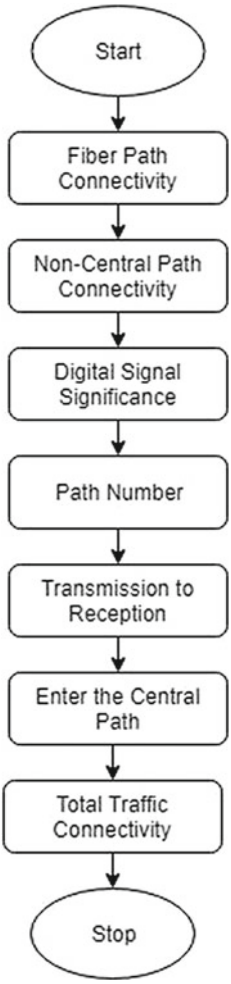


Fig. 1 Fiber demand distribution analysis

2 Problem Statement

A. Fiber Demand Distribution Link Matrix Cost Matrix Demand Total Traffic Connectivity Digital Cross Connectivity.
B. Fiber Network User Survivability Link Matrix Cost Matrix Demand before Failure Demand After Failure Fiber User Survivability Ratio [6].
C. Optical Network Assortment (ONA) Direct Method Indirect Method Hubs Parcel Lists.
D. Fiber Optic Network Integration Traffic In Each Link Fiber Network User Survivability Ratio Optical Network Assortment Direct Indirect Method Optical carrier Capacity [7].

3 Analysis Design and Implementation

The Fiber Network Digital Demand Distribution flowchart is shown in Fig. 1, Total Traffic Connectivity is as shown in Table 1.

The Flowchart of Protection Restoration system (PRS) is shown in Figs. 2 and 3, and PRS output is shown in Table 2.

The Optical Network Assortment Direct and Indirect Method (ONADIM) flowchart as shown in Fig. 4, and output of ONADIM is shown in Table 3.

The Optical Network Integration (ONI) flowchart is shown in Fig. 5.

Table 1 Total traffic connectivity

0	68	0	0	0	0	40	0	0	0	0	0
68	0	42	0	0	62	0	55	0	0	0	0
0	42	0	56	0	0	0	0	0	0	0	0
0	0	56	0	95	0	0	0	84	0	0	0
0	0	0	95	0	78	0	0	0	85	0	0
0	62	0	0	73	0	66	0	0	0	0	0
40	0	0	0	0	0	0	0	0	0	55	0
0	0	0	0	0	0	0	0	0	0	15	46
0	0	0	52	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	44	31	0	0	31
0	0	0	41	0	0	0	0	0	0	0	27
0	0	0	0	15	0	0	0	0	31	27	0

Table 2 PRS ratio

Protection ratio	%Failure before demand after demand
Fiber survivability ratio	175/190 = 92.10%

Table 3 ONADIM output

Demand pair	DS#3
(1, 2)	10
(2, 3)	12
(4, 5)	18
(5, 6)	26
(7, 8)	28
(1, 12)	18
(7, 11)	35
(9, 10)	45
(10, 11)	30
(11, 12)	22
(2, 5)	24
(3, 7)	20
(4, 9)	8
(5, 11)	6

Total Traffic Connectivity for N × N network is calculated which uses the ring architecture. The Total Traffic connectivity in each link is determined first by considering the Fiber Path Connectivity matrix followed by cost matrix. Digital signal levels are taken into consideration to represent the Total Traffic connectivity in different levels through which optical cross connectivity is obtained which is depicted in Table 1.

When the link fails in the network, 1:2DP restores the communication between source to destination by rerouting the information in different routes. PRS ratio is calculated by using restoration link demand to total link demand in the network which is depicted in Table 2.

It deals with Network Assortment, where the routing takes place through direct and indirect path. Here the information will be divided into the parcel list with respect to each hub in a given network topology.

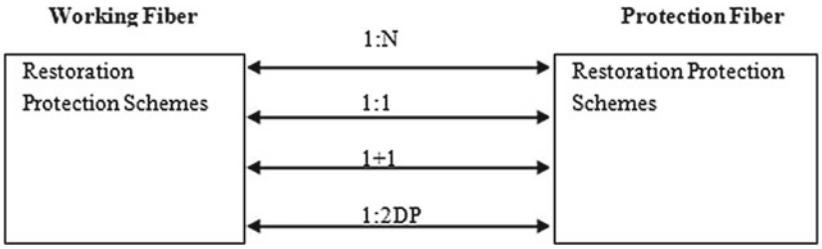


Fig. 2 Restoration protection schemes (RPS)

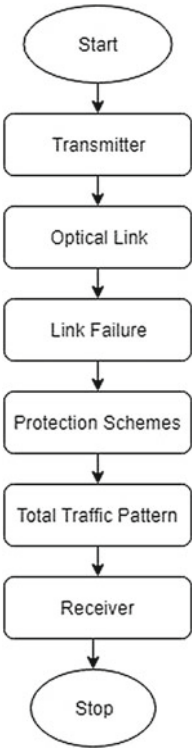


Fig. 3 Flowchart of protection restoration system (PRS)

In this approach for N × N Network the total traffic connectivity, 1:2DP and how the routing takes place with direct and indirect paths is calculated. The capacity for each node is also considered and is arranged in Queue in Increasing Order which is depicted in Fig. 6.

4 Conclusion

For N × N Network total traffic pattern, protection ratio is calculated for end to end connectivity. The routing takes place determined by considering direct and indirect path (ONADIM). The capacity for each node is arranged in ascending order so that the node with shortest capacity is preferred first for processing and further it can be enhanced by using Optical layer approach where the packets will send the information without leaving the optical domain.

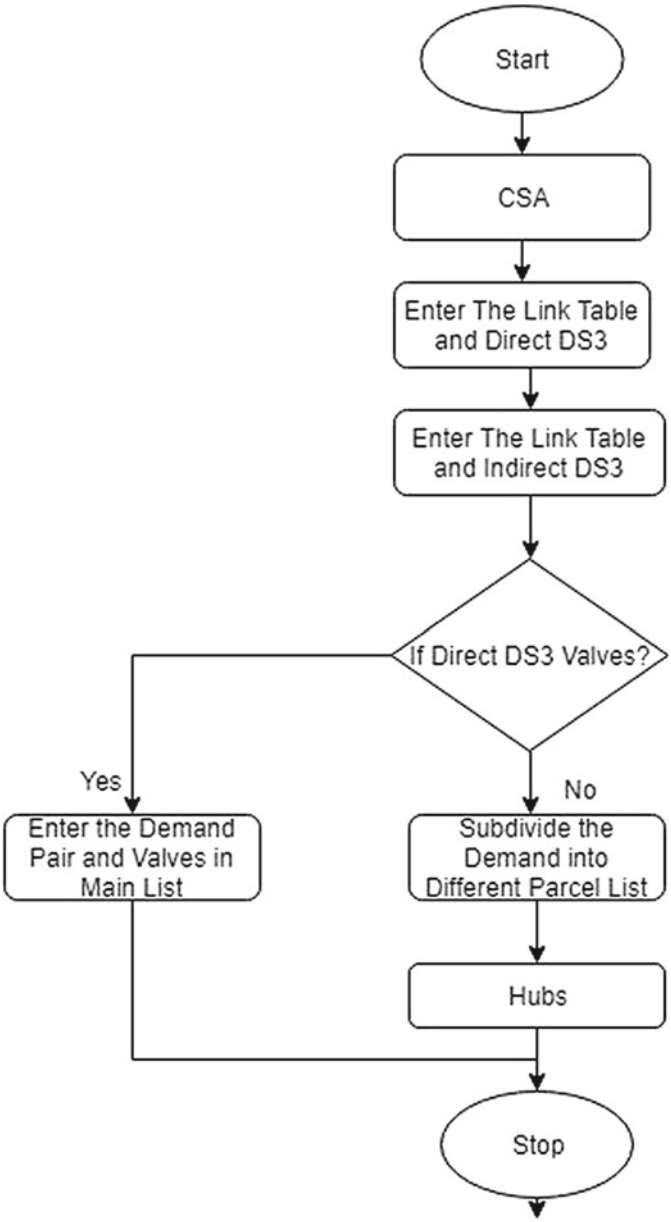


Fig. 4 ONADIM flowchart

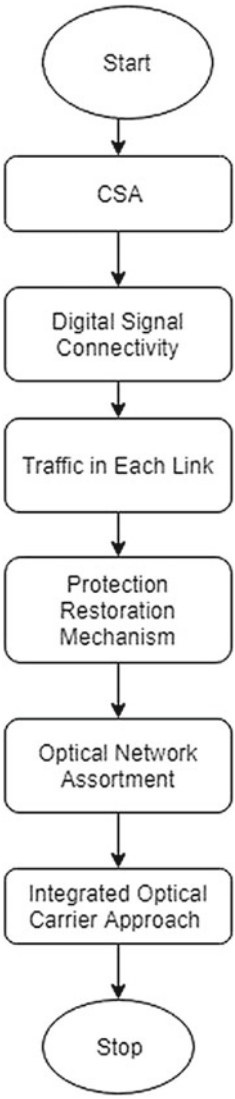


Fig. 5 ONI flowchart

Fig. 6 ONI output

Capacity of node 1: 110

Capacity of node 2: 190

Capacity of node 3: 150

Capacity of node 4: 170

Capacity of node 5: 210

Capacity of node 6: 230

Capacity of node 7: 140

Capacity of node 8: 240

Capacity of node 9: 175

Capacity of node 10: 100

Capacity of node 11: 130

Capacity of node 12: 250

The capacity Arranged in Queue in Ascending order As follows

100,110,130,140,150,170,175,190,210,230,240,250

References

1. R. Skoog, A. von Lehmen, Metro network design methodologies and demands. IEEE J. Light Wave Technol. **22**(11), 2680–2692 (2015)
2. Y.S. Kavian, H.F. Rashvand, M.S. Leeson, W. Ren, E.L. Hines, M. Naderi, Network topology effect on QoS delivering in survivable DWDM optical networks. J. Telecommun. Inf. Technol. 68–71 (2016)
3. D. Zhou, S. Subramaniam, Survivability in optical networks. IEEE Netw. **14**, 16–23 (2015)
4. J. Strand, A. Chiu, R. Tkach, Issues for routing in the optical layer. IEEE Commun. Mag. **39**, 81–87 (2013)
5. H. Zang, C. Ou, B. Mukherjee, Path-protection routing and wavelength assignment (RWA) in WDM mesh networks under ductlayer constraints. IEEE-ACM Trans. Netw. **11**(2), 248–258 (2015)
6. X. Shao, L. Zhou, X. Cheng, W. Zheng, Y. Wang, Best effort shared risk link group (SRLG) failure protection in WDM networks, in *Proceedings of IEEE International Conference on Communication* (2015), pp. 5150–5154
7. K.V.S.S.S.S. Sairam, C. Singh et al. Optical network survivability: an overview. Indian J. Sci. Res. **14** (2), 383–386 (2017)

An Optimized Architecture for Unpaired Image-to-Image Translation

Mohan Nikam

Abstract Unpaired image-to-image translation was aimed to convert the image from one domain (input domain A) to another domain (target domain B), without providing paired examples for the training. The state-of-the-art Cycle-GAN demonstrated the power of generative adversarial networks with cycle consistency loss. While its results are promising, there is scope for optimization in the training process. This paper introduces a new neural network architecture, which only learns the translation from domain A to B and eliminates the need for reverse mapping (B to A), by introducing a new Deviation loss term. Furthermore, few other improvements in the Cycle-GAN are found and utilized in this new architecture, contributing to significantly lesser training duration.

Keywords Artificial intelligence · Computer vision · Image processing
Neural networks · Unsupervised learning

1 Introduction

Transferring characteristics from one image to another is at the core of image-to-image translation. The data are said to be unpaired, when there is no one-to-one correspondence between training images from input domain A and the target domain B, for example, converting an image of apple to an image of orange or image of a horse into an image of a zebra. The challenge is also to keep the background of the image intact. Everything except the apple (in case of conversion from apple to orange) must not get altered. Therefore, unpaired image-to-image translation becomes an even difficult problem than having paired examples. However, there are many scenarios where such translation is necessary. Also, collecting unpaired data is much easier than deriving a paired set of images. The paper "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks", famously known as Cycle-GAN by

M. Nikam (✉)
Medi-Caps Institute of Science and Technology, Indore, India
e-mail: mohannikam19@gmail.com

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_57

Park et al. [1] offers an approach for creating such a system. In order to drive the learning process for creating such a system, they introduced cycle consistency loss, which involves not only learning the mapping for conversion A to B, but also the transitive mapping (B to A). This works very well for unpaired translation in many kinds of images. However, the learning of reverse mapping turns out to be a redundant step which takes equal amount of computational time to train, as it is consumed for training the translation A to B.

This paper proposes a new architecture, which learns only the forward mapping (from domain A to B) and eliminates the need of backward mapping (from domain B to A) while keeping the goodness achieved by this backward mapping (directing the training to proceed in right direction), by introducing a new loss term, deviation loss. The training duration for this architecture is found to achieve a significant speedup as compared with Cycle-GAN.

2 Related Work

Cycle-GAN. Unpaired Image-to-Image Translation Cycle-Consistent Adversarial Networks by Park et al. [1] provided an approach for this problem using cycle consistency loss. Further comparison with their approach is presented in next section.

Generative Adversarial Network (GAN). This paper by Pouget-Abadie et al. [2] involves a generator and a discriminator optimizing against each other in a 2-player zero-sum game. Generator tries to fool the discriminator and discriminator tries to correctly classify a real image and a fake image, coming from the output of generator.

Autoencoders. It involves two neural networks: One is used to encode an image and another is used to decode it. Purpose of using autoencoders [3] is for dimensionality reduction or simply compression of images. In this paper, convolutional autoencoder is used for sandwiching between translator (discussed in next section). In convolutional autoencoder, encoder consists of convolutional neural network [4–6], instead of fully connected neural network, which is more effective for encoding image. Likewise, for decoding the image, deconvolution is performed, to bring the same image from the encoded image.

Neural Artistic Style Transfer. This work [7] and its successors [8, 9] deals with transferring artistic characteristics from one image (preferably painting) to another image (a photograph). For doing so, they proposed extracting style of an image by taking feature encodings from the intermediate layers of convolutional neural network and pretrained to encode the features effectively and obtaining its gram matrix, and combining this extracted style with the extracted content, by getting the feature encodings from lower layers of the same convolutional neural network.

Disco-GAN. This work [10] is similar to Cycle-GAN for unpaired image translation, with difference that their approach involves two reconstruction losses instead of one single cycle consistency loss and different measures for distance between the images. However, their approach too involves learning the reverse mapping (B to A).

Conditional-GAN. Image-to-Image translation with Conditional Adversarial Networks by Zhu et al. [11] have proposed an approach for image-to-image translation, but by using paired examples for the training process. Their approach involved using Generative Adversarial Networks [2] in conditional setting.

3 Proposed Approach

There are four different neural networks; each would perform a specific function after the completion of training process as follows (Fig. 1):

i. *Encoder*: This network encodes the input image from any domain (A or B), by performing convolution [4–6] operations, to downsample the image. This is different from encoder in Cycle-GAN, which used separate encoders for A and B.

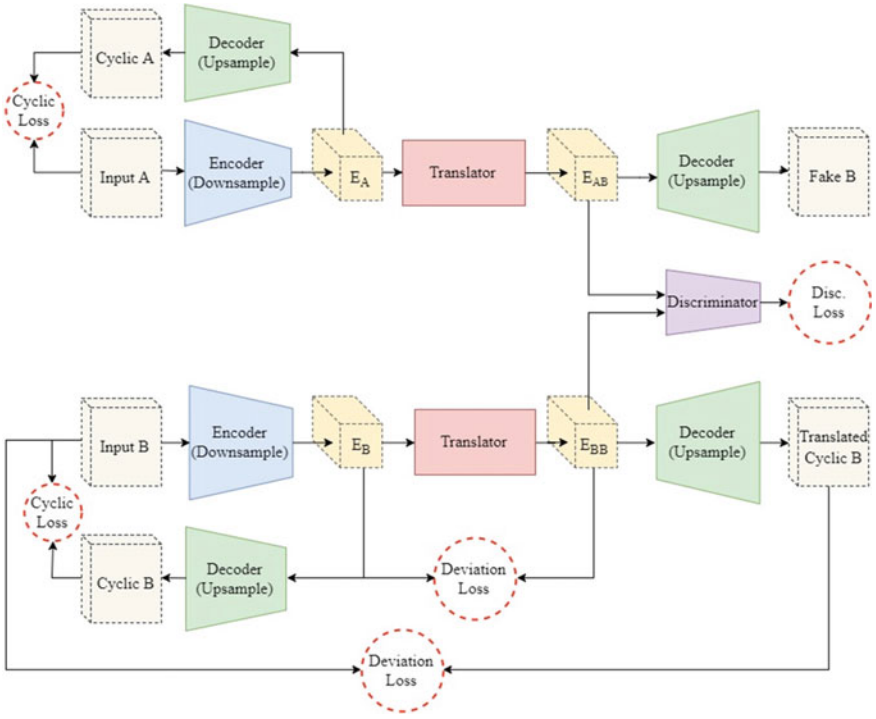


Fig. 1 Proposed architecture. The aim is to obtain image *Fake B*, from the image *Input A*. Neural networks are represented by solid boundaries and those having same color represents same network. The orange colored circles indicate *loss terms*

ii. *Decoder*: This network learns to decode the encoded image from any domain (A or B), by performing deconvolution operation, to upsample the encoded image. This is different from Cycle-GAN as well, since it used separate decoders for A and B.
iii. *Translator*: It is created using residual networks [12]. It inputs the encoded features from encoder (of both domains A or B) and ensures that output is only encoding from domain B, i.e., it selectively translates the encodings from domain A, while allows images from domain B to be passed unmodified. Thus, only one translator is used here, whereas in Cycle-GAN, there were two translators. (They call it generators A2B and B2A).
iv. *Discriminator*: This network inputs the encoded image and identifies whether the encoding is real or fake. This is different from Cycle-GAN, as its discriminator takes in the whole image and not its encodings (Figs. 2 and 3).

Following are the loss functions used to train and optimize the above-mentioned neural networks:

i. *Cyclic Loss*: This loss is for encoder and decoder networks. It works the same way as for autoencoders. The loss term is defined as the difference between the input image and the cyclic image (which is obtained by encoding and then decoding back the input image).

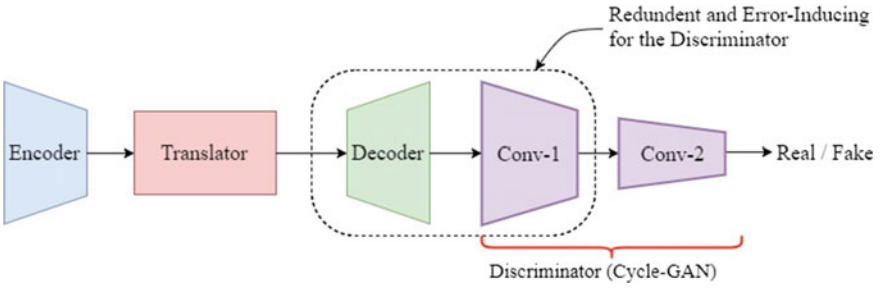


Fig. 2 Cycle-GAN upsamples the translated encoding and discriminator needs to again downsample it to the same level *(Conv-1)* and further convolute *(Conv-2)* for classification

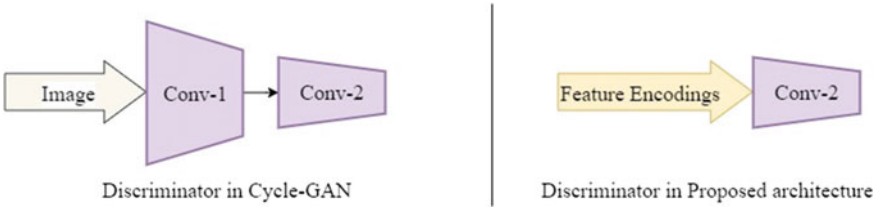


Fig. 3 Cycle-GAN discriminator versus discriminator in proposed architecture, which takes downsampled encodings as input as opposed to whole image. This results in lesser complex discriminator and still not compromising in the accuracy

ii. *Deviation Loss*: This loss is for the translator network. It is defined as the sum of following two terms:

   a. Difference between the encodings of image from domain B, before and after passing through the translator.
   b. Difference between the input image from domain B and the image obtained by encoding, translating, and then decoding the image (translated cycle B).[1]

This loss is introduced as a replacement for the cycle consistency loss which was present in the Cycle-GAN architecture.

The deviation loss regularizes the training of translator, by directing the translator to translate only the bare-minimum part of encoded image of domain A, to make it appear like real encoding from domain B. Also, it enforces the spatial features to be kept intact throughout the translation.

iii. *Discriminator Losses*: These losses are for optimization of translator network and the discriminator Network. If the discriminator correctly classifies the encodings of the fake image, the translator network is penalized. If the discriminator incorrectly classifies either of real or fake encodings, the discriminator network is penalized. This loss is kept similar to that of Cycle-GAN, but the structure of discriminator has changed in the proposed architecture.

Explanation. Considering an example for converting the image of an apple to an orange, the goal is to perform this task while keeping the background intact. Forward pass involves downsampling of the input image of an apple, translating it to the encoding of orange, and upsampling it, to produce the image of an orange. Deviation loss ensures the output of translator is always the feature encodings of orange. Thus, the image of orange is unchanged (including background), whereas the image of apple changes in such a way that apples are converted into oranges (since discriminator network is forcing this conversion) while everything in the background is unchanged (since deviation loss is resisting this change). The key idea is that translator network learns not to alter the background and orange but to necessarily convert the apple to orange.

4 Experimental Evaluation

The performance of this architecture is compared with the Cycle-GAN implementation, on Tensorflow Framework, on Intel® AI DevCloud using Intel Xeon Gold 6128 processors (Table 1).

Furthermore, it is observed that due to using same neural network for encoding and decoding, and also using less complex Decoder, the proposed system converges nearly twice as fast, i.e., it needs nearly half the number of epochs required by Cycle-GAN to produce the same result.

[1] This term is yet to be tested for importance. However, it does not hurt the performance.

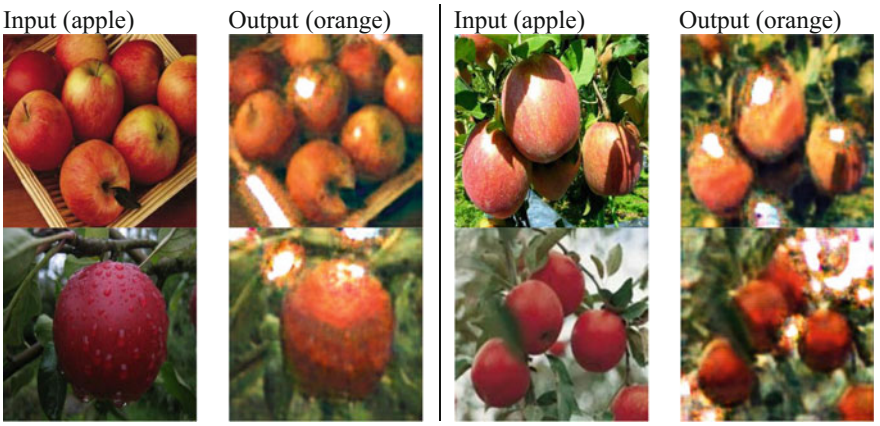
Table 1 Comparison of time taken by Cycle-GAN and proposed architecture

No. of epoch(s)	Time by Cycle-GAN (min)	Time by proposed architecture (min)	Speedup
1	66.27	32.92	2.0128x
2	132.54	65.84	2.0130x
3	198.81	98.76	2.0138x
15	994.09	493.80	2.0131x

Thus, net effective speedup could reach four times the Cycle-GAN. The work on this is still ongoing to present wider variety of results.

5 Result and Discussion

The neural networks were trained on images of apple and oranges collected from ImageNet and were directly available from [13]. The images were of the dimension 256×256 pixels. The training set consists of 1177 images of class apple and 996 images of class orange.



The important aspects of the proposed architecture are:

1. Elimination of second translator (to translate B to A).
2. Using same neural network to encode images from both domains (A or B), and same neural network to decode images from both domains (A or B).
3. Discriminator takes downsampled image encoding as input, as opposed to taking whole image which was the case with discriminator in Cycle-GAN.

Use of deviation loss, in lieu of cycle consistency loss, from Cycle-GAN.

6 Conclusion

This paper introduces a new architecture for unpaired image-to-image translation.

The proposed architecture is able to simplify the architecture of state-of-the-art Cycle-GAN and hereby achievie experimental speedup of 2.01x for the training process, which under subjective consideration for convergence can be as much as 4x.

Acknowledgements I would like to give my sincere thanks to Prof. Dheeraj Rane, HOD, CSE, Medi-Caps University, for his invaluable support and constant encouragement. The computational resources for this work were provided by Intel Corporation on Intel AI DevCloud under the Student Ambassador for AI program. The work was also partly supported by Amazon for AWS EC2 instances through their student scholarship.

References

1. T. Park, P. Isola, A.A. Efros, J.-Y. Zhu, Unpaired image-to-image translation using cycle-consistent adversarial networks, https://arxiv.org/abs/1703.10593
2. J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair. A. Courville, Y. Bengio, I.J. Goodfellow, Generative Adversarial Networks, https://arxiv.org/abs/1406.2661
3. Autoencoders—Chapter 14, the deep learning book, Ian J. Goodfellow, http://www.deeplearningbook.org/contents/autoencoders.html
4. K. Simonyan, A Zisserman, Very deep convolutional networks for large-scale image recognition, https://arxiv.org/abs/1409.1556
5. I. Sutskever, G.E. Hinton, A. Krizhevsky, Imagenet classification with deep convolutional neural networks. NIPS (2012)
6. K. Fukushima, Neocognitron: a self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biol. Cybern. (1980)
7. A.S. Ecker, A.M. Bethge, L.A. Gatys, A neural algorithm of artistic style, arXiv:1508.06576v2, https://arxiv.org/abs/1508.06576v2
8. M. Bethge, A. Hertzmann, E. Shechtman, L.A. Gatys, Preserving color in neural artistic style transfer, https://arxiv.org/abs/1606.05897
9. Y. Nikulin, R. Novak, Improving the neural algorithm of artistic style, https://arxiv.org/abs/1605.04603
10. M. Cha, H. Kim, J.K. Lee, J. Kim, T. Kim, Learning to discover cross-domain relations with generative adversarial networks, https://arxiv.org/abs/1703.05192
11. J.Y. Zhu, T. Zhou, A.A. Efros, P. Isola, Image-to-image translation with conditional adversarial networks, https://arxiv.org/abs/1611.07004
12. X. Zhang, S. Ren, J. Sun, K. He, Deep residual learning for image recognition, https://arxiv.org/pdf/1512.03385.pdf
13. T. Park, UC Berkeley, apple2orange Dataset, Cycle-GAN. https://people.eecs.berkeley.edu/~taesung_park/CycleGAN/datasets/apple2orange.zip

Performance Evaluation of Adaptive Shrinkage Functions for Image Denoising

Ajay Kumar Boyat and Brijendra Kumar Joshi

Abstract Denoising and compression is performed on natural image. In this paper, we compared the performance of various adaptive shrinkage functions. SURE shrinkage, Universal shrinkage and Bayesian shrinkage functions are analyzed in many recent algorithms. In this paper, we have used heursure, rigrsure, sqtwolog and minimax functions and also evaluated their performance in wavelet sub-band such as approximation and detail coefficients. The work also proposed a new scaled shrinkage or threshold function in Bayesian frame. Monarch image was taken in account as Gaussian contaminated image; generally, Gaussian noise is found in electronic hardware. We mainly worked on the performance evaluation of various adaptive shrinkage functions for higher peak signal-to-noise ratio (PSNR).

Keywords SURE shrinkage · Universal shrinkage · Bayesian shrinkage functions PSNR

1 Introduction

An excess of denoising algorithms has been developed using adaptive shrinkage functions, of which the inverse filtering, pseudo-inverse filtering, wiener filtering and constraint least square filter are most popular, but these are very old algorithms. Linear filtering is easier and faster. It has very limited capabilities making it off, nowadays. Adaptive shrinkage functions in wavelet domain have been found effective around the world. In image denoising studies, we have been found various adaptive shrinkage approaches based on wavelet domain. However, we also mentioned that transform domain denoising methods were developed based on few coefficients and that preserves special information of image such as edges. Transform domain method is prime method rather than equivalent spatial domain method.

A. K. Boyat (✉) · B. K. Joshi
Military College of Telecommunication Engineering, DAVV, Mhow, Indore, India
e-mail: a_boyat@yahoo.co.in

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_58

2 Proposed Method

The wavelet transform commonly applied in image compression. However, single wavelet does not fulfill all the desired traits. Thus, multi-wavelet transform is introduced that overcome the limitations of single wavelet. It provides bi-orthogonal property that requires in multiresolution analysis. We differentiate multi-wavelet with single wavelet by using doubly scaling and wavelet functions.

In the notion of multi-wavelet, multi-scale function is given as:

$$\Phi(t) = [\phi_1(t), \phi_2(t), \phi_3(t), \ldots \phi_n(t)]^T \tag{1}$$

And multi-wavelet function is also likewise given as:

$$\Psi(t) = [\psi_1(t), \psi_2(t), \psi_3(t), \ldots \psi_n(t)]^T \tag{2}$$

If n $=$ 1 then the function is single wavelet function and if n $\geq$ 2 the above function is commonly referred to as multi-wavelet function. The multi-scale and multi-wavelet are given as:

$$\Phi(t) = \sqrt{2} \sum_{n=-\infty}^{\infty} H_n \phi(2t - n) \tag{3}$$

$$\Psi(t) = \sqrt{2} \sum_{n=-\infty}^{\infty} G_k \phi(2t - k) \tag{4}$$

The above two-scale equations are referred to as analysis and synthesis filters, respectively, that resemble the perfect reconstruction bank. It also possesses bi-orthogonal symmetry and approximation at higher-order derivatives. Single wavelet provides bi-channel filter bank, whereas multi-wavelet provides multi-channel filter bank that reflects extra degree of freedom. Multi-wavelet construction is an alternative to the single wavelet but does not show quantization as it happened in single wavelet [1]. Following subsections described the proposed method in adaptive shrinkage aspects.

2.1 Stein's Unbiased Risk Estimator (SURE)

SURE is found effective for estimation of the risk or error without any bias is given as:

$$\left\| \hat{\mu}_i - \mu_i \right\|^2 \tag{5}$$

In multivariate observations, reference signal $x_i, i = 1, 2, \ldots d$ follows normal distribution that is $N(x_i, 1)$ and mean is $\mu_i, i = 1, 2, \ldots d$. The method is analyzing the Stein's result of [1] to estimate the risk in unbiased fashion.

$$\hat{\mu}_i^t(x) = \eta_{(t)} x_i \cdot E\left\|\hat{\mu}_i^t(x) - \hat{\mu}_i\right\|^2. \tag{6}$$

2.2 Bayes Shrinkage Function

Bayes shrink function involves finding out a unique shrinkage for every sub-band and better then SURE shrinkage function. The parameter σ_x which is the standard deviation shows how dense the spread function is, while β represents the shape parameter. However, wavelet coefficients are well approximated and effectively produced the empirical values of shape parameter β and standard deviation σ_x in each sub-band. Consequently, we try to found the accurate and best value of threshold T for which Bayesian risk follows least square norms. The expected mean square error in case of Bayesian paradigm is given as:

$$T_{B(\sigma_X, \beta)} = \arg \min_T \tau(T) \tag{7}$$

The above shrinkage function heavily dependents on σ_X and β. However, lack of closed solution, we use numerical methods to find parametric value of T_B. It can be observed that the threshold value given by $T_B(\sigma_X) = \frac{\sigma^2}{\sigma_X}$ using Eq. (7) is very close to T_B. Thus, estimation precise and has further benefits.

3 Simulation Results

Recently, most methods have been focused on Bayesian estimation. The data driven and sub-band adaptive shrinkage is used in Bayesian method. Gaussian, Laplacian and mixture of Gaussian priors have been implemented in Bayesian paradigm [1–5].

The wavelet coefficients exist in various sub-bands such as LL_k, LH_k, HL_k and HH_k at scale k= 1, 2, 3, …, where LL_k is referred to as approximation coefficients and LL_k, LH_k, HL_k and HH_k are referred to as details coefficients. Time-spectral relationship is deployed using approximation and detail coefficients. These coefficients are further improved via adaptive shrinkage function in order to accelerate the smooth and sharp coat (Fig. 1).

$$g(j, k) = f(j, k) + \hat{\sigma}_n z(j, k) \tag{8}$$

From Eq. (9) through Eq. (10) suggests noise standard deviation $\hat{\sigma}_n$ is to be locally fitted. A $\hat{\sigma}_n$ was found using median absolute division (MAD) for level dependent

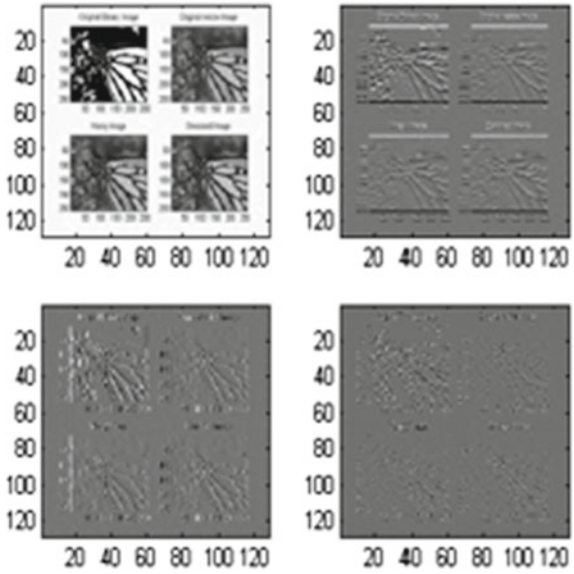


Fig. 1 Wavelet decomposition using various sub-band

diagonal detail sub-band. Further development is considered in detail components in [1–5], which is given in Eq. (9).

$$\hat{\sigma}_n = \frac{median|HH_k|}{0.6745} \tag{9}$$

where HH_k diagonal detail sub-bands are used at scale $k = 2$ in this work, beyond this scale no improvement was observed. Noisy characteristics are modeled as zero mean, and noise variance is σ_G^2, variance of unknown original image is $\sigma_{F_x}^2$ and σ_n^2 is variance of Gaussian noise contaminated image is estimated by Eq. (8).

$$\sigma_G^2 = \sigma_{F_x}^2 + \hat{\sigma}_n^2. \tag{10}$$

$$\sigma_G^2 = \sigma_Y^2. \tag{11}$$

Thus, Bayesian shrinkage function is

$$T_B = \frac{\sigma_n^2}{\sigma_{F_x}}. \tag{12}$$

where (Fig. 2 and Table 1)

$$\sigma_{F_x}^2 = \sqrt{\max(\hat{\sigma}_Y^2 - \hat{\sigma}_n^2), 0} \tag{13}$$

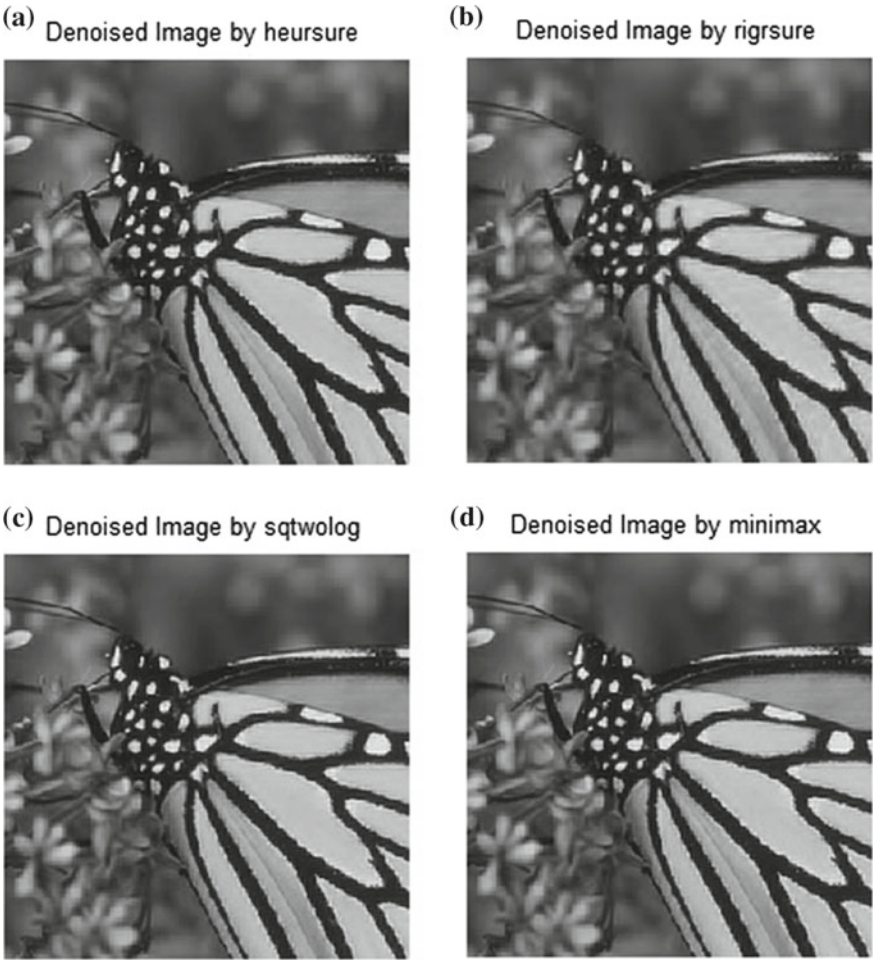


Fig. 2 Denoised monarch image by various SURE shrinkage functions

Table 1 Obtained PSNR at low and high range of variance σ^2

Shrinkage function	PSNR at $\sigma^2 = 2$ (dB)	PSNR at $\sigma^2 = 20$ (dB)
heursure	37.88	22.08
rigrsure	32.33	22.08
sqtwolog	37.88	23.45
minimax	39.79	23.06

4 Conclusion

Wavelet threshold-based filters are used in majority to solve the noise problems for natural images. Thus, we deploy the wavelet decomposition using scale coefficients. We also select the spatial adaptive threshold in Bayesian paradigm. Spatial adaptive threshold concept is used in current time in many algorithms. To wash the effect of blur and accurate enhancement of the edges is the main tenet of the new adaptive algorithm in this work. The above considerations are thoroughly described in the paper. In this work, we discussed how degraded function was used and effectively constructed the restored image using inverse filtering in wavelet domains. The proposed method provides the selection criterion for effective estimation of adaptive shrinkage functions. Finally, we proved that Bayesian shrinkage is better than the SURE shrinkage.

References

1. M. Jansen, *Noise Reduction by Wavelet Thresholding* (Springer-Verlag, New York, 2001)
2. E.P. Simoncelli, E.H. Adelson, Noise removal via Bayesian wavelet coring, in *Proceedings of IEEE International Conference on Image Processing*, Lausanne, Switzerland, vol. 1, Sept 1996, pp. 379–382
3. D.L. Donoho, I.M. Johnstone, Ideal spatial adaptation via wavelet shrinkage. Biometrika **3**, 192–206, Mar 1994, 5 Sep 1994
4. D.L. Donoho, I.M. Johnstone, Wavelet shrinkage: asymptopia? J. R. Stat. Soc. **57**(2), 301369 (1995)
5. D.L. Donoho, Denoising by soft-thresholding. IEEE Trans. Inf. Theory **41**(3), 613–627 (1995)

Fusions of Palm Print with Palm-Phalanges Print and Palm Geometry

Himanshu Purohit and Pawan K. Ajmera

Abstract Multimodal biometrics is an advanced technology and core of machine learning and mankind machine interaction. It is based on different physical traits for person identification. In existing commercial biometric systems, most are based on processing of a single trait for recognition, which is known as unimodal system. Drawbacks of these systems are lack of similarities among same class, limited degrees of freedom, imposter attacks, and absence of universality. Some of these issues are handled by fusion of features of more than one trait for person recognition. This paper presents a brief review on past research and fusion of palm print, dorsal and hand geometry-based multimodal biometric system.

Keywords Unimodal · Multimodal · Fusion · Palm print · Dorsal

1 Introduction

Biometrics is the combination of bio (life) and metric (to measure), used for person authentication in next-generation devices and application. Now day's biometric authentication is crucial in making interface for interactions between person–person and man–machine. In biometric system, access to service is controlled by identification process. Any physical or behavioral trait can be a biometric modality if it fulfills the criteria such as availability among all users, distinctiveness, stability over time frame, and easy to collect from users. The most common physical traits for identification are: face, iris, fingerprints, palm prints, and hand vein and hand geometry. The soft biometric are behavioral features like gait, voice, signature and keyboard stroke, or typing patterns [1].

H. Purohit (✉) · P. K. Ajmera
EEE Department, BITS Pilani, Pilani, India
e-mail: p2015502@pilani.bits-pilani.ac.in; shalabh227@gmail.com

P. K. Ajmera
e-mail: pawan.ajmera@pilani.bits-pilani.ac.in

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8_59

The biometric system functions in two steps:

- Enrollment step: The biometric data of user is fetched by a sensor, and processed data template is stored in a database. It can be positive enrollment or negative enrollment.
- Authentication step: by some methods, user claim identity (1–n) or system verifies (1–1) the claimed user identity against stored templates.

2 Types of Biometric System

The biometric system can be classified in various categories such as multisensor, multi-instances, multimodal, multi-algorithm, and multipresentation. It can be define as basic unimodal or fusion-based multimodal systems.

2.1 Unimodal Biometric Systems

The unimodal biometric system (UBS) uses only one type of biometric trait for authentication. Following are the main challenges faced by this mode: corrupt data, lack of universality, intraclass variation, similarities between classes, and imposter attacks. In UBS, main blocks are user interface, data acquisition, matching module, and decision-making module. In most cases, the UBS causes comparatively *higher* false acceptance rate (FAR) and false rejection rate (FRR) [2].

2.2 Multimodal Biometric System

The multimodal biometric system (MBS) is fusion-based model where features taken from multiple traits such as finger, palm, iris, and ear are fused together. The combination of these input vectors is fused at different levels of system design for better accuracy and low FAR. The fusion-based systems can be categories as pre-matching fusion and post-matching fusion models. The feature-level and sensor-level designs are *fusion before matching*-based system, whereas score, rank, and decision are *fusion after matching*-level systems [3] (Fig. 1).

The fusion is a process where mixing of different preprocessed extracted information feature vectors from multiple input traits takes place. It is important to determine that which types of information may be fused together and at which level. Jain et al. have proposed to consider soft indexes such as sex, origin, physical dimension, and eye color into recognition process [3].

- Sensor level: In this level, output of different sensor is directly concatenated for fusion. This is most basic level of fusion and less preferred.

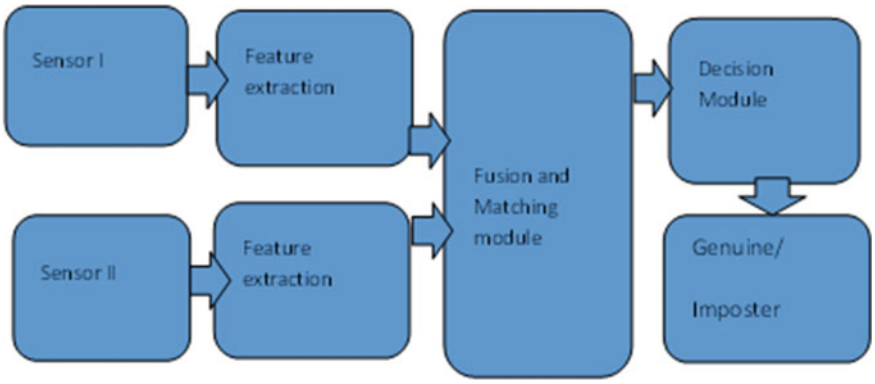


Fig. 1 Multimodal biometric architecture

- Feature level: In this method, after initial processing, features are extracted separately from each biometric trait and arranged in vector form. These vectors are concatenated to form a common feature vector which is used for authentication process. Principal component analysis is the most common technique used at this level of fusion [2].
- Score level: In this case, features are processed separately and compared with stored database to generate individual score. All these scores are combined to make final decision. There are several techniques such as min–max, average, highest score method which are used to calculate scores [3].
- Decision level: After independent pre-classification of each modality, final decision is made by some logic gate application or mathematical criteria.

3 Related Work

In recent years, palm-related feature fusion-based system multimodal design has attracted researchers. The MBS can operate in different integration scenarios like multiple sensors, multiple biometrics, multiple instances, and multiple algorithms for same biometrics, etc. [4]. Ross et al. has tested feature-level fusion scheme on face and hand geometry, which are considered as weaker biometric traits [5]. For FAR close to 0.01%, the GAR has been found to improve from 50 to 65%.

M. Haghighat et al. 2016 have proposed class-based discriminant correlation analysis (DCA) of feature level and experimentally proved that it is better than canonical correlation analysis method [6]. The less used method of Mahalanobis distance technique was used for novel fusion algorithm design and support vector machine algorithm utilized for system training purpose. The design was tested on CASIA iris database and real fingerprint database [7].

Table 1 Summary of work done in MBS system design [14]

Sr. no.	Combinations	Authors	Fusion level
1	Speech + Face	Poh	Score
2	Speech + Face	Kitter	Score
3	Fingerprint + Signature	Ching	Score
4	Face + Palm print	Feng	Feature
5	Face + Iris	Rose	Score
6	Signature + Iris	Wang	Feature
7	Face + Fingerprints	Jing	Feature
8	Fingerprint + Face + Hand geometry	Nandkumar	Score
9	2D Face + 3D Ear	Mahoor	Score
10	Face + Finger veins	Imran	Score
11	Face + Palm print	Linlin	Score
12	Face + Eye	Kawuolak	Feature
13	Iris + Fingerprint	Ujwalla	Feature
14	ECG + Speech	Bugdol	Feature
15	Fingerprint + Face	Ghate	Score
16	Iris + Palm print + Face	Ren-He	Feature
17	Multiple combination	Saif	Feature
18	Fingerprint + Iris	Jain	Score

Recently, Saif et al. in 2017 have presented innovative combination of Gaussian copula and QR codes to overcome error rate problem and security issue [8]. In a short review, Celik in 2017 has discussed the work done by Haghighat et al. in brief [9]. A wearable biometric system has been developed by Uhl et al. and that is worn on the arm, and authentication is done once the user is recognized by the system [10]. In their work, Xiaona et al. have proposed an improved feature-level fusion algorithm based on kernel canonical correlation analysis (KCCA) and tested it for ear and face biometrics. The experimental results were found to be better than unimodal system [11]. In the year 2016, Muhatahir et al. have presented state of the art survey on biometric sensing systems [12]. This survey shows that less work has been reported in palm print, dorsal hand vein, and hand geometry combination. It is in chronological order, starting from 2001 to 2017 (Table 1).

4 Palm-Phalanges with Palm Print and Palm Geometry

The palm-phalanges joint due to its uncommon pattern can be utilized in person authentication. The skin in center of fingers contains folding that is known as phalanges. There are ridges in this area to increase friction and multiple parallel lines which reflects unique information about individual.

A survey on aging effect on biometric traits is presented in [13]. As per that it is assumed that stability of phalanges is like palm print only. Here, in this work, we have taken palm print phalange with hand geometry and palm print together. Hand-based biometric are better in stability and taking samples is also convenient for users.

4.1 Palm-Phalanges Feature Extraction

In phalanges feature extraction, posture of all input fingers must be identical so that same details can be captured from joints [4]. As key points, coordinates of all fingertips and valleys between fingers are selected, and finally the centroid from each processed image is extracted. So that hand image could be rotated to get the line joining the tips of the ring, and index finger is in same plane.

For image enhancement, Rayleigh distribution-based adaptive histogram equalization (AHE) is used on selected ROI. Post initial processing, improved ROI is subdivided into non-overlapping units or windows of specific size. In total, every region of interest has contributed 100 windows. Then Gaussian membership function (GMF)-based features of palm-phalanges (size 100) are extracted from each window. The mean and average absolute deviation (AAD) are also used to extract features with (GMF). In this work, mean features have been taken for further processing.

4.2 Palm Print Feature Extraction

Similar procedure is applied for palm print feature extraction. Finger valley points are used for ROI selection, and later ADE is used for enhancement. The enhanced ROI is partitioned into 100 non-overlapping windows. Then GMF, mean, and AAD feature were extracted. For classification, Random Forest method is used.

4.3 Palm Geometry Feature Extraction

The palm print images are used for palm geometry. First of all ROI identification and enhancement of palm print is done. From that height and width features of palm

print are taken. For classification, test image is compared with database, and random forest classifier is used.

4.4 Score-Level Fusion

All the scores generated for palm print, palm-phalanges and palm geometry are fused using simple sum rule. Before fusion, min–max normalization method is applied so that all features can be transformed to common domain.

5 Experimental Flow and Results

- Database

We have used real-time standard database of limited size. The testing with other open access database is the task of future experiments.

- Preprocessing

In this step, hand sample was made straight using position of fingertips and position of centroid. The ROI was selected using finger valleys of corrected hand samples.

- Feature extraction process, methods of classification, and final fusion (Figs. 2 and 3).

AHE was applied on each ROI. The sequence of different extraction process was as follows, first GMF, and then AAD and mean features were taken. For classification KNN, SVM, and Random Forest were used. At last, sum rule of score-level fusion is used to calculate final score. The uniqueness of palm-phalanges and geometry is

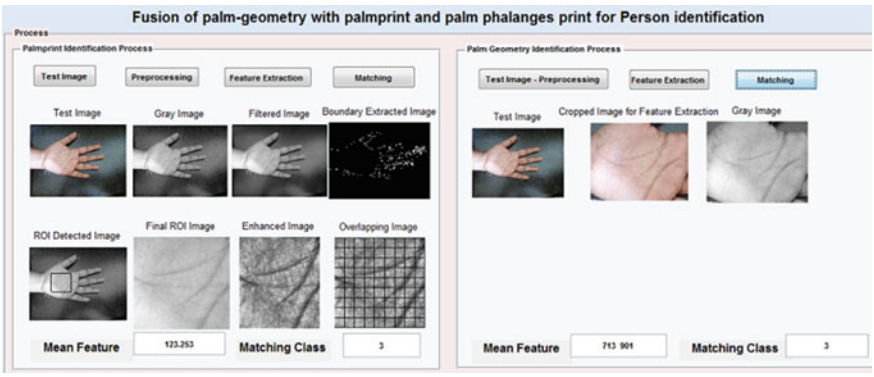


Fig. 2 Palm print and Palm Geometry feature extraction

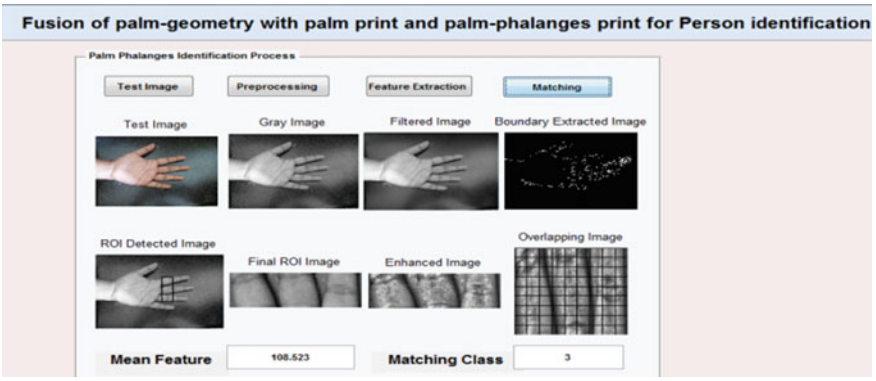


Fig. 3 Palm-phalanges feature extraction

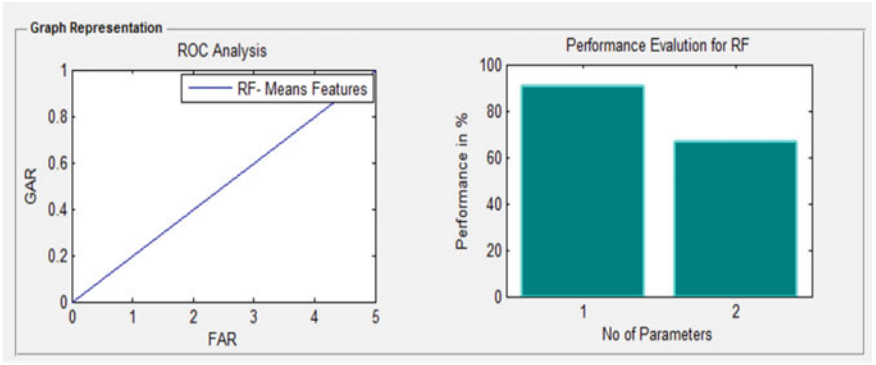


Fig. 4 ROC and performance evaluation

the main factor in performance. Fusion at feature level with DCA can be good path for real-time implementation (Fig. 4).

Fusion with AAD and GMF are also suitable in specific application. So we propose DCA implementation at feature level and real-time hardware implementation of shared method. Vertex 4/5 of Xilinx or DSK6713 can be used as base platform for hardware realization.

In experimental result, overall 90.30% accuracy and 66.66% specificity have been obtained (Fig. 5). Here in this work, only mean features were used. The comparative study with AAD- and GMF-based features would be a matter of further investigation.

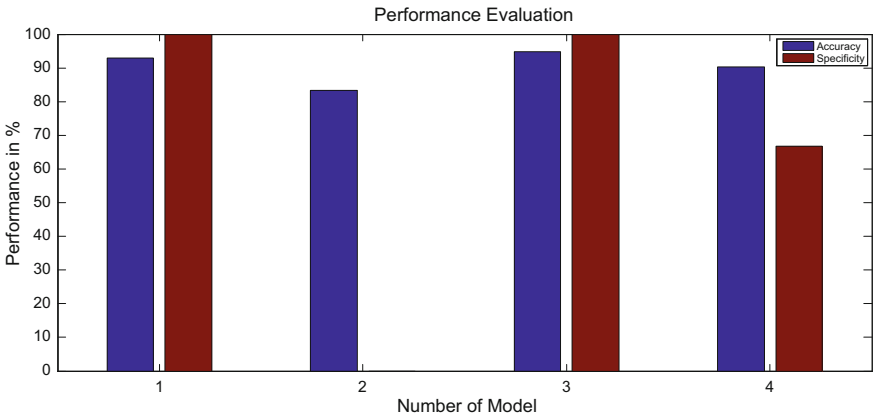


Fig. 5 Performance and specificity

References

1. A.K. Jain, A. Ross, S. Prabhakar, An introduction to biometric recognition. IEEE Trans. Circuit Syst. Video Technol. **14**, 4–20 (2004)
2. A.K. Jain, A. Ross, Information fusion in biometrics. Pattern Recogn. Lett. **24**, 2115–2125 (2003)
3. A.A. Ross, K. Nandkumar, A.K. Jain, *Handbook of Multibiometrics*, International Series on Biometrics, vol. 6 (Springer Publisher, 2006), XXI, p. 198
4. A.S. Raju, V. Udayashankara, Biometric person authentication: a review, in *Proceeding of International Conference on Contemporary Computing and Informatics (ICI3)*, Chennai, India, July 2014
5. M.M. Monwar, M.L. Gavrilova, Multimodal biometric system using rank level fusion approaches (2009)
6. M. Haghighat, M. Abdel-Mottaleb, W. Alhalabi, Discriminant correlation analysis: real time feature level fusion for multimodal biometric recognition. IEEE Trans. Inf. Forensic Secur. **4**, 1–11 (2016)
7. U. Gawande, A novel algorithm for feature level fusion using SVM classifier for multibiometrics-based person authentication. Appl. Comput. Intell. Soft Comput. (2013)
8. S. Al Zahir, Precise multimodal Biometric fusion method using copula and QR codes. J. Biostat. Biom. **1**(5), 1–7 (2017)
9. N. Celik, A short review of multimodal biometric recognition systems. J. Biom. Biostat. **8**(3) 2017
10. C. Rathgeb, A. Uhl, A survey on biometric cryptographic and cancelable biometrics. EURASIP J. Inf. Secur. **8**(3) (2017)
11. X. Xu, Y. Zhao, H. Li, The study of feature level fusion algorithm for multimodal recognition. IEEE Trans. Inf. Forensics Secur. **7**(1), 255–268 (2012)
12. M.O. Oloyede, Gerhard P. Hancke, Unimodal and multimodal biometric sensing systems: a review. IEEE Access **4**, 7532–7558 (2016)
13. S.M.S. Islam, M. Beennamoun, A.S. Mian, D.R. Davis, Score level fusion of ear and face local 3D feature for fast and expression-invariant Human recognition. *ICIAR 2009*, LNCS 5627 (Springer, Berlin, Heidelberg, 2009), pp. 387–396
14. M. Ghayoumi, A review of multimodal biometric systems: fusion methods and their application, in *Proceedings of ICIS IEEE Conference* (Las Vegas, July 2015)

Author Index

R. Kamal et al. (eds.), *International Conference on Advanced Computing Networking and Informatics*, Advances in Intelligent Systems and Computing 870,
https://doi.org/10.1007/978-981-13-2673-8